Міністерство освіти і науки України
Харківський національний
університет імені В. Н. Каразіна

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ В НАУКОЄМНИХ ТЕХНОЛОГІЯХ

ЗБІРНИК НАУКОВИХ ПРАЦЬ МІЖНАРОДНОЇ НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

Харків, 21-23 квітня 2021 року

Харків
2021

Міністерство освіти і науки України
Харківський національний
університет імені В. Н. Каразіна

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ В НАУКОЄМНИХ ТЕХНОЛОГІЯХ

ЗБІРНИК НАУКОВИХ ПРАЦЬ МІЖНАРОДНОЇ НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

Харків, 21-23 квітня 2021 року

Харків
2021

Реєстраційне посвідчення в ДНУ «Український інститут науково-технічної експертизи та інформації» (посвідчення № 810 від 14.12.2020 р.).

Затверджено до друку рішенням Вченої ради Харківського національного університету імені В. Н. Каразіна (протокол № 6 від 31.05.2021 р.).

Редакційна колегія:
Азарєнков Н.А. (гол. редактор), д.ф.-м.н., академік НАН України, проф., ІВТ ХНУ імені В.Н. Каразіна
Ванін В .А., д.т.н., проф., НТУ «ХПІ»
Горбенко І.Д., д.т.н., проф., ФКН ІВТ ХНУ імені В.Н. Каразіна
Доля Г.М., проф., д.т.н., проф. ФКН ІВТ ХНУ імені В.Н. Каразіна
Жолткевич Г.Н., д.т.н., проф., ФМІ ХНУ імені В.Н. Каразіна
Куклін В.М., д.ф.-м.н., проф., ФКН ІВТ ХНУ імені В.Н. Каразіна
Лазурик В.Т., д.ф.-м.н., проф., ФКН ІВТ ХНУ імені В.Н. Каразіна
Рассомахін С.Г., д.т.н., доц., ФКН ІВТ імені В.Н. Каразіна
Споров О.Є., к.ф.-м.н., доц. ХНУ імені В.Н. Каразіна
Стєрвоєдов М.Г., к.т.н., доц., ФКН ІВТ імені В.Н. Каразіна
Толстолузька О. Г., д.т.н.,с.н.с., доц., ФКН ІВТ ХНУ імені В.Н. Каразіна
Ткачук М.В., д.т.н., проф., ФКН ІВТ ХНУ імені В.Н. Каразіна
Харченко В.С., д.т.н., проф., НАУ імені М.Є. Жуковського
Шматков С.І., д.т.н., проф., ХНУ імені В.Н. Каразіна.
Шульга М.Ф. д.ф.-м.н., акад. НАНУ, проф., ННЦ ХФТІ НАНУ

Адреса редакційної колегії: 61022, м. Харків, майдан Свободи, 6,
ХНУ імені В. Н. Каразіна, к. 534.
Тел. +380 (57) 705-42-81, email: kmht@karazin.ua.

Комп'ютерне моделювання в наукоємних технологіях: Збірник наукових праць міжнародної науково-технічної конференції (м. Харків, 21-23 квітня 2021 року) – Харків: ХНУ ім. В.Н. Каразіна, 2021. – 308 с.

Доповіді, що увійшли до збірника, висвітлюють такі напрямки:

1 секція: Математичне моделювання технологічних процесів та приладів.
2 секція: Моделювання інформаційних процесів у складних і розподілених системах.
3 секція: Системи автоматизованого збору та когнітивного представлення наукових даних.
4 секція: Моделювання фізичних процесів в радіаційних, плазмових та інших сучасних технологіях.
5 секція: Безпека інформаційних систем і технологій.
6 секція: Моделі процесів розробки та оцінки якості програмного забезпечення.
7 секція «Студентів, аспірантів та молодих вчених».

Для викладачів, наукових працівників, аспірантів, студентів вишів.
Робочі мови конференції: українська, англійська.

ISBN 978-966-285-699-6

© Харківський національний університет імені В.Н.Каразіна, 2021

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ В. Н. КАРАЗІНА
ННЦ ХАРКІВСЬКИЙ ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
MAX PLANCK INSTITUTE OF MICROSTRUCTURE PHYSICS
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА
INSTITUTE OF NUCLEAR CHEMISTRY AND TECHNOLOGY (Warsaw, Poland)
РІВНЕНСЬКИЙ ДЕРЖАВНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ
НАЦІОНАЛЬНИЙ АЕРОКОСМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ М. Є. ЖУКОВСЬКОГО (ХАРКІВ)
ЗАТ « ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ » (ХАРКІВ)
ХЕРСОНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ТОВ "БЮРО ІРІС" (КИЇВ)
TEAM INTERNATIONAL SERVICES, INC. (Lake Mary, USA)

ПРОГРАМНИЙ КОМІТЕТ:

Азарєнков М.О., акад. НАНУ, проф., д.ф.-м.н., Харків, голова
Бардачов Ю. М., проф., д.т.н., Херсон
Бомба А.Я., проф., д.т.н., Рівне
Буй Д. Б., проф., д.ф.-м.н., Київ
Ванін В .А., проф., д.т.н., Харків
Горбенко І.Д., проф., д.т.н., Харків
Доля Г.М., проф., д.т.н., Харків
Жолткевич Г.М., проф., д.т.н., Харків
Куклін В.М., проф., д.ф.-м.н., Харків
Лазурик В.Т., проф., д.ф.-м.н., Харків
Рассомахін С.Г., проф., д.т.н., Харків
Савула Я. Г, проф., д.ф.-м.н., Львів
Споров О. Є., доц., к. ф.-м.н., Харків
Стєрвоєдов М.Г., доц., к.т.н., Харків
Styervoyedov A. Dr., Halle, Germany
Толстолузька О.Г., проф., д.т.н., Харків
Ткачук М.В., проф., д.т.н., Харків
Харченко В.С., проф., д.т.н., Харків
Хомченко А.Н. проф., д.ф.-м.н., Миколаїв
Шматков С.І., проф., д.т.н., Харків
Шульга М.Ф., акад. НАНУ, проф., д.ф.-м.н., Харків
Zimek Z., Ph.D., Warsaw, Poland
Яновський В.В., проф., д.ф.-м.н., Харків

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ:

Лазурик В.Т., д.ф.-м.н., проф., декан ФКН ХНУ імені В.Н. Каразіна, голова,
Споров О.Є., к.ф.-м.н., доц. ХНУ імені В.Н. Каразіна, заст. голови,
Толстолузька О.Г., д.т.н., проф. ХНУ імені В.Н. Каразіна, заст. голови
Ткачук М.В., д.т.н., проф., зав. каф. МСіТ ХНУ імені В.Н. Каразіна
Куклін В.М., д.ф.-м.н., проф., зав. каф. ШІ та ПЗ ХНУ імені В.Н. Каразін,
Дюльдя С.В., к.ф -м.н., ХФТІ,
Єсін В.І., д.т.н., проф., ХНУ імені В.Н. Каразіна,
Артюх О.А., зав. лаб. ХНУ імені В.Н. Каразіна,
Шевцов С. О., директор ТОВ Бюро ІРІС, (Київ)
Жолткевич Г.М., д.т.н., проф., декан ФМІ ХНУ імені В.Н. Каразіна,
Ванін В .А., д.т.н., проф., НТУ «ХПІ» (Харків),
Зінов'єв Д.В., ст. викл. ХНУ імені В.Н. Каразіна,
Рассомахін С.Г., д.т.н., проф., зав. каф. БІСТ ХНУ імені В.Н. Каразіна,
Styervoyedov A. Dr., Max Planck Institute of Microstructure Physics (Germany),
Петерсен С., виконавчий директор TEAM International (Харків),
Стєрвоєдов М.Г., к.т.н., доц., зав. каф. ЕУС ХНУ імені В.Н. Каразіна,
Шматков С.І., д.т.н., проф., зав. каф. ТПС ХНУ імені В.Н. Каразіна.
Кругол М.М., асистент НТУ «ХПІ»

http:// www.univer.kharkov.ua
http:// www-csd.univer.kharkov.ua

ЗМІСТ

ЗМІСТ 4

АНДРЄЄВ Ф. М., ОСИПЧУК А. В., СТЄРВОЄДОВ М. Г.
ВИБІР ЕЛЕМЕНТІВ ДЕТЕКТОРНОГО БЛОКУ СИСТЕМИ ДЛЯ ВИМІРЮВАННЯ КООРДИНАТ ДЖЕРЕЛ ГАММА-ВИПРОМІНЮВАННЯ. 10

АНЖУРОВ В.Є., ТОЛСТОЛУЗЬКА О.Г.
МОДЕЛЬ КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ АНАЛІЗУ ІНДЕКСІВ АКЦІЙ НА ФОНДОВОМУ РИНКУ 14

АСЄЄВ Б.А, ПОПОВА А.О, ГУЩІН І.В.
АНАЛІЗ КОМП'ЮТЕРНИХ СИСТЕМ ТА ЕЛЕКТРОННИХ СКЛАДОВИХ ДЛЯ ВІДСТЕЖУВАННЯ ПАСАЖИРСЬКИХ ПЕРЕВЕЗЕНЬ 16

БАКУМЕНКО Н.С, ДОНЕЦЬ В.В, ШЕВЧЕНКО Д.О, ОДИНЕЦЬ О.О, УГРЮМОВ М.Л.
МЕТОДИ КЛАСТЕРИЗАЦІЇ ДАНИХ НА ОСНОВІ ІНФОРМАЦІЙНИХ КРИТЕРІЇВ. 20

БЕЗЛЮБЧЕНКО А. В., УГРЮМОВ М. Л., МЕНЯЙЛОВ Є. С.
МЕТОД ГЛИБОКОГО НАВЧАННЯ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ У РАЗІ НЕДИФЕРЕНЦІЙОВАНИХ ФУНКЦІЙ ВТРАТ. 24

БЕЗСМЕРТНИЙ Д.Р., КОНДРАТЮК Д.С.
МОДЕЛІ КЛАСИФІКАЦІЇ СТАНІВ КОМП'ЮТЕРНОЇ МЕРЕЖІ НА ОСНОВІ ОПТИМІЗАЦІЇ РОЮ ЧАСТОК ТА БДЖОЛИНОГО РОЮ. 28

BELKOV D.O.
COMPUTER MODEL OF THE SYSTEM OF ANALYSIS AND FORECASTING OF FINANCIAL TIME SERIES USING NEURAL NETWORK MODELS. 32

BENHADDOU M., KHRUSLOV M.
ROBUST ESTIMATION FOR MODEL CONSUMPTION ENERGY CARS. 35

БІЛЬСЬКИЙ Г.М., ЛАБЕНКО Д.П.
СТРУКТУРА КОМПЛЕКСУ ПЗ ДЛЯ УПРАВЛІННЯ ПРОГРАМНИМ ПРОДУКТОМ ТА ЙОГО РОЗРОБКОЮ НА ПРИКЛАДІ ІНТЕРАКТИВНОЇ СИСТЕМИ РОЗКЛАДУ ЗАНЯТЬ ДЛЯ ЗАКЛАДУ ВИЩОЇ ОСВІТИ. 37

БОМБА А.Я, МОРОЗ І.П.
МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ СТАЦІОНАРНИХ ПРОЦЕСІВ В АКТИВНІЙ ОБЛАСТІ ПОВЕРХНЕВО-ОРІЄНТОВАНИХ P-I-N СТРУКТУР МЕТОДАМИ ТЕОРІЇ ЗБУРЕНЬ 41

БОНДАРЕНКО Є.В.
МЕТОД ВІДНОВЛЕННЯ ДАНИХ В АВТОМАТИЗОВАНИХ СИСТЕМАХ КОНТРОЛЮ ТА ДІАГНОСТИКИ СТАНУ ОБ'ЄКТІВ УПРАВЛІННЯ. 45

БРАТУСЬ М.А.
ІМОВІРНІСНІ МЕТОДИ ПОБУДОВИ ПРОБНИХ ВИБІРОК ПО ЗАДАНОМУ ЗАКОНУ РОЗПОДІЛУ. 49

БУБЕР Д.И. ПАВЛОВ А.Н.
МЕТОДИКА РАСЧЕТА НАДЕЖНОСТИ ДЛЯ ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ НА БАЗЕ "ТОЛСТЫХ" КЛИЕНТОВ 53

БУДЬКО В.В., ТОЛСТОЛУЗЬКА О.Г.
АНАЛІЗ КОМП'ЮТЕРНИХ ЗАСОБІВ ІНКЛЮЗИВНОГО НАВЧАННЯ ДЛЯ ЛЮДЕЙ З ВАДАМИ СЛУХУ. 58

БУЄВИЧ-СИСОЄВ В.М., ШМАТКОВ С.І.
ІНФОРМАТИВНІСТЬ ПАРАМЕТРІВ КОМП'ЮТЕРНИХ СИСТЕМ 60

BUZKO K.S., NARIEZHNII O.P., GRINENKO T.O.
ANALYSIS OF CYBER THREATS IN THE LOCAL SYSTEM OF DIFFERENTIAL CORRECTION OF GPS SIGNALS 63

BULAVIN D. O.
QUALITY CONTROL AND PROJECT MANAGER ROLE IN OUTSOURCING/ OUTSTAFFING COMPANIES 65

БУРЛАЙ І.С., БАКУМЕНКО Н.С.
КОМП'ЮТЕРНА МОДЕЛЬ СИСТЕМИ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ЦІН НА ЕНЕРГОНОСІЇ ЗА ДОПОМОГОЮ МЕТОДУ ARIMA (p,q,k) 68

ВАНІН В.А., КРУГОЛ М.М., ЛАЗУРЕНКО О.П.
МАТЕМАТИЧНІ МОДЕЛІ ТА РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ ГІДРАВЛІЧНИХ СИСТЕМ ТЕС. 71

ВАРЛАМОВА Н.В., СТЄРВОЄДОВ М.Г.
ІНФОРМАЦІЙНО-ВИМІРЮВАЛЬНА ПСИХОДІАГНОСТИЧНА СИСТЕМА: ПРИНЦИПИ І МЕТОДИ ПОБУДОВИ 75

VIERUSHKIN I.A.
HYPERSINGULAR INTEGRAL EQUATION METHOD IN NUMERICAL SIMULATING FREQUENCIES AND MODES OF CIRCULAR PLATE IMMERSED INTO LIQUID. 79

ВІЛІГУРА В.В.
СИСТЕМАТИЗАЦІЯ ЗАГРОЗ І ВРАЗЛИВОСТЕЙ ХАРАКТЕРНИХ ДЛЯ БАЗ ДАНИХ І СУБД. 83

ВОЛИНСЬКИЙ В.В, БЕРДНІКОВ А.Г.
МОДЕЛЮВАННЯ SMART-ЛАБОРАТОРІЇ ДЛЯ НАВЧАЛЬНОГО ПРОЦЕСУ. 87

ГАВРИЛЮК Є.А., СПОРОВ О.Є., ГУЩИН І.В.
ПОРІВНЯННЯ ЗАСТОСОВНОСТІ АЛГОРИТМІВ НАВЧАННЯ З ПІДКРІПЛЕННЯМ У МУЛЬТИАГЕНТНИХ РОЗПОДІЛЕНИХ СИСТЕМАХ 90

ГАЙКОВА В.В., МАЛАХОВ С.В.
УЗАГАЛЬНЕННЯ УМОВ РЕАЛІЗАЦІЇ АКЦІЙ КІБЕРБУЛІНГУ З УРАХУВАННЯМ ОСОБЛИВОСТЕЙ СУЧАСНИХ ІНФОРМАЦІЙНИХ СИСТЕМ 95

ГАЛЮК Я.О, ТОВСТОКОРЕНКО О.Ю.
ПРО МЕТОДИ РОЗРОБКИ ТА КРИТЕРІЇ ОЦІНКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМ «РОЗУМНИЙ БУДИНОК» 100

ГАМЗАЄВ Р.О., ТКАЧУК М.В., ШЕВКОПЛЯС Д.О.
ЗАСТОСУВАННЯ ЗНАННЯ-ОРІЄНТОВАНИХ МЕТОДІВ І ТЕХНОЛОГІЙ ДЛЯ МОДЕЛЮВАННЯ ВАРІАБЕЛЬНОСТІ В РОЗРОБЦІ ЛІНІЙОК ПРОГРАМНИХ ПРОДУКТІВ. 104

ГАРМАШ Д.В, МАЛЄЄВА А.А.
ГЕНЕРАЦІЯ КЛЮЧІВ І ПІДПИСУ ЗА ДОПОМОГОЮ БАГАТОВИМІРНОГО КРИПТОГРАФІЧНОГО АЛГОРИТМУ RAINBOW 108

ГЕТМАНЕЦЬ О.М.
МОДЕЛЮВАННЯ ВИРОБІТКУ ВІТРОВОЇ ЕНЕРГЕТИЧНОЇ УСТАНОВКИ. 111

ГОЛОВЧЕНКО В.Б., СТРІЛЕЦЬ В.Є.
ВИПАДКОВИЙ ЛІС – УНІВЕРСАЛЬНИЙ МЕТОД МАШИННОГО НАВЧАННЯ. 115

ГОНЧАРОВ М.О, МАЛАХОВ С.В.
МОДЕЛЮВАННЯ ПРОЦЕДУР ПІДГОТОВКИ ДАНИХ СТЕГАНОАЛГОРІТМА З БАГАТОРІВНЕВИМ МУЛЬТИПЛЕКСУВАННЯМ КОНТЕНТУ 118

ГОРБЕНКО І.Д., ЗАМУЛА О.А., ХО ЧІ ЛИК
ОПТИМІЗАЦІЯ ПОШУКУ СКЛАДНИХ СИГНАЛІВ НА ОСНОВІ ОЦІНКИ ЇХ КОРЕЛЯЦІЙНИХ І СТАТИСТИЧНИХ ХАРАКТЕРИСТИК 123

ГОРБЕНКО Ю.І., ПОНОМАР В.А., ЄСІНА М.В., ГОРБЕНКО І.Д.
СТАН РОЗРОБКИ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ ДОКАЗОВО СТІЙКИХ ЕЛЕКТРОННИХ ПІДПИСІВ ДЛЯ ПОСТКВАНТОВОГО ПЕРІОДУ. 127

ГУЩІН І.В., СЕРДЮК С.А, СПОРОВ О.Є.
ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ЗАСТОСУВАННЯ АРХІТЕКТУРИ НЕЙРОННОЇ МЕРЕЖІ GAN 131

ДЕРЕВ'ЯНКО Я.А., ГОРБЕНКО І.Д.
ПОРІВНЯННЯ ШВИДКОДІЇ РЕАЛІЗАЦІЙ АЛГОРИТМУ ЕЛЕКТРОННОГО ПІДПИСУ FALCON НА C ТА PYTHON. 134

ДОЛЯ Г.М, БОНДАРЕНКО К. О.
ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЛАЗЕРНОГО ДЕТЕКТУВАННЯ АКУСТИЧНИХ КОЛИВАНЬ У ПОВІТРІ З ВИКОРИСТАННЯМ СВІТЛОПОВЕРТАЮЧИХ ПОВЕРХОНЬ. . . . 138

ДОРОШЕНКО М.І., БУЄВИЧ-СИСОЄВ В.М.
МОДЕЛЬ ОЦІНКИ ПАРАМЕТРІВ КОМП'ЮТЕРНОЇ МЕРЕЖІ. 143

Д'ЯЧЕНКО А.С., ГОРБЕНКО І.Д.
ДОСЛІДЖЕННЯ МАТЕМАТИЧНОЇ СТРУКТУРИ ТА ПРОЦЕСУ ГЕНЕРАЦІЇ КЛЮЧОВОЇ ПАРИ У АЛГОРИТМІ ЕП RAINBOW. 146

ДЯЧЕНКО В.Г., МІНЯЙЛОВ Є.С., УГРЮМОВ М.Л.
ІМОВІРНІСТНІ МЕТОДИ ГЕНЕРАЦІЇ ПРОБНИХ ВИБІРОК, ЩО МАЮТЬ ЗАДАНИЙ РОЗПОДІЛ 150

ЄСІН В.І, СОКОЛОВСЬКИЙ К.О.
КОНТРОЛЬ ЦІЛІСНОСТІ ЗБЕРЕЖЕНИХ ПРОГРАМ БАЗ ДАНИХ НА ОСНОВІ ВИКОРИСТАННЯ СТРУКТУРИ ҐЕШ-ДЕРЕВА 154

ЖМИРОВ Д.А, БЕРДНІКОВ А.Г.
МОДЕЛЮВАННЯ РИЗИКІВ ПРИ РЕАЛІЗАЦІЇ ІТ-ПРОЕКТІВ 158

ЗІНОВ'ЄВ Д.В., ТКАЧУК М.В.. ТРІЩЕНКО І.В.
МОДЕЛІ ТА ТЕХНОЛОГІЇ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ СЕРВІС-ОРІЄНТОВАНИХ ПРОГРАМНИХ СИСТЕМ: СУЧАСНИЙ СТАН ТА ПЕРСПЕКТИВНІ НАПРЯМКИ ДОСЛІДЖЕНЬ 161

КАМИШНІК Д.О, СПОРОВ О.Є, ГУЩИН І.В.
АНАЛІЗ СИСТЕМ ГЕНЕРАЦІЇ ТЕКСТУ НА БАЗІ НАУКОВИХ ТЕЗИСІВ 165

КАПЛУН В.М., ТОЛСТОЛУЗЬКА О.Г.
РОЗРОБКА КОМП'ЮТЕРНОЇ МОДЕЛІ АВТОМАТИЗОВАНОГО РОБОЧОГО МІСЦЯ ПСИХОЛОГА 169

КАПТЬОЛ Є.Ю., КУЗНЕЦОВА К.О.
АНАЛІЗ ПОСТКВАНТОВОГО АЛГОРИТМУ ЕЛЕКТРОННОГО ПІДПИСУ RAINBOW ТА ПОСТКВАНТОВИХ АТАК НА НЬОГО 172

КАПУСТА М.А.
МОБІЛЬНИЙ РОБОТ ДЛЯ РОЗПІЗНАВАННЯ ТА СПОСТЕРЕЖЕННЯ 176

КАЧКО О.Г., КАНДІЙ С.О.
ОПТИМІЗАЦІЯ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ЕП FALCON З ВИКОРИСТАННЯМ AVX512 ІНСТРУКЦІЙ 180

КИРИЛОВА Є.О.
МАТЕМАТИЧНА МОДЕЛЬ СИСТЕМИ АВТОМАТИЗОВАНОГО РАДІОМОНІТОРИНГУ. . . 184

КОВАЛЕНКО А. В., БЕРДНІКОВ А. Г.
ПОРІВНЯЛЬНИЙ АНАЛІЗ ЕФЕКТИВНОСТІ ОСНОВНИХ ІНСТРУМЕНТІВ КОНТРОЛЮ ЯКОСТІ ІТ-ПРОЕКТУ 187

КОЗЛОВ В.О., ЛАЗУРИК В.М.
ДОСТУП ДО БАЗ ДАНИХ З ВИКОРИСТАННЯМ ІНСТРУМЕНТІВ PRISMA 190

КОНОНЕНКО Є.С.
КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ КОЛИВАНЬ КРУГЛИХ ПЛАСТИН ЗА РІЗНІ УМОВИ ЗАКРІПЛЕННЯ МЕТОДОМ СКІНЧЕННИХ ЕЛЕМЕНТІВ 196

КУРГАН А.Р., КОТВИЦЬКИЙ А.Т.
ФІЗИКО-МАТЕМАТИЧНІ МЕТОДИ АНАЛІЗУ РУХОМИХ ПЛАТФОРМ. 199

ЛИСИЦЯ О.Ю., КОВАЛЬОВ А.В., МИХАЙЛЕНКО Т.П., ПЕТУХОВ І.І.
МОДЕЛЮВАННЯ ПРОЦЕСІВ УТВОРЕННЯ МАСЛЯНОЇ ПЛІВКИ НА СТІНКАХ КАМЕРИ ПІДШИПНИКА ГТД 202

МАЗОРЧУК М.С. ЛІСОВА Т.В.
МОДЕЛІ АНАЛІЗУ ЛАТЕНТНИХ ФАКТОРІВ У МІЖНАРОДНИХ ДОСЛІДЖЕННЯХ ОЦІНЮВАННЯ ЯКОСТІ ОСВІТИ (НА ПРИКЛАДІ PISA) 205

МАКСИМЕНКО В.Ю., ТОЛСТОЛУЗЬКА О.Г.
ДОСЛІДЖЕННЯ КОМП'ЮТЕРНОЇ МОДЕЛІ ВЕБ-СЕРВІСУ ПЕРЕВІРКИ ЗНАНЬ. 208

МАКСИМУК А.Р., БАКУМЕНКО Н.С.
МОДЕЛЬ ІНФОРМАЦІЙНОЇ СИСТЕМИ КЛАСИФІКАЦІЇ ПАЦІЄНТІВ ЗА ДОПОМОГОЮ ЙМОВІРНІСНИХ ШТУЧНИХ МЕРЕЖ. 211

МАРЧЕНКО І.Г., АКСЕНОВА В.Ю., ЖИГЛО А.В., ТКАЧЕНКО В.І.
КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ ПРИСКОРЕННЯ ДИФУЗІЇ ЧАСТИНОК В ПРОСТОРОВО-ПЕРІОДИЧНИХ СТРУКТУРАХ ЗОВНІШНІМИ ПОЛЯМИ 214

МИРОНЕНКО М.Л.
ДОСЛІДЖЕННЯ ВІЛЬНИХ КОЛИВАНЬ РІДИНИ В ОБОЛОНКАХ, ЧАСТКОВО ЗАПОВНЕНИХ РІДИНОЮ, ПРИ РІЗНИХ ПАРАМЕТРАХ ПЕРЕВАНТАЖЕННЯ 218

МІГАЛЬ Д.О., БОБУХ В.А.
ВАРІАЦІЇ ПОСТКВАНТОВОГО МЕТОДУ ЕЛЕКТРОННОГО ПІДПИСУ RAINBOW 221

МОРОЗ О. Ю., ТОЛСТОЛУЗЬКА О. Г.
АНАЛІЗ ЗАСОБІВ ТЕХНОЛОГІЙ ВЕРИФІКАЦІЇ ПАРАЛЕЛЬНИХ ПРОГРАМ 224

НЄВЄЖИНА В.Ю., СТРІЛЕЦЬ В.Є.
МОДЕЛІ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ НЕЧІТКОЇ ЛОГІКИ 228

ОСТРЯНСЬКА Є.В., ЄСІНА М.В.
ГЕНЕРАЦІЯ ЗАГАЛЬНОСИСТЕМНИХ ПАРАМЕТРІВ ДЛЯ СХЕМИ ЕЛЕКТРОННОГО ПІДПИСУ RAINBOW 231

ПАВЛОВ А.М, ХИЖНЯК О.В.
ОПТИМІЗАЦІЯ СТРУКТУРИ СКЛАДНОЇ СИСТЕМИ ЗА КРИТЕРІЄМ НАДІЙНОСТІ 235

ПАНЧЕНКО Н.В.
МАТЕМАТИЧНА МОДЕЛЬ СИСТЕМИ ВЕРИФІКАЦІЇ ВІДБИТКІВ ПАЛЬЦІВ НА ОСНОВІ ЦИЛІНДР-КОДІВ. 239

ПЕТУХОВ І.І., ЛИСИЦЯ О.Ю.
ОСОБЛИВОСТІ CFD-МОДЕЛЮВАННЯ РОБОЧОГО ПРОЦЕСУ СТРУМИННОГО ОХОЛОДЖУВАЧА РІДИНИ. 242

ПОМОГАЄВ К.О, АРТЮХ О.А.
МОДЕЛЬ 3D-ВІЗУАЛІЗАЦІЇ НАВЧАЛЬНИХ ПРИМІЩЕНЬ КАФЕДРИ. 245

ПОНОМАРЕНКО В.В.
ПОРІВНЯННЯ І АНАЛІЗ CONTENT MANAGEMENT SYSTEM ТА WEB CONTENT MANAGEMENT SYSTEM. 249

ПУДОВКІНА Л.Ф.
СЦЕНАРІЇ ТЕСТУВАННЯ ВИМОГ І ЯКІСТЬ ПРОГРАМ 251

РОСОХАТИЙ К.С.
МОБІЛЬНА GPS-ОРІЄНТОВАНА СИСТЕМА ДОПОВНЕНОЇ РЕАЛЬНОСТІ ДЛЯ ТУРИСТИЧНОЇ СФЕРИ 253

РУЖАНСЬКА А. В., ВАСИЛЬЄВА Л. В.
ОГЛЯД ТА АНАЛІЗ БІБЛІОТЕК КОМП'ЮТЕРНОГО ЗОРУ ДЛЯ ОБРОБКИ ЗОБРАЖЕНЬ, ВИКОРИСТОВУВАНИХ В МАШИННОМУ НАВЧАННІ 256

РУЗУДЖЕНК С.Р.
РОЗРОБКА СТРУКТУРНОЇ СХЕМИ СИСТЕМИ ВЗАЄМОДІЮЧИХ ТЕХНОЛОГІЙ ДЛЯ НАВЧАННЯ НЕЙРОННИХ МЕРЕЖ 260

СЄМЄНЯКІНА В.С., ЄСІНА М.В..
АТАКИ НА ЕЛЕКТРОННИЙ ПІДПИС DILITHIUM. 264

СЄРІКОВА О.М., СТРЕЛЬНІКОВА О.О.
МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ПАРАМЕТРІВ РУКАВНИХ ФІЛЬТРІВ 268

СТЄРВОЄДОВ М.Г.,АРТЮХ С.О.
ЦІЛІ ТА ЗАВДАННЯ СТВОРЕННЯ ІНФОРМАЦІЙНО-УПРАВЛЯЮЧИХ СИСТЕМ ДЛЯ ПРИСКОРЮВАЧІВ ЗАРЯДЖЕНИХ ЧАСТОК 271

TEROKHIN V., STYERVOYEDOV M.
THE INTELLIGENT DETECTOR FOR SENSOR NODES OF THE RADIATION MONITORING NETWORK. 273

ТЕСЛЕНКО А.Ю., ПОГОРІЛА К.В.
МАТЕМАТИЧНА МОДЕЛЬ ПОБУДОВИ ТА ОБРОБКИ OFDM СИГНАЛІВ НА ФІЗИЧНОМУ РІВНІ WIFI ПРОТОКОЛІВ 276

ТКАЧЕНКО А.М., СТРІЛЕЦЬ В.Є.
ШТУЧНІ НЕЙРОННІ МЕРЕЖІ У ЗАДАЧАХ КЛАСИФІКАЦІЇ. 279

ТОВСТОКОРЕНКО О.Ю.
ПРО УПРАВЛІННЯ ВАРІАБЕЛЬНІСТЮ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У РОЗРОБЦІ ТА СУПРОВОДІ СИСТЕМ «РОЗУМНИЙ БУДИНОК». 282

TOLSTOLUZKYI Y. D., BERDNIKOV A. G
INVESTIGATION OF PARAMETERS OF SEMANTIC-NUMERICAL SPECIFICATIONS OF PARALLEL PROCESSES 285

ТУРЕНКО М.С., СТРІЛЕЦЬ В.Є.
АНАЛІЗ СТРАТЕГІЙ ТЕСТУВАННЯ ВЕБ-САЙТІВ. 289

УМАНЕЦЬ К.В.
ПРОБЛЕМИ РАНЖИРУВАННЯ РЕЛЕВАНТНИХ РЕЗУЛЬТАТІВ, ЩО ПОВЕРТАЮТЬСЯ ПО РІЗНИХ ПОШУКОВИХ ЗАПИТАХ. 292

УСАТОВА О.О.
МОДЕЛЮВАННЯ РУХУ РІДИНИ В ЦИЛІНДРИЧНИХ ОБОЛОНКАХ. 295

ШАРОВ В.О., БЕРДНІКОВ А. Г.
МОДЕЛЮВАННЯ КОРИГУВАЛЬНОГО КАСКАДНОГО КОДУ В КАНАЛАХ ПЕРЕДАЧІ ДАНИХ СИСТЕМИ УПРАВЛІННЯ 298

ШВИДКИЙ Ю.К. СПОРОВ О.Є. ГУЩІН І.В.
ДОСЛІДЖЕННЯ КОМПРОМІСІВ ПРИ ПРОЕКТУВАННІ РОЗПОДІЛЕНИХ СИСТЕМ ІЗ ВИКОРИСТАННЯМ САР ТЕОРЕМИ. 303

УДК 004.3+621.039 **СЕКЦІЯ 7**

АНДРЄЄВ Ф. М., ОСИПЧУК А. В., СТЄРВОЄДОВ М. Г.

ВИБІР ЕЛЕМЕНТІВ ДЕТЕКТОРНОГО БЛОКУ СИСТЕМИ ДЛЯ ВИМІРЮВАННЯ КООРДИНАТ ДЖЕРЕЛ ГАММА-ВИПРОМІНЮВАННЯ

Актуальною остається проблема створення сучасних, ефективних і надійних систем контролю за радіаційною обстановкою та пошуку ядерних радіоактивних матеріалів і радіаційних джерел, які вийшли з-від регульованого нагляду. Такі засоби вимірювання іонізаційних випромінювань (основа вказаних систем), як іонізаційні камери, пропорційні та газорозрядні лічильники, сцинтиляційні та піроелектричні детектори не в повному обсязі відповідають потребам сьогодення. Для захисту навколишнього середовища та запобіганню наслідків ядерної контрабанди є перспективній метод неруйнівного контролю з використанням напівпровідникових детекторів з поглиначами [1─3].

Поза увагою дослідників виявилося питання, пов'язане з науково обґрунтованим вибором елементів засобу вимірювання іонізаційних випромінювань, в якому використовується вказаний вище метод.

Мета статті: вибір елементів детекторного блоку засобу, в якому використовується метод поглинання для визначення координат джерела гамма-випромінювання.

В даному способі для визначення напрямку на імпульсні та постійні джерела гамма-випромінювання використовується відношення сигналів, які приходять з детекторів, що розташовані в поглиначі заданої геометричної форми.

Проходження моноенергетичного пучка гамма-випромінювання через речовину описується законом Бугера, що враховує тільки ефект поглинання гамма-квантів,

$$I = I_0 e^{-\mu(E)h}, \quad (1)$$

де I_0 інтенсивність гамма-випромінювання до входу в речовину ($h=0$), h ─ товщина речовини, $\mu = \mu_{ФЭ} + \mu_{КЭ} + \mu_{ПАР}$ ─ лінійний коефіцієнт ослаблення гамма-випромінювання, який залежить від енергії гамма-випромінювання, I_0 і характеристик поглинаючої речовини, $\mu_{ФЭ}, \mu_{КЭ}, \mu_{ПАР}$ ─ коефіцієнти ослаблення, які обумовлені фотоефектом, Комптон-эфектом, утворенням електронно-позитронних пар відповідно. Енергія від джерела гамма-випромінювання, розташованого на відстані R від точки прийому, поширюється в радіальному напрямку, так що в точці прийому щільність потоку гамма-випромінювання дорівнює:

$$\gamma_0 = \frac{I_0}{4\pi R^2} \left(\frac{МэВ}{см^2 с}\right).$$

У разі використання ефекту поглинання гамма-випромінювання для пошуку, виявлення і вимірювання положення джерел на місцевості щільність потоку гамма-випромінювання, який дійшов до детектора, має вигляд:

$$\gamma_d = \frac{I_0 e^{-\mu(I_0)h}}{4\pi R^2} f_П(I_0), \quad (2)$$

де $f_П(I_0)$ ─ енергетичний чинник накопичення для даної речовини, що залежить від інтенсивності гамма-випромінювання і враховує розсіювання гамма-квантів, що має місце при Комптон-ефекту. Даний фактор $f_П(I_0)$ визначається для поглинаючої речовини експериментально. З урахуванням (2) щільність потоку гамма-випромінювання яке фіксується детектором j записується:

$$j = \gamma_d(I_0) f_d(I_0) = \frac{I_0 e^{-\mu(I_0)h}}{4\pi R^2} f_П(I_0) f_d(I_0), \quad (3)$$

© АНДРЄЄВ Ф. М., ОСИПЧУК А. В., СТЄРВОЄДОВ М. Г., 2021

де $f_d(I_0)$ – залежність рахункового числа від щільності потоку гамма-випромінювання для детектора. Передбачалося, що відстань між поглиначем і детектором у багато разів менше відстані до джерела гамма-випромінювання R і його впливом на процес прийому гамма-квантів можна знехтувати. Чи не враховувалася також енергія іонізуючого випромінювання, поглинена в повітрі. Вираз (3) характеризує рівняння гамма-локації, коли ефект поглинання гамма-випромінювання використовується для виявлення і визначення напрямку на точкові джерела гамма-випромінювання. Для визначення напрямку на точкове джерело гамма-випромінювання детектор X розташований в асиметричному поглиначі - «ракушці», товщина якого h_X лінійно змінюється в залежності від вимірюваного кута α, що характеризує напрямок на джерело випромінювання:

$$h_X = c_X + b_X \frac{\alpha}{360^0}, \tag{4}$$

де c_X – визначає мінімальну товщину поглинача для $\alpha = 0^0$, а b_X – максимальну товщину при $\alpha = 360^0$. Детектор Y розташований у центрі коаксіального циліндра-поглинача (КЦ) з постійною товщиною поглинача в діапазоні кутів 2π радіан, яка дорівнює $h_Y = c_T$. Підраховується відношення амплітуд імпульсів у режимі струму на детекторах X і Y, яке має назву коефіцієнт пропорційності

$$K_{XY} = \exp[-\mu(h_X - h_Y)] = \exp\left[-\mu\left(c_X - c_Y + b_X \frac{\alpha}{360^0}\right)\right] = f_{XY}(\alpha). \tag{5}$$

Кожне конкретне значення відношення інтенсивності гамма-випромінювання, що вимірюються відповідними детекторами, тобто коефіцієнт пропорційності, однозначно зв'язане з напрямком на джерело гамма-випромінювання. Коефіцієнти пропорційності в теорії просторових вимірювань мають назву пеленгаційних характеристик, оскільки характеризують залежності чисел від пеленгу (напрямку) на джерело гамма-випромінювання [4, С. 163]. Коефіцієнт K_{XY} – теоретична пеленгаційна характеристика пристрою $f_{XY}(\alpha)$. Реальна пеленгаційна характеристика відрізняється від теоретичної наявністю зони невизначеності поблизу переходу товщини поглинача від максимальної до мінімальної $(360-0)^0$, де здійснюється зміна коефіцієнта пропорційності K_{XY} від максимального до мінімального, має місце неоднозначність, знак похибки невідомий. Тому вимірювання напрямку на джерело випромінювання з допомогою цих детекторів здійснюється в діапазоні кутів від 5^0 до 355^0 [1, С. 36].

У сучасній науково-технічній літературі детально розглянуті питання застосування НПД в спектрометрії, радіометрії, дозиметрії. Цього не можна сказати, якщо прилад призначений для вирішення завдань координатометрії.

В випадку координатометрії до НПД пред'являються такі вимоги:

- Володіння необхідними операційними властивостями, тобто зручність роботи з ним;
- Володіння необхідними електрофізичними характеристиками:

висока ефективність реєстрації гамма-випромінювання; максимально можлива площа робочої зони; максимально можливий показник; мінімально можливий енергетичний еквівалент шуму; Такі вимоги випливають з необхідності забезпечення максимального відношення сигнал / шум при проведенні вимірювань;

- Оперативність вимірювань;
- Прийнятні масо-габаритні характеристики.

Матеріалом для виготовлення напівпровідникових детекторів (ППД), які реєструють гамма-випромінювання, є германій, кремній і двокомпонентні напівпровідникові кристали, які мають широку заборонену зону. З урахуванням першого вимоги зі списку можливих матеріалів можна виключити Ge, оскільки наведеними характеристиками він володіє при температурі рідкого азоту, а розроблювальний пристрій має бути переносним, компактним. Крім GaAs, CdTe, HgI2 інші щірокозонние напівпровідникові матеріали, будучи перспективними для виготовлення охолоджуваних ППД, поступаються перерахованим в ступеня опрацювання і в порівнянні з ними знаходяться в початковій стадії практичного використання. Тому перелік можливих матеріалів

далі обмежимо рамками 4-х матеріалів: кремній (Si), дііодід ртуті (HgI2), арсенід галію (GaAs), телурид кадмію (CdTe).

Висока ефективність реєстрації гамма-випромінювання передбачає виконання умов;

- Трек частки повинен повністю укладатися в чутливої області ППД

- Заряди, які генеруються гама - квантами, повинні як можна повніше збиратися на електродах.

Завдяки цім умовам забезпечується пропорційність сигналу на виході детектору енергії частинок і їх інтенсивності.

При реєстрації гамма - квантів повне число нерівноважних носіїв N в треку дорівнює:

$$N = E / \varepsilon \text{ ,} \tag{12}$$

де E ─ енергія, яка втрачається частинкою в обсязі ППД; ε ─ енергія, необхідна для освіти однієї електрон-діркової пари.

Коли на поверхню детектора з робочою поверхнею S в одиницю часу t впливає випромінювання з інтенсивністю I_0, число поглинених квантів дорівнюватиме

$$N_{П} = I_0\left[1 - \exp(-\mu d_{П})\right]St \text{ ,} \tag{13}$$

$d_{П}$ ─ товщина шару поглинання.

Щоб мати хороший показник $N_{П}$, матеріал ППД повинен мати високий атомний номер Z і щільність ρ (г / см3), так як перетин фотоефекту збільшується як Zn (n = 3,5─5) і пропорційно щільності. Кремній (Si) за цими параметрами поступається бінарним матеріалами. Наприклад, у порівнянні з телуриду кадмію (CdTe) товщина шару поглинання кремнію повинна бути більше в 2,6 рази.

Як випливає з (12), для збільшення ефективності реєстрації ε повинна бути мінімальною. Кремній має найменшу середньої енергією ε, що витрачається на народження однієї електрон-діркової пари, яка дорівнює 3.61 еВ.

Порівняльний теоретичний і експериментальний аналіз ефективності ППД на GaAs і Si проведено в [5]. Встановлено, що приладові характеристики Si ППД (еквівалентний енергетичний шум ENC 2 кеВ, енергетичне дозвіл FWHM 10 кеВ) при кімнатній температурі вище, ніж у кращого зразка GaAs ППД з «чистого» шару арсеніду галію, що має ENC 8,3 кеВ, FWHM 15, 1 кеВ. При таких рівнях шумів виграш у відношенні сигнал / шум при використанні Si ППД залишає 6 дБ, що призводить до збільшення дальності виявлення джерела гамма - випромінювання в 2 рази.

Кремній, як речовина для ППД, за показниками ефективності реєстрації випромінювання, збору заряду (ССЕ) і енергетичним еквівалентом вхідного шуму $\sigma_E(эВ)$ перевищує бінарні матеріали, і кращий при вирішенні задач координатометріі в разі низької енергії та інтенсивності реєстрованих гамма - квантів. У разі реєстрації гамма - квантів високих енергій і інтенсивності ситуація змінюється, так як кремній менш радіаційностійких в порівнянні з бінарними матеріалами.

В роботі [6] проведено експериментальне дослідження CdTe і CdZnTe ППД для вимірювання рентгенівського і гамма-випромінювання. Показано, що струми витоку, що визначають рівень шуму, для CdZnTe ППД значно менше в порівнянні з CdTe ППД (для електронів ─ 26 нА проти 95 нА; для дірок ─ 2,3 нА проти 8.6 нА). Це означає, що по еквівалентному енергетичному шуму ENC CdZnTe ППД має перевагу перед CdTe ППД.

Детальне визначення радіаційного ресурсу ППД на основі широкозонних CdTe (CdZnTe) ППД проведено в [7]. Експериментально встановлено, що деградація CdTe при гамма-опроміненні починається з доз менших, ніж для CdZnTe. Величини граничних доз поглинання - випромінювання, при якій характеристики ППД неприйнятно змінюються, складають приблизно 200 кГр і 800 кГр (8 × 107 рад) для дозиметрів на основі CdTe і CdZnTe відповідно.

З цієї причини вибір необхідно робити з двох матеріалів: GaAs і CdZnTe.

За товщиною епітаксійних шарів GaAs ППД поступаються CdZnTe ППД і використовуються, найчастіше, в спектрометрії гамма-випромінювання низьких енергій. Крім невеликої товщини, недоліком ППД на основі GaAs є також мала робоча поверхня, а це важливий

фактор для координатометрії. Є ще одна причина, по якій перевагу слід віддати CdZnTe. GaAs в більшій мірі схильний до деградації від частини випромінювання, зумовленої нейтронами і протонами [7, С. 59]. Це цілком можливо при координатометріі, наприклад, ядерних вибухів.

Наведені дані дозволяють стверджувати, що в разі створення інформаційно-вимірювальних систем координатометрії джерел гамма-випромінювання високих енергій та інтенсивностей в якості ППД слід вибирати CdZnTe детектори. Тим паче, в ННЦ ХФТІ розроблені CdZnTe ППД для реєстрації гамма-випромінювання, енергетичний діапазон яких становить від 20 до 3000 кеВ, діапазон робочих температур ─ від-40 до +50 0С, енергетичної дозвіл ─ порядку декількох відсотків [8].

ЛІТЕРАТУРА

1. Григор’єв О. М. Дослідження способів локації джерел проникаючих ядерних випромінювань / О. М. Григорьев, Л. Б. Беденко, О. В. Сакун, С. М. Коваленко // Системи озброєння і військова техніка, 2006. – випуск 4(8). − С. 33−37.
2. Спосіб визначення напрямку на імпульсні джерела гамма-випромінювання : Патент UA 108262, G21J 5/00. / О. М. Григор'єв, З. В. Білик, О. В. Сакун, В. В. Марущенко – u 2013 05335 ; опубліковано 10.04.2015. Бюл. № 7.
3. Пристрій для визначення напрямку в просторі на точкові постійні та імпульсні джерела гамма-випромінювання : Патент UA 85493 U, G01T 1/16, G01T 1/167, G01T 1/169, G21J 5/00 / О. М. Григор'єв, З. В. Білик, О. В. Сакун, В. В. Марущенко – u 2013 05331 ; опубліковано 25.11.2013. Бюл. №22.
4. Сайбель А. Г. Основы радиолокации / А. Г. Сайбель // − М.: «Советское радио», 1961. – 384 с.
5. Беспалов В. А. Электрофизические свойства GaAs слоев и особенности характеристик детекторов частиц высоких энергий на их основе / В. А. Беспалов, А. В. Воронцов, А. А. Горбацевич и др.// [Электронный ресурс]. ─ Режим доступа: http://elibrary.lt/resursai/Uzsienio%20leidiniai/ioffe/ztf/2004/03/ztf_t74v03_06.pdf
6. Прохорец И. М. Моделирование и экспериментальное исследование CdTe (CdZnTe) детекторов измерения рентгеновского и -излучений [Электронный ресурс] ─ Режим доступа: https://cyberleninka.ru/article/n/modelirovanie-i-eksperimentalnoe-issledovanie-cdte-cdznte-detektorov-dlya-izmereniya-rentgenovskogo-i-gamma-izlucheniy/viewer
7. Давыдов Л. Н. Радиационная стойкость полупроводниковых детекторов корпускулярного и гамма-излучения / Л. Н. Давыдов, А. А. Захарченко, Л. В. Кутний и др.// ─ Харків: Вісник Харківського університету, № 657, серія фізична «Ядра, частинки, поля», вип.1 (26), 2005. ─ С. 3─21.
8. Кутний В. Е. Разработка дозиметрических и спектрометрических блоков регистрациигамма-излученияна основе полупроводниковых соединений CdTe (CdZnTe) для АЭС Украины / В. Е. Кутний. А. В. Рыбка, Д. В. Кутний и др. // ─ Вопросы атомной науки и техники. Серия; «Физика радиационных повреждений и радиационное материаловедение», 2004. ─ № 3. ─ С. 96─100.

АНДРЄЄВ Фелікс Михайлович – д. т. н., професор кафедри електроніки та управляючих систем Харківського національного університету імені В.Н. Каразіна.

Наукові інтереси:

– *інформаційно-вимірювальні системи, радіолокація.*

ОСИПЧУК Андрій Володимирович – викладач кафедри електроніки та управляючих систем Харківського національного університету імені В.Н. Каразіна.

Наукові інтереси:

– *інформаційно-вимірювальні системи, електроніка.*

СТЄРВОЄДОВ Микола Григорович – к. т. н., завідувач кафедри електроніки та управляючих систем Харківського національного університету імені В.Н. Каразіна.

Наукові інтереси:

– *кіберфізичні системи, електроніка.*

УДК 004.94 **СЕКЦІЯ 2**

АНЖУРОВ В.Є., ТОЛСТОЛУЗЬКА О.Г.

МОДЕЛЬ КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ АНАЛІЗУ ІНДЕКСІВ АКЦІЙ НА ФОНДОВОМУ РИНКУ

Вступ

Кожна людина шукає шляхи зберегти/збільшений її зароблений тим чи іншим чином фінансовий капітал. Саме тому сьогодні тема фондового ринку та його цінних паперів має актуальний характер, тому що це є один з найстаріших засобів інвестування. Фондовий ринок як явище з'явився в Америці в 1970 році, що є роком виникнення першої фондової біржі. Компанії зрозуміли, що капітал можна надбати не лише шляхом продажу власної продукції, а й за рахунок вкладів звичайних людей в бізнес компаній. Таким чином, ці люди ставали співвласниками мізерної частини бізнесу та мали відповідний заробіток з цього

Як працює біржа

Як було сказано в розділі Вступ, акціонери інвестують власні кошти в бізнес тієї чи іншої компанії, купуючи один з видів цінних паперів:

- Акції
- Облігації

Акція – цінний папір, що робить її власника співвласником бізнесу, до якого цей папір належить. "Співвласник Facebook", надто звучно. На ділі ви стаєте співвласником тієї частини, яку придбали. Наприклад, компанія випустила 2 млн. акцій і ви придбали одну з них. Це значить, що ви можете претендувати на 1/2000000 частину з того капіталу, який компанія виділяю зі свого заробітку на своїх акціонерів (інші частини вона може реінвестувати, направити на саморозвиток чи купити іншу компанію). Ця 1/2000000 частина має назву «дивіденд».

Також, інвестор може розраховувати на приріст ціни акції, яку має. Це може статися в разі зросту ціни компанії в цілому за рахунок її розвитку на ринкові. Наприклад, акціонер придбав 1 акцію вартістю 100$. Коли компанія дорожчає на 2%, дорожчає і акція, а значить, тепер вона коштує 102$. Продавши її, акціонер заробляє різницю у 2$.

Облігація – цінний папір, що видає компанія взамін на позикові гроші. Це працює, як кредит у звичайному житті, але тут в позик даєте ви. На це компанія дає вам розписку, в якій вказано коли та з яким відсотком будуть повернені гроші.

Маючі ці 2 засоби, інвестори примножують свій капітал, вдало працюючи з ринком та аналізуючи поточну ситуацію кожної з кампаній.

Модель системи аналізу індексів на фондовому ринку

Існує багато методів виявити чи підходить компанія під визначення «надійна». Це значить, що вклавши в неї гроші, інвестор хочу мати відчуття того, що ця компанія не зникне завтра через високу конкуренцію. Одним з таких методів є метод аналізу індексів компанії. Нижче наведені основні індекси та їх визначення:

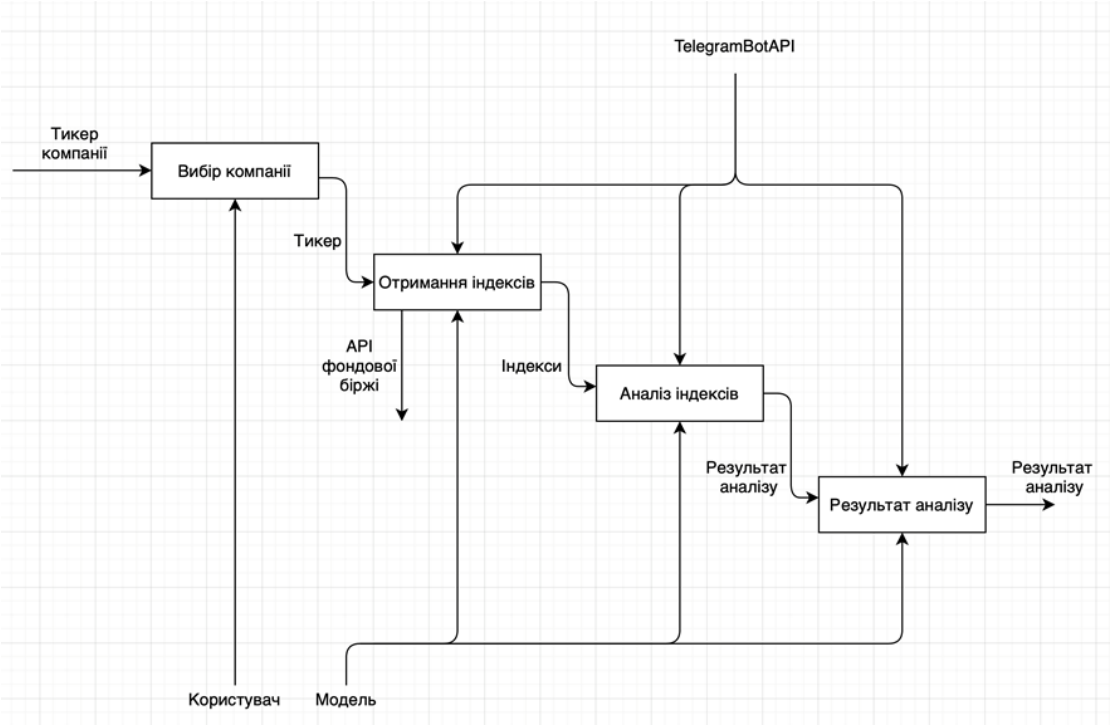
- P/E – відношення ціни акції до прибутку компанії на акцію. Попит на акцію збільшує її ціну. Це індекс показує переоцінені та недооцінені акції. Також, маючі цей індекс, можна оцінити за скільки років одна акція окупається
- P/B – відношення ціни акції до капіталу компанії на акцію.
- P/S – відношення ціни компанії до її прибутку. Цей індекс показує в скільки разів менше чи більше компанія отримує в порівнянні з тим, скільки коштує. Очевидно, чи менше цей показник, тим компанія більш сильна
- P/FSF – ціна акції до вільних грошей на акцію. Цей індекс показую наскільки компанія сильно реінвестує прибуток в саморозвиток, а отже, можна оцінити перспективність такої компанії

© АНЖУРОВ В.Є., ТОЛСТОЛУЗЬКА О.Г., 2021

Модель системи аналізу індексів на фондовому ринку

Для роботи з моделлю системи аналізу був обраний додаток Telegram, що дозволяє взаємодіяти з моделлю через телефон та інші види пристроїв. Цей додаток є популярнішим за всі інші на пострадянському просторі, а отже, є ідеальним варіантом. Користувач спілкується з моделлю шляхом введення повідомлень в телефоні та отримує в той самий чат відповіді моделі.

На мал.1 наведена IDEF0 діаграма моделі системи аналізу:



Мал. 1 IDEF0 діаграма моделі комп'ютеризованої системи аналізу індексів акцій

Результатом роботи моделі є рекомендації щодо роботи з обраною компанією. Аналізуючи індекси компанії, отримані з біржі, можна робити такі висновки:

- компанія може бути вигідною для роботи на короткому періоді (менше 1 року). Цей варіант більше підходить для трейдерів, ніж для інвесторів
- компанія вигідна для роботи на середньому періоді (1-10 років)
- компанія підходить для роботи на довгому терміні (більше 10 років)

Дані результати є лише порадою та результатом сухого аналізу індексів без врахування подій у реальному світі, що звісно впливають на стан фондового ринка.

ЛІТЕРАТУРА

1. Ковальов В. М. Сучасні методології опису бізнес-процесів. Методологія IDEF0: Журнал" Консультант директора", № 12, Червень, 2004 р.
2. Бенджамін Грехам. Розумний інвестор. Москва, 2002 р.
3. Telegram Bot API: [Електронний ресурс] 2020, URL: https://core.telegram.org/bots/api

АНЖУРОВ Валентин Євгенович – студент; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: triathlete433@gmail.com; ORCID: 0000-0002-5486-781X.

Наукові інтереси: *комп'ютерне програмування*.

ТОЛСТОЛУЗЬКА Олена Геннадіївна – професор кафедри теоретичної та прикладної системотехніки, д. т. н., с. н. с.; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: elena.tolstoluzka@karazin.ua; ORCID: 0000-0001-2741-180.

Наукові інтереси: *технології автоматичного проектування паралельних програм*.

УДК 004.031.42 **СЕКЦІЯ 2**

АСЄЄВ Б.А, ПОПОВА А.О, ГУЩІН І.В.

АНАЛІЗ КОМП'ЮТЕРНИХ СИСТЕМ ТА ЕЛЕКТРОННИХ СКЛАДОВИХ ДЛЯ ВІДСТЕЖУВАННЯ ПАСАЖИРСЬКИХ ПЕРЕВЕЗЕНЬ

Вступ до засобів відстежування

У сучасному житті існує безліч прикладів застосування засобів відстежування починаючи з маячків для ошийників домашніх улюбленців і закінчуючи системами слідкування за космічними ракетами та супутниками. Якщо поглянемо у недалеке минуле та прослідкуємо за розвитком засобів відстежування, то можемо побачити, що все починалось зі звичайних пристроїв, званих трекерами, що накопичували у собі інформацію про власне місцезнаходження. Вони могли віддавати її після того, як транспорт чи інший об'єкт відстежування прибуде до кінцевої точки, тобто, постфактом. У таких системах було багато проблем, тим не менш це був перший крок до налагодження відстежування місцезнаходження. Потім люди почали використовувати мобільний зв'язок та такі технології як SMS і CSD. Даний спосіб хоч і працював повільно на ті часи, але це було набагато краще ніж отримувати дані офлайн засобом, це навіть можна назвати першим кроком до відстежування у режимі реального часу. Після цього людство зробило ще один великий крок у даній галузі та відкрило технології GPRS та EV-DO. Був зменшений час виділений на відправлення даних та збільшена ефективність систем відстежування загалом. Потім було прийняте рішення з'єднати попередню систему з web-орієнтованими технологіями, нехтуючи тим, що системи почали проектувати централізованими і це почало приносити великі втрати при збої пристроїв, так як усе обладнання стало знаходитись в одному місці. Та останнім кроком до покращення систем відстежування було запровадження ГЛОНАСС та GPS(Global Positioning System). Наразі системи можна вважати найбільш удосконаленими через використання хмарних технологій, що зменшують час на обробку та відправку даних та витрати на збереження.

Аналіз та порівняння способів визначення місцезнаходження

Існує два основних способи відстежування місцезнаходження: за допомогою стільникової вишки (LBS/GSM) та за допомогою супутника (GPS). Перший варіант визначає локацію за допомогою тріангуляції. Тобто, для коректної роботи нам необхідно мати в наявності три вишки стільникового зв'язку.

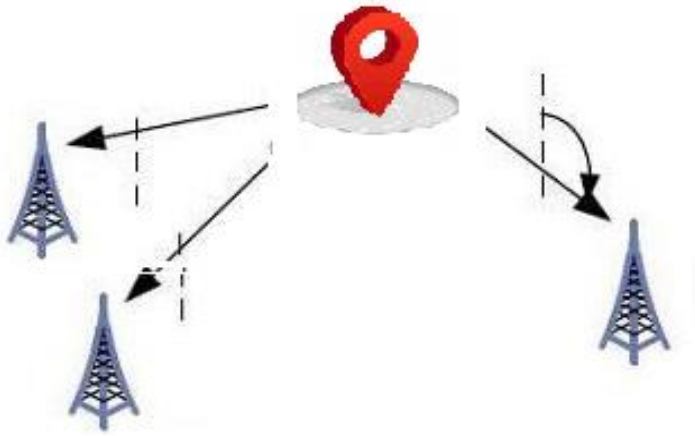


Рис. 1 Модель роботи LBS/GSM системи

© АСЄЄВ Б.А, ПОПОВА А.О, ГУЩІН І.В., 2021

Цей спосіб має досить невелику точність геопозиціонування об'єкту і тому це є великим мінусом через те, що це працює тільки у зоні покриття стільникового зв'язку, а у таких місцях як метро, підвали чи міжміські траси, де сигнал з вишками дуже нечіткий або його немає зовсім, відстежити локацію корректно майже неможливо. Щодо точності його вимірювання, у великих містах вона досягає до 2.7 км, а вже на відкритій місцевості цей показник змінюється на 22 кілометри[1].

Другий спосіб пов'язаний, як було вже описано вище, з супутником та теж з методом тріангуляції, але у цьому випадку вимірюється відстань до навігаційних супутників. Для обчислення географічних координат треба мати кількість супутників від трьох, а для того, щоб обчислити ще й висоту над поверхнею від чотирьох.

Рис. 2 Модель роботи GPS системи

Головна логіка цієї системи зав'язана на GPS-модулі, тобто невеликий пристрій, який можна встановити в авто або інший транспортний засіб. Цей спосіб має більш високу точність у порівнянні з попереднім, вона лише залежить від кількості супутників, що знаходяться над об'єктом, крім цього перешкодами для поганого з'єднання може бути як невелика хмарність, так і звичайне приміщення з бетонними стінами та металевими кріпленнями[2].

Виходячи з цього можна зробити узагальнення використання цих двох основних способів визначення місцезнаходження – для відстежування невеликих об'єктів у зоні покриття стільникового зв'язку (наприклад міський транспорт) найчастіше використовується GSM/LBS, але за умови, що точність не грає велику роль для кінцевого користувача. При відстежуванні наприклад кораблів чи поїздів використовують GPS, бо це є більш надійним та точним способом визначенням геолокації[3].

Комп'ютерні системи та електронні складові

Невід'ємною частиною системи GPS є невеликий за своїми розмірами GPS-модуль. Він має певний набір мікросхем, які також включають у себе процесор, що відповідає за всі розрахунки отриманих даних. З поширених наборів мікросхем GPS можна виділити такі як u-Blox, SiRF та SkyTraq. Прикладом GPS-модулів є такі моделі як LR9101LP/LR9102, LR9548S, LR9552 від компанії Leadtek, NEO-4S, LEA-4H, LEA-5S від u-Blox, eMD3500F від eRide та інші.

Такі модулі ми можемо зустріти у більшості звичайних GPS-навігаторів які наявні у сучасних машинах чи встановлюються додатково і виглядають як невеликий пристрій. Функціонал такого модуля досить великий, він включає в себе отримання широти, довготи та висоти над рівнем моря, швидкості руху, дати та часу UTC, та кількості з'єднаних з модулем супутників.

Рис. 3 Приклад навігаційного чіпу виробництва компанії Leadtek

Такі модулі мають такі відмінності як:
- процесор;
- чутливість до зміни локації;
- час визначення локації;
- точність передаваємих координат;
- вид інтерфейсу та протоколу взаємодії;
- діапазон робочих температур;
- робоча напруга живлення.

До прикладу можна навести приймач NEO-6M. Він використовує набір мікросхем u-Blox, тому підтримує формати даних NMEA та UBX. Також чіпсети мають безліч послідовних інтерфейсів для передачі даних, часто зустрічаються такі як UART, I2C, SPI та USB. Наприклад зазначений раніше GPS NEO-6M підтримує інтерфейси UART, I2C або SPI для передачі даних. Загалом ці модулі виконують функцію приймача сигналу від супутників таких компанії як NAVSTAR, GALILEO та ГЛОНАСС [4,5]. Отриманий з модулів сигнал використовують у різноманітних комп'ютерних системах для роботи з маршрутами та аналізом вихідних даних.

Найвідомішою з великих комп'ютерних навігаційних систем на даний момент є електронно-картографічна навігаційно-інформаційна система ECDIS (ЕКНІС, англ. Electronic Chart Display and Information System, ECDIS)[6]. Вона використовується серед моряків та забезпечує безпеку судноводіння і може служити альтернативою традиційним паперовим картам. Система визначає дані за допомогою глобальних супутникових навігаційних систем, таких як GPS і навігаційних параметрів, таких як показники радарів, відносна та абсолютна швидкість судна, швидкість та напрямок вітру і параметри магнітного і гіроскопічного компасів. Вона служить для отримання достовірної інформації про картографічну і навігаційну обстановку[7].

Рис. 4 Приклад системи ECDIS на кораблі

Система постійно підключена до двох незалежних GPS-приймачів і вона продовжить працювати, навіть якщо один з них стане несправним. Але як було розглянуто у попередньому розділі, сигнали GPS можуть бути помилковими. Цю проблему вирішує суднова радіолокаційна станція (радар) яка сполучена з ECDIS. Звіривши розташування судна по радару і електронних картам, вона постійно перевіряє точність GPS-приймача. Також для коректування геолокації у даної системи присутня функція зміни джерел позиціонування, тобто можна вибрати нерухому ціль, наприклад маленький острів чи маяк і система буде орієнтуватись на цей об'єкт[8].

На даний момент людство досить далеко продвинулось з часів коли для навігації використовувався компас а для визначення координат звичайна географічна карта до супутникової навігації з високою точністю визначення геолокації.

ЛІТЕРАТУРА

1. https://www.ixbt.com/mobile/gsm-nets.html
2. https://gpsmarker.ru/info/blog/lbs-i-a-gps-v-chem-raznitsa.html
3. http://www.tk-chel.ru/download/doc/BasicGPS.pdf
4. https://gssc.esa.int/navipedia/index.php/GPS_Receivers
5. https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_(GPS.G6-HW-09005).pdf
6. https://www.marineinsight.com/marine-navigation/what-is-electronic-chart-display-and-information-system-ecdis/
7. https://www.navico-commercial.com/simradcommercial/ecdis/ecdis-e5024-system/
8. https://cirspb.ru/blog/info-navigation/ecdis-type/

ПОПОВА Анастасія Олександрівна – студентка кафедри моделювання систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: xa11867792@student.karazin.ua; ORCID: 0000-0003-1145-5695.

Наукові інтереси:

– *моделювання складних комп'ютерних систем,*
– *аналіз комп'ютерних систем*

АСЄЄВ Богдан Андрійович – студент кафедри теоретичної та прикладної схемотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: asieiev2021ku21@student.karazin.ua; ORCID: 0000-0001-5749-0598.

Наукові інтереси:

– *моделювання складних комп'ютерних систем,*
– *аналіз комп'ютерних систем*

ГУЩІН Іван Валерійович – старший викладач кафедри штучного інтелекту і програмного забезпечення; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: gushchin.iv@gmail.com; ORCID: 0000-0002-1917-716X.

Наукові інтереси:

– *комп'ютерне моделювання фізичних процесів;*
– *розробка програмного забезпечення в області аналіза даних.*

УДК 004.62 **СЕКЦІЯ 2**

БАКУМЕНКО Н.С, ДОНЕЦЬ В.В, ШЕВЧЕНКО Д.О, ОДИНЕЦЬ О.О, УГРЮМОВ М.Л.

МЕТОДИ КЛАСТЕРИЗАЦІЇ ДАНИХ НА ОСНОВІ ІНФОРМАЦІЙНИХ КРИТЕРІЇВ

Вступ

Кластеризація в галузі аналізу даних і машинного навчання є одним з ефективних методів опрацювання експериментальних даних в різноманітних галузях науки і техніки (серед них можна виділити медичні спостереження, спостереження за економічною діяльністю, класифікації процесів та інше) [1, 2]. Наприклад, в маркетингу клієнтів можна згрупувати в кластери на основі їх потреб, вибору бренду, психологічних профілів або інших параметрів. Вагомою перевагою кластерного аналізу – є можливість проводити розбивку елементів кластера на кілька об'єктів з огляду на вплив кожного з змінних описуваної вибірки [2]. Крім цього метод є відносно не чутливий до довільної природи вхідних даних.

Огляд базового методу нечіткої кластеризації c-середніх

Метод нечіткої кластеризації c-середніх дозволяє виробляти нечітку класифікацію (розбивку) елементів вихідної множини потужністю N на задане число підмножин – «класів» [1]. Метод нечіткої кластеризації c-середніх відрізняється від його попередника методу k-середніх методом визначення приналежності об'єкта до можливого класу.

Алгоритм c-середніх [3] включає послідовність операцій:

1. На першому етапі задаються випадковим чином k центрів кластерів c_j, j = 1, ..., k;
2. Далі розраховується матриця приналежності елементів до кластерів w_{ij}. На цьому етапі є різні варіації способу визначення відстані між можливим центром і елементом підмножини. При нормального розподілу даних, матриці приналежності буде виглядати наступним чином:

$$w_{ij} = \frac{\aleph(d(x_i, c_j) \mid \mu = 0, \sigma_j)}{\sum_{i=1}^{P_i} \aleph(d(x_i, c_j) \mid \mu = 0, c_j)}, \quad (1)$$

де x_i – i-й елемент множини, i = 1, ..., P_j, c_j – центр j-ого кластера, $d(x_i, c_j)$ – відстань між точками x_i, c_j, $\aleph(d(x_i, c_j) \mid \mu = 0, \sigma_j)$ – щільність ймовірності нормального розподілу в точці $d(x_i, c_j)$.

3. Далі необхідно перемістити центри кластерів, враховуючи матрицю приналежності:

$$c_j \leftarrow \frac{\sum_{i=1}^{P_j} w_{ij} x_i}{\sum_{i=1}^{P_j} w_{ij}}, \quad (2)$$

4. Потім необхідно розрахувати функцію втрат (на цьому етапі можна скористатися принципом максимальної правдоподібності). У разі нормального розподілу функція втрат дорівнює:

$$Loss_function = \sum_{j=1}^{k} \sum_{i=1}^{P_j} d(x_i c_j)^2 w_{ij} \rightarrow \min, \quad (3)$$

5. У разі якщо значення функції втрат не зменшилось, треба перейти до пункту 2. Так само тут може бути функція зупинки, відповідно до якої вихід може наступити раніше, при досягненні якоїсь умови (наприклад, досягнення певної кількості ітерацій).

Метод нечіткої кластеризації c-середніх так само має ряд істотних недоліків обмежують його застосування, наприклад якщо цільовий кластер має більш складну форму ніж просто m-мірна сфера, або у разі перетину кластерів і високою шуму даних, то метод нечіткої кластеризації не здатний якісно визначати приналежність елементів до того, чи іншого кластеру [2, 4].

Постановка проблеми

Через те що в більшості реальних наборів даних присутній широкий розкид можливих значень змінних стану елементів множини, що визначає складну m-мірну форму можливого кластера [4], це призводить до неможливості визначити приналежність елемента, так само

© БАКУМЕНКО Н.С, ДОНЕЦЬ В.В, ШЕВЧЕНКО Д.О, ОДИНЕЦЬ О.О, УГРЮМОВ М.Л., 2021

виникають складнощі з великою кількістю кластерів і великою кількістю змінних стану елементів. Тому виникає необхідність модифікувати існуючий метод c-середніх [2, 3]. Шляхів цих модифікацій кілька, одним з найбільш очевидних є модифікація методу оцінки відстані між центром кластера і передбачуваним елементом (або оцінки приналежності даного елемента кластеру).

Огляд існуючих рішень

У роботах [1, 5] використовувалася модифікація функції оцінки приналежності елемента кластеру і модифікація розрахунку матриці приналежності. У відповідно до яких використовувалося припущення про розподіл Коші, тому для визначення приналежності використовувалися відстані Махаланобіса:

$$d^2(x_i, c_j) = MD^2(x_i, c_j) = \frac{1}{A}(x_i - c_j)T\hat{\sum}_j^{-1}(x_i - c_j), \quad (4)$$

де A – розмірність простору стану, $\hat{\sum} = \sum + \lambda E$ – регуляризована коваріаційна матриця, λ – постійна, що більше нуля.

Тоді враховуючи припущення про розподіл Коші, матриця приналежності розраховується за наступним виразом [1]:

$$w_{ij} = \frac{\rho(x_i, c_j)}{\sum_{j=1}^{P_j}\rho(x_i, c_j)}, \text{ де } \rho(x_i, c_j) = \frac{1}{\pi\eta[1 + \frac{MD^2(x_i, c_j)}{\eta_2}]}. \quad (5)$$

Так само проблему з кластеризації m-мірних еліпсів усувається за допомогою схожих на представлені вище методів Mixture models і Gaussian mixture models, які успішно використовувалася у визначенні стану справності / несправності пристрою [5]. Або завдяки використанню інформації про похідну у визначенні відстані [6].

Проведені експерименти показали поліпшення якості кластеризації, але так само все ще показувало залежність від характеру розподілу даних вибірки, тобто новий метод не дозволяв в достатній мірі змоделювати залежності в наборі даних [1, 5]. А використання похідної в модифікованій відстані може мінімізувати суму квадратних відстаней всередині кластерів [6].

Так само була запропонована ідея обліку характеру розподілу даних у вибірках з урахуванням відносної ентропії в нечіткої кластеризації c-середніх, відповідно до якої був змінений алгоритм обчислення кластерів [7, 8], але була залишена колишня міра обчислення приналежності (відстані) між елементами і центрами у формі евклідової відстані, що не враховувало ентропію даних [7].

Запропоноване рішення

Для усунення проблем базового методу c-середніх, а так само його модифікацій Mixture models , Gaussian mixture models та [1] була запропонована ідея, що була надихнута спробою враховувати ентропію даних [7], а також оцінки інформаційної відстані Кульбака-Лейблера [9].

Відстань Кульбака-Лейблера є несиметричною мірою інформаційної розбіжності двох розподілів ймовірності, метод добре зарекомендував себе в прикладній статистиці, механікі рідин і машинному навчанні. І на відміну від розглянутих модифікацій відстань Кульбака-Лейблера дозволить оцінювати міру інформаційної відмінності [9].

Перед розглядом запропонованого методу, введемо деякі позначення: $X = \{x_m\}$ – вектор змінних стану, який характеризує запис в множині $m = 1,..., M$, M – розмірність вектору змінних стану; $F = \{f_i\}$ – вектор цільових функцій, $i = 1,..., I$, де I – розмірність вектору цільових функцій. Візьмемо $M\alpha[f_i], M\alpha[x_m]$ – математичне очікування f_i, x_m, тоді D_{f_i}, D_{x_m} – дисперсія f_i, x_m та $\sigma_{f_i}, \sigma_{x_m}$ – середньоквадратичне відхилення f_i, x_m.

Дисперсія та середньоквадратичне відхилення f_i при $x_m = \text{var}$ визначається наступними виразами:

$$D_{f_i|x_m} = \text{var}[M_\alpha[f_i[X_{-m}]]], \forall n = 1,..., M, n \neq m, xn = const, \quad (6)$$

$$\sigma_{f_i|x_m} = sqrt[D_{f_i|x_m}]. \quad (7)$$

Для оцінки інформативності змінних стану застосовувалися слідуючи формули. Використовуючи формулу (6), отримаємо коефіцієнт інформативності:

$$\beta_{f_i} = \frac{D_{f_i|x_m}}{E_{f_i}}, \quad (8)$$

де E_{f_i} – енергія сигналу. Відношення сигнал шум, маємо з виразу (7) що є не що іншим як коефіцієнтом впливу:

$$\varphi_{im} = SNR_{f_i|x_m} = \frac{\sigma_{f_i|x_m}}{\sigma_{x_m}}. \quad (9)$$

Відносна ентропія Кульбака-Лейблера визначається виразом [9]:

$$D_{KL}(f_i, X) = \sum_{m=1}^{M} \rho(x_m \mid f_i) \log 2[\frac{\rho(x_m \mid f_i)}{\rho(x_m)}]. \quad (10)$$

Взаємна інформація визначається виразом:

$$H_{im} = \frac{1}{2}\log_2[(SNR_{f_i|x_m})^2] = \frac{1}{2}\log_2[\beta_{im}\frac{E_{f_i}}{D_{x_m}}]. \quad (11)$$

Далі для оцінки якості кластеризації даних замінено формулу (3) на вираз для оцінки взаємної інформації, що буде функцією втрат в запропонованій модифікації методу с-середніх:

$$H(X, M) = -\frac{1}{J\sum_{j=1}^{J} P_j}\sum_{j=1}^{J}\{P(M_j^{(t+1)})\sum_{i=1}^{P_j} D_{KL}(x_i, M_j^{(t+1)})\} \to \min, \quad (12)$$

де M_j змінних стану приналежних j–тому кластеру.

Висновки

Впровадження в метод нечіткої кластеризації c-means інформаційних критеріїв дозволить усунути недоліки розроблених раніше методів, а саме адаптацію методу до широкого розкиду відхилень змінних стану і кластерам зі складною m-мірну формою. Також цій метод дозволить врахувати інформативність змінних стану елементів, що підвищить якість кластеризації.

ЛІТЕРАТУРА

1. Mykhaylo Ugryumov, Nina Bakumenko, Viktoriia Strilets. Application of the C-Means Fuzzy Clustering Method for the Patient's State Recognition Problems in the Medicine Monitoring Systems. *Conference: Proceedings of the 3rd International Conference on Computational Linguistics and Intelligent Systems (COLINS-2019). Volume I: Main Conference Kharkiv, Ukraine*, 10p., April 2019. URL: https://www.researchgate.net/publication/338819685 (Last accessed 15.02.2021).
2. Salar Askari. Fuzzy C-Means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: Review and development. Expert Systems with Applications, volume 165, August 2020. DOI: 10.1016/j.eswa.2020.113856. URL: https://doi.org/10.1016/j.eswa.2020.113856 (Last accessed: 27.02.2021).
3. Md. Abu Bakr Siddique, Rezoana Bente Arif, Mohammad Mahmudur Rahman Khan, Zahidun Ashrafi. Implementation of Fuzzy C-Means and Possibilistic C-Means Clustering Algorithms, Cluster Tendency Analysis and Cluster Validation. *ArXiv e-Journal* (ISSN 2331-8422). May 2019. URL: https://arxiv.org/abs/1809.08417v3 (Last accessed 27.02.2021).
4. Roland Winkler, Frank Klawonn, Rudolf Kruse. Problems of Fuzzy c-Means Clustering and Similar Algorithms with High Dimensional Data Sets. *Conference: 34th Annual Conference GfKl 2010, 3rd German-Japanese Workshop.* July 2010. DOI: 10.1007/978-3-642-24466-7-9. URL: https://www.researchgate.net/publication/225005676 (Last accessed: 20.02.2021).

5. Nagdev Amruthnath, Tarun Gupta. Fault Class Prediction in Unsupervised Learning using Model-Based Clustering Approach. February 2018. DOI: 10.13140/RG.2.2.22085.14563. URL: https://www.researchgate.net/publication/322900854 (Last accessed: 24.02.2021).
6. Yinfeng Menga, Jiye Liangb, Fuyuan Caob, Yijun He. A new distance with derivative information for functional k-means clustering algorithm. *Information Sciences, volumes 463–464,* October 2018, Pages 166-185. URL: https://doi.org/10.1016/j.ins.2018.06.035 (Last accessed: 26.02.2021).
7. M.Zarinbala, M.H.Fazel Zarandia, I.B.Turksen. Relative entropy fuzzy c-means clustering. *Information Sciences, volume 260,* 1 March 2014, Pages 74-97. URL: https://doi.org/10.1016/j.ins.2013.11.004 (Last accessed: 26.02.2021).
8. Frank Nielsen. On the Jensen–Shannon Symmetrization of Distances Relying on Abstract Means. *Sony Computer Science Laboratories, Takanawa Muse Bldg., 3-14-13, Higashigotanda, Shinagawa-ku, Tokyo 141-0022, Japan, Entropy 2019, 21(5), 485.* May 2019. URL: https://doi.org/10.3390/e21050485 (Last accessed 27.02.2021).
9. Kullback, S., Leibler, R.A. On information and sufficiency. *Annals of Mathematical Statistics. 22 (1): 79–86,* 1951. DOI: 10.1214/aoms/1177729694.

БАКУМЕНКО Ніна Станіславівна – к.т.н., доцент кафедри ТПС; факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: n.bakumenko@karazin.ua; ORCID: 0000-0003-3496-7167.

Наукові інтереси:

– методи машинного навчання в багатокритеріальних задач робастної оптимального проектування та інтелектуального діагностування систем (ROD & IDS) в умовах невизначеності.

ДОНЕЦЬ Володимир Віталійович – аспірант 1-ого курсу факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: vol.donets@gmail.com; ORCID: 0000-0002-5963-9998.

Наукові інтереси:

– методи машинного навчання в багатокритеріальних задач робастної оптимального проектування та інтелектуального діагностування систем (ROD & IDS) в умовах невизначеності **.**

ШЕВЧЕНКО Дмитро Олександрович – студент групи КІ-51 факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: dimyich24@gmail.com; ORCID: 0000-0002-7897-250X.

Наукові інтереси:

– методи машинного навчання в багатокритеріальних задач робастної оптимального проектування та інтелектуального діагностування систем (ROD & IDS) в умовах невизначеності **.**

ОДИНЕЦЬ Олександр Олександрович – студент групи КУ-41 факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: odinecaleksandr17@gmail.com; ORCID: 0000-0002-5683-9820.

Наукові інтереси:

– методи машинного навчання в багатокритеріальних задач робастної оптимального проектування та інтелектуального діагностування систем (ROD & IDS) в умовах невизначеності.

УГРЮМОВ Михайло Леонідович – д.т.н., проф., професор кафедри ТПС; факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: ugryumov.mykhaylo52@gmail.com; ORCID: 0000-0003-0902-2735.

Наукові інтереси:

– методи машинного навчання в багатокритеріальних задач робастної оптимального проектування та інтелектуального діагностування систем (ROD & IDS) в умовах невизначеності.

УДК 004.852 **СЕКЦІЯ 2**

БЕЗЛЮБЧЕНКО А. В., УГРЮМОВ М. Л., МЕНЯЙЛОВ Є. С.

МЕТОД ГЛИБОКОГО НАВЧАННЯ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ У РАЗІ НЕДИФЕРЕНЦІЙОВАНИХ ФУНКЦІЙ ВТРАТ

Нейромережеві технології знайшли застосування в економіці, медицині, промисловості, багатьох інших областях науки і техніки, здатні вирішувати практично будь-які завдання, пов'язані з моделюванням, прогнозуванням, оптимізацією [1]. Їм приписують властивості самонавчання, самоорганізації, здатності до обробки образної інформації на противагу звичайним алгоритмам, які також традиційно вважаються жорстко заданими, не здібними до навчання і призначеними для обробки символьної інформації.

Основними функціями штучних нейронних мереж (ШНМ) є:

– функція апроксимації, яка актуальна при вирішенні задач, ідентифікації моделей на основі обробки даних;

– функція класифікації та розпізнавання образів, яка актуальна, наприклад, при вирішенні задач діагностування станів об'єкта;

– функція прогнозування, актуальна при оцінці майбутньої поведінки системи за наявною послідовністю її попередніх станів;

– функція ідентифікації та оцінювання, актуальна при вирішенні задач управління динамічними процесами;

– функція асоціативного управління.

Дана робота присвячена використанню методів чисельного диференціювання при вирішенні некоректних задач відновлення багатовимірних статистичних залежностей на основі емпіричних даних, побудові нейромережевих моделей (на прикладах односпрямованої багатошарової і радіально-базисної ШНМ) за допомогою deep learning methods.

Постановка задачі

Задана векторна функція набором навчальних пар $\left(\vec{Y}^{(0)},\vec{F}\right)_p$, $p=1...P$, де $\vec{Y}^{(0)},\vec{F}$ – вектор входу, розмірності H_0, і виходу, розмірності H_{K+1}, відповідно. Необхідно апроксимувати дану вибірку. Результатом вирішення задачі повинен бути якийсь математичний механізм, в результаті роботи якого можна було б отримати будь-яке значення векторної функції $\vec{Y}_p^{(\mathrm{K}+1)}\left(\vec{Y}_p^{(0)}\right)$, представленою даною навчальної вибіркою, по заданому вектору входу, в діапазоні, обмеженому вхідними даними.

Структура та метод навчання односпрямованої багатошарової штучної нейронної мережі (ОБМ) і радіально-базисної штучної нейронної мережі (РБМ) описано в роботах [2-6].

Виберемо в якості функції втрат скалярну згортку цільових функцій [2]:

$$E=\frac{1}{2PH_{K+1}}\sum_{p=1}^{P}\gamma^{P-p}\left\{\sum_{i=1}^{H_{K+1}}\left(f_{fit}\left[t_p^2\right]+\beta_F\cdot f_{fit}\left[Ro_p\right]\right)\right\},$$

де k = 1 – число прихованих шарів в ШНМ,

$$t^2=\frac{n_\alpha MD^2(\vec{F}\,|\,\vec{D})}{2},\ Ro=\frac{\left|\chi^2-k\right|}{\sqrt{2k}},k=n_\alpha-3,n_\alpha\ \square\ 4,\ \chi^2=\frac{n_\alpha}{M}(\vec{\sigma}_f)^T\hat{R}\vec{\sigma}_f,$$

$$MD^2(\vec{F}\,|\,\vec{D})=\frac{1}{H_{K+1}}\left[\vec{F}(\vec{\mathrm{X}},\mathrm{W})-\vec{D}\right]^T\hat{\Sigma}^{-1}\left[\vec{F}(\vec{\mathrm{X}},\mathrm{W})-\vec{D}\right],$$

$$\vec{F}=\{\mathrm{f}_i\},i=1...\mathrm{H}_{K+1},\ \vec{D}=\{\mathrm{d}_i\},i=1...\mathrm{H}_{K+1},$$

$\vec{X}=\{x_{m}\},m=1...M$, W – матриця параметрів ШНМ.

© БЕЗЛЮБЧЕНКО А. В., УГРЮМОВ М. Л., МЕНЯЙЛОВ Є. С., 2021

$\bar{\sigma}_d = \{\frac{\sigma_{f_i}}{\sigma_{d_i}}\}, i = 1...H_{K+1}$, $\hat{\Sigma} = \Sigma + \delta E$ – коваріаційна матриця, $\delta > 0$,

$\hat{R}$ – кореляційна матриця.

Введена функція втрат є важко аналітично диференційованою.

Ініціалізація початкових значень ваг для ОБМ:

а) $w_{ij}(0) = N(w_{ij} \mid 0, \sqrt{\frac{2}{H_0 + H_2}})$ – He/Xavier ініціалізація;

б) $w_{ij}(0) = \pm r, r = \sqrt{\frac{6}{H_0 + H_2}}$, рівномірний розподіл.

Введемо позначення:

$E_{ij} = E(w_{ij}(W_{-ij}))$, $w_{ij} = \text{var}$, $w_{ln} = const$, $l \neq i$, $k \neq j$;

$S_{ij} = -E'_{ij}, E'_{ij} = \underset{w_{ij}}{\text{Der}}[E \mid W_{-ij}]$ – часткова похідна $E(w_{ij} \mid W_{-ij})$ по w_{ij} ;

ОБМ: $h = 0...H_0, j = 0...H_1, i = 1...H_2$. Знаходяться $\underset{(1)}{\{w_{ij}\}} \longrightarrow \underset{(2)}{\{w_{jh}\}}$ методом зворотного поширення помилки.

РБМ: $j = 0...H_1, i = 1...H_2$ (при інтерполяції: $w_0 = 0$). Знаходяться: $\underset{(1)}{\{w_j\}} \longrightarrow \underbrace{\{c_j\}, \{\sigma_j\}}_{(2)}$.

Чисельне диференціювання функції втрат при побудові нейромережевих моделей, що навчаються.

В основі чисельного диференціювання лежить апроксимація функції, від якої береться похідна, інтерполяційним многочленом. Основними завданнями є обчислення її часткових похідних в опорній точці шаблону.

Розглянемо метод численного диференціювання функції втрат, що забезпечує перший порядок точності апроксимації часткових похідних. Згенеруємо вибірку значень ваг з вибраною щільністю розподілу вірогідності та обчислимо відповідні їм значення функції втрат $\{E(l_j \mid l_i)\}$ в околиці опорної точки.

Нехай $l^\circ = (l_j^\circ, l_i^\circ)$, l_j° – вектор параметрів ШНМ, l_i° – вектор вихідних сигналів ШНМ, $\square E(l_j \mid l_i) = E(l_j \mid l_i) - E(l_{j,0} \mid l_{i,0})$.

Для ОБМ, знаходимо $\{E'_{ij}(l_0)\}$.

$\overbrace{[\square w_{ij}(l)]}^{матриця} \overbrace{\{E'_{ij}(l_0)\}}^{вектор} = \overbrace{\{\square E(l_j \mid l_{i,0})\}}^{вектор}$ – система лінійних алгебраїчних рівнянь.

Далі знаходимо $\{E'_{jh}(l_0)\}$.

$[\square w_{jh}(l)]\{E'_{jh}(l_0)\} = \{\square E(l_h \mid l_i)\}$,

$i = 1...H_2; j = 0...H_1; h = 0...H_0, l = \{l_n\}, n = 1...N; N = (H_0 + 1) \cdot H_1 + (H_1 + 1) \cdot H_2$ – кількість точок.

Для РБС, знаходимо $\{E'_{c_j}\}, \{E'_{\sigma_j}\}$.

Розглянемо метод численного диференціювання функції втрат, що забезпечує другий порядок точності апроксимації часткових похідних. Згенеруємо вибірку значень ваг та обчислимо відповідні їм значення функції втрат $\{E_{ij}\}$, $E_{ij} = E(\mathrm{w}_{ij} \mid \mathrm{W}_{-ij})$ в околиці опорної точки відповідно до вибраного шаблону. Триточковий шаблон чисельного диференціювання зображено на рисунку 1.

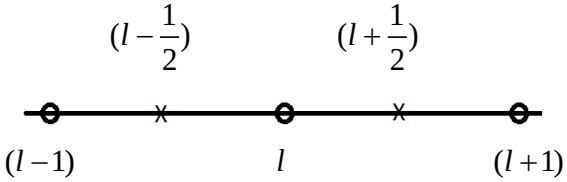


Рис. 1. Триточковий шаблон

Представимо сіткову функцію на обраному шаблоні в формі многочлену Лагранжу, отримаємо:

$$E'_{ij}(\hat{l}) = E_{ij}(l-1)[\frac{2\hat{w}_{ij} - w_{ij,(l+1)} - w_{ij,l}}{(w_{ij,(l+1)} - w_{ij,l})(w_{ij,(l-1)} - w_{ij,(l+1)})} +$$

$$E_{ij}(l)[\frac{2\hat{w}_{ij} - w_{ij,(l-1)} - w_{ij,(l+1)}}{(w_{ij,l} - w_{ij,(l-1)})(w_{ij,l} - w_{ij,(l+1)})}] + ,$$

$$E_{ij}(l+1)[\frac{2\hat{w}_{ij} - w_{ij,(l-1)} - w_{ij,l}}{(w_{ij,(l+1)} - w_{ij,(l-1)})(w_{ij,(l-1)} - w_{ij,l})}]$$

$$E'_{ij}(l) = \min\text{mod}[E'_{ij}(l-\frac{1}{2}), E'_{ij}(l+\frac{1}{2})],$$

$$\min\text{mod}(a,b) = \frac{1}{2}[sign(a) + sign(b)]\min(|a|,|b|).$$

Аналогічним чином можливо отримати $\{E'_{jh}(l)\}$.

Обчислення часткових похідних функції втрат можливо виконувати із застуванням сплайнів Хіроши Акіма (кінцево-різницева схема на основі 5-ти точкового шаблону).

Розглянемо метод численного диференціювання функції втрат, що забезпечує третій порядок точності апроксимації часткових похідних –TVD-ISNAS кінцево-різницева схема Зійлеми:

$$E'_{ij}(l \pm \frac{1}{2}) = \begin{cases} \dfrac{\Delta E_{ij}[(l+\frac{1}{2}) \pm \frac{1}{2}]}{\dfrac{[\Delta w_{ij}(l) + \Delta w_{ij}(l+1)]}{2}} \cdot \dfrac{[\Delta E_{ij}(l)]^2 + 3\Delta E_{ij}(l)\Delta E_{ij}(l+1)}{[E_{ij}(l) + \Delta E_{ij}(l+1)]^2}, \text{ при } \Delta E_{ij}(l) \cdot \Delta E_{ij}(l+1) > 0 \\ 0, \text{ при } \Delta E_{ij}(l) \cdot \Delta E_{ij}(l+1) \le 0 \end{cases},$$

де $\Delta E_{ij}(l) = E_{ij}(l) - E_{ij}(l-1), \Delta E_{ij}(l+1) = E_{ij}(l+1) - E_{ij}(l),$

$\Delta w_{ij}(l) = w_{ij}(l) - w_{ij}(l-1); w_{ij}(l+1) = w_{ij}(l+1) - w_{ij}(l),$

$E'_{ij}(l) = \min\text{mod}[E'_{ij}(l-\frac{1}{2}), E'_{ij}(l+\frac{1}{2})].$

Аналогічним чином можливо отримати $\{E'_{jh}(l)\}$.

Висновки

У роботі описано використання методів чисельного диференціювання функції втрат при вирішенні некоректних задач відновлення багатовимірних статистичних залежностей на основі емпіричних даних у формі нейромережевих моделей (на прикладах односпрямованої багатошарової і радіально-базисної ШНМ), що навчаються.

ЛІТЕРАТУРА

1. Тархов Д. А. Нейронные сети. Модели и алгоритмы. Кн.18. Справочное издание. (Серия 'Нейрокомпьютеры и их применение'). М. Радиотехника, 2005. 256с.
2. Chernysh S. V., Meniailov Ie. S., Ugryumova K. M., Ugryumov M. L. A method synthesis of selection function scalar convolutions for the multi-objective decision-making problems. *Bulletin of V. Karazin Kharkiv National University. Series «Mathematical Modelling. Information Technology. Automated Control Systems»*, 2015. Issue 27. pp. 172-180.
3. Meniailov, I., Ugryumov, M., Chumachenko, D., Bazilevych, K., Chernysh, S., Trofymova, I. Non-linear Estimation Methods in Multi-objective Problems of Robust Optimal Design and Diagnostics of Systems Under Uncertainties. *Advances in Intelligent Systems and Computing*, vol. 1113, 2020, pp. 198-207. https://doi.org/ 10.1007/978-3-030-37618-5_18.
4. Ugryumova, K., Meniailov, I., Trofymova, I., Ugryumov, M. and Myenyaylov, A. Synthesis Of Robust Optimal Control Program For Axial Flow Compressor Turning Guide Vanes, *International Journal of Computing*, 19(3), 2020. pp. 347-354. doi: 10.47839/ijc.19.3.1882.
5. Bakumenko N., Strilets V., Meniailov I., Chernysh S., Ugryumov M., Goncharova T. Synthesis Method of Robust Neural Network Models of Systems and Processes. *Integrated Computer Technologies in Mechanical Engineering*. 2021. vol 188. Springer, Cham. https://doi.org/10.1007/978-3-030-66717-7_1
6. Методи машинного навчання у задачах системного аналізу і прийняття рішень: монографія / В.Є. Стрілець, С.І. Шматков, М.Л. Угрюмов, Є.С.Мєняйлов, С.В. Черниш, К.М. Угрюмова – Харків : Харківський національний університет імені В. Н. Каразіна, 2020. – 195 с. (ISBN 978-966-285-627-9)

БЕЗЛЮБЧЕНКО Артем В'ячеславович - аспірант кафедри математичного моделювання та штучного інтелекту Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут», Чкалова 17, м. Харків, Україна, 61070; e-mail: a.bezlyubchenko@khai.edu; ORCID: 0000-0002-0310-1835.

Наукові інтереси:

- *математичне моделювання;*
- *системи та засоби штучного інтелекту;*
- *методи обчислень.*

УГРЮМОВ Михайло Леонідович - д.т.н., професор, професор кафедри теоретичної та прикладної системотехніки Харківського національного університету імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: m.ugryumov@karazin.ua; ORCID: 0000-0003-0902-2735.

Наукові інтереси:

- *методи обчислювального інтелекту в багатокритеріальних задачах робастного оптимального проектування та інтелектуального діагностування систем і процесів в умовах невизначеності.*
- *системи підтримки прийняття рішень.*

МЕНЯЙЛОВ Євген Сергійович – старший викладач кафедри теоретичної та прикладної інформатики, Харківського національного університету імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: j.menyailov@khai.edu; ORCID: 0000-0002-9440-8378.

Наукові інтереси:

- *методи обчислювального інтелекту в багатокритеріальних задачах робастного оптимального проектування та інтелектуального діагностування систем і процесів в умовах невизначеності;*
- *системи підтримки прийняття рішень.*

УДК 519.6 + 519.85, 517.977 **СЕКЦІЯ 2**

БЕЗСМЕРТНИЙ Д.Р., КОНДРАТЮК Д.С.

МОДЕЛІ КЛАСИФІКАЦІЇ СТАНІВ КОМП'ЮТЕРНОЇ МЕРЕЖІ НА ОСНОВІ ОПТИМІЗАЦІЇ РОЮ ЧАСТОК ТА БДЖОЛИНОГО РОЮ

Класифікація – один з розділів машинного навчання, присвячений вирішенню наступного завдання. Є безліч об'єктів (ситуацій), розділених деяким чином на класи. Задана кінцева множина об'єктів, для яких відомо, до яких класів вони належать. Ця множина називається навчальною вибіркою. Класова приналежність інших об'єктів не відома. Потрібно побудувати алгоритм, здатний класифікувати довільні об'єкти з початкової множини.

Для ефективного вирішення практично значимих задач оптимізації необхідно використання паралельних методів оптимізації. Використання паралельних обчислювальних систем тягне за собою необхідність максимально ефективно відобразити використовувані алгоритми оптимізації на апаратну архітектуру.

Найбільш перспективними з точки зору проектування паралельних алгоритмів з урахуванням архітектури паралельних систем виглядають *популяційні алгоритми*, так як вони володіють природним паралелізмом. У них на логічному рівні паралельно еволюціонує значне число потенційних рішень.

Популяційні алгоритми мають ряд переваг в порівнянні з класичними при вирішенні мультимодальних задач оптимізації високої розмірності: універсальність застосування, простота реалізації, легка масштабованість та ін. Потрібно зазначити, що популяційні алгоритми дозволяють також ефективніше виявляти рішення, близькі до оптимального.

Роєвий інтелект

Під роєвою оптимізацією розуміється клас алгоритмів, спрямованих на вирішення складних оптимізаційних задач (дискретна оптимізація, багатовимірна оптимізація і багатокритеріальна оптимізація), робота яких заснована на моделюванні колективної поведінки різних колоній живих організмів.

Системи роєвого інтелекту (swarm intelligence) представляють собою моделі різноманітних типів колективної поведінки в колоніях різних живих організмів – зграях птахів (bird flocks), косяків риб (fish schools), колоніях мурашок і т.д.

Відмінною особливістю таких моделей є те, що кожен окремий організм в цій моделі виконує дуже прості дії, підлеглі своїй локальної мети, взаємодіючи з обмеженим числом інших організмів в розглянутій колонії.

Бджолиний пошук

Базовий варіант цього алгоритму є комбінацією алгоритмів локального і випадкового пошуку.

Алгоритм бджолиного пошуку:

Спочатку з вулика вилітають в випадковому напрямку якась кількість бджіл-розвідників, які намагаються відшукати ділянки, де є нектар. Через якийсь час бджоли повертаються у вулик і особливим чином повідомляють іншим де і скільки вони знайшли нектару.

Після цього на знайдені ділянки відправляються інші бджоли, причому чим більше на даній ділянці передбачається знайти нектару, тим більше бджіл летить в цьому напрямку. А розвідники знову відлітають шукати інші ділянки, після чого процес повторюється.

Програмна реалізація задачі класифікації на мові програмування Python

Для початку, нам потрібно згенерувати дані, на яких будуть проводитися експерименти (рис.1).

```
def generateData(numberOfClassEl, numberOfClasses):
    data = []
    for classNum in range(numberOfClasses):
        # Choose random center of 2-dimensional gaussian
        centerX, centerY = random.random() * 5.0, random.random() * 5.0
        # Choose numberOfClassEl random nodes with RMS=0.5
        for rowNum in range(numberOfClassEl):
            data.append([[random.gauss(centerX, 0.5), random.gauss(centerY, 0.5)], classNum])
    return data
```

Рис. 1 Код створення даних

© БЕЗСМЕРТНИЙ Д.Р., КОНДРАТЮК Д.С., 2021

Код візуалізації даних (було обрано 3 класи, в кожному з них 40 елементів). Щоб подивитися на отриману вибірку, потрібно виконати наступний код (рис. 2).

```
def showData(nClasses, nItemsInClass):
    trainData = generateData(nItemsInClass, nClasses)
    classColormap = ListedColormap(['#FF0000', '#00FF00', '#FFFF00'])
    pl.scatter([trainData[i][0][0] for i in range(len(trainData))],
               [trainData[i][0][1] for i in range(len(trainData))],
               c=[trainData[i][1] for i in range(len(trainData))],
               cmap=classColormap)
    pl.show()


showData(3, 40)
```

Рис. 2 Код візуалізації даних

Приклад отриманого в результаті виконання цього коду зображення: (рис 3).

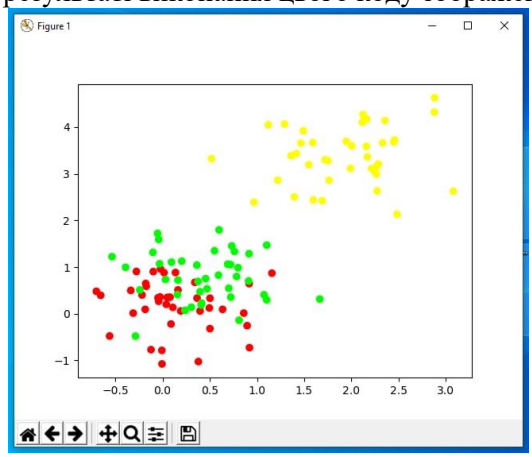


Рис 3 Візуалізація

Отже, маємо набір об'єктів, для кожного з яких заданий клас. Тепер потрібно розбити цю множину на дві частини: навчальну вибору і тестову вибірку. Для цього служить наступний код (рис. 4).

```
def splitTrainTest(data, testPercent):
    trainData = []
    testData = []
    for row in data:
        if random.random() < testPercent:
            testData.append(row)
        else:
            trainData.append(row)
    return trainData, testData
```

Рис. 4 Код поділу об'єктів на навчальну вибірку і тестову вибірку

Тепер, маючи навчальну вибірку, можна реалізувати і сам алгоритм класифікації (рис. 5).

```
def classifyKNN(trainData, testData, k, numberOfClasses):
    # Euclidean distance between 2-dimensional point
    def dist(a, b):
        return math.sqrt((a[0] - b[0]) ** 2 + (a[1] - b[1]) ** 2)

    testLabels = []
    for testPoint in testData:
        # Calculate distances between test point and all of the train points
        testDist = [[dist(testPoint, trainData[i][0]), trainData[i][1]] for i in range(len(trainData))]
        # How many points of each class among nearest K
        stat = [0] * numberOfClasses
        for d in sorted(testDist)[0:k]:
            stat[d[1]] += 1
        # Assign a class with the most number of occurrences among K nearest neighbours
        testLabels.append(sorted(zip(stat, range(numberOfClasses)), reverse=True)[0][1])
    return testLabels
```

Рис. 5 Алгоритм класифікації

Для визначення відстані між об'єктами можна використовувати не тільки евклідову відстань: також застосовуються манхеттенська відстань, косинусна міра, критерій кореляції Пірсона та ін.

Перевірка роботи алгоритму (до) (рис. 6).

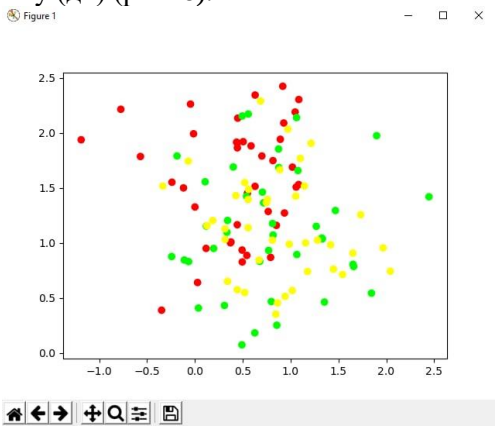


Рис.6 Перевірка роботи алгоритму (до)

Перевірка роботи алгоритму (після) (рис. 7).

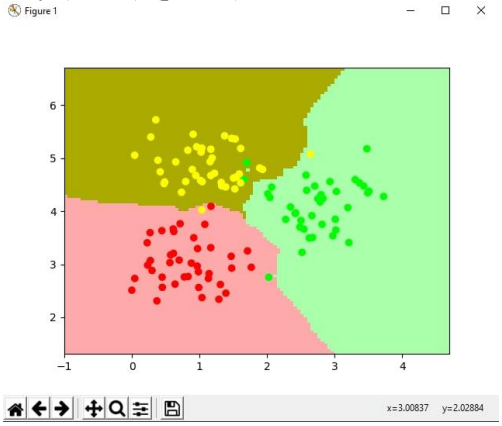


Рис. 7 Перевірка роботи алгоритму (після)

Висновок

kNN - один з найпростіших алгоритмів класифікації, тому на реальних завданнях він часто виявляється неефективним. Крім точності класифікації, проблемою цього класифікатора є швидкість класифікації: якщо в навчальній вибірці N об'єктів, в тестовій виборі M об'єктів, а розмірність простору - K, то кількість операцій для класифікації тестової вибірки може бути оцінений як O (K * M * N).

Програмна реалізація роєвих алгоритмів на мові програмування Python

Алгоритм оптимізації бджолиного рою (результат) (рис. 8).

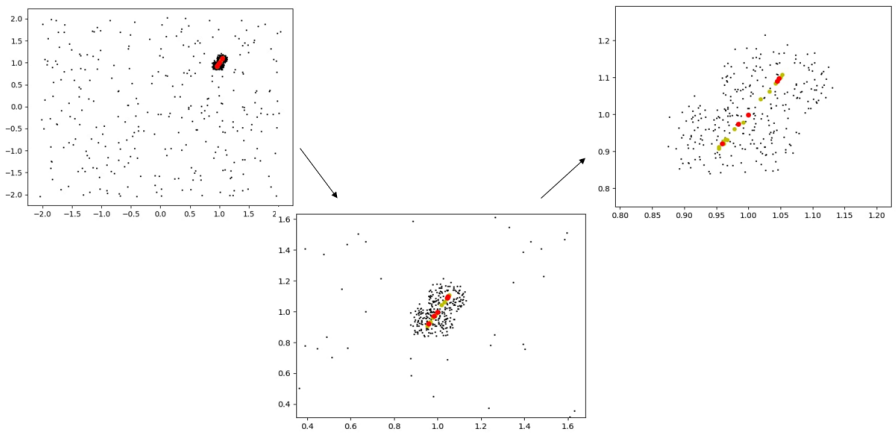


Рис. 8 Алгоритм оптимізації бджолиного рою

Алгоритм оптимізації роєм часток (результат) (рис. 9).

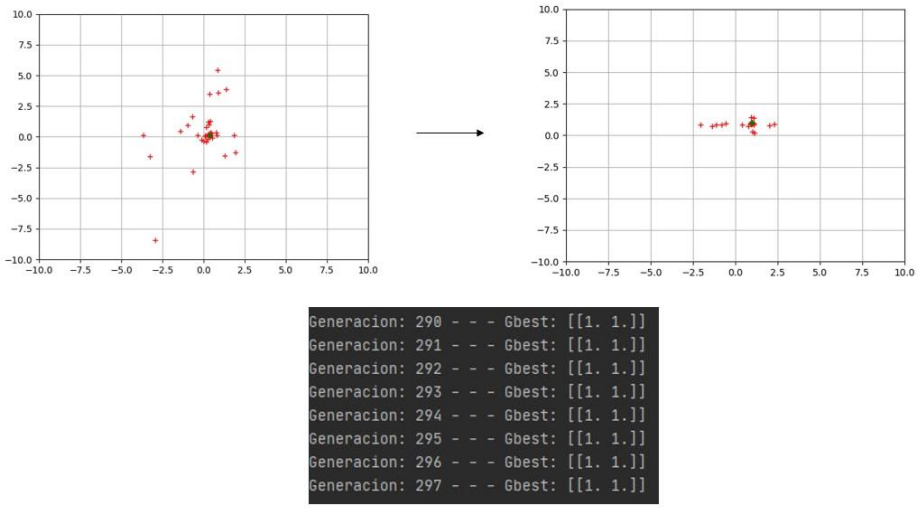


Рис. 9 Алгоритм оптимізації роєм часток

Був розроблений код роботи класифікації, який розділяє безліч об'єктів на класи, для того, щоб в подальшому об'єднати з алгоритмами роєвої оптимізації.

Проаналізувавши підхід до вирішення задачі класифікації, можемо зробити висновок, що популяційні алгоритми не підходять для вирішення завдань класифікації, через те, що вони можуть знаходити тільки екстремуми функції, а в задачі класифікації потрібно визначити належність об'єкта до якогось певного класу за допомогою навчальної вибірки (найчастіше це масив ознак), тому популяційні алгоритми не можуть впоратися з цим завданням .

ЛІТЕРАТУРА

1. Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы вдохновленные природой. 2017.
2. Darrel Whitley, A Genetic Algorithm Tutorial, Statistics and Computing (4): 65–85, 1994.
3. Darrel Whitley, An Overview of Evolutionary Algorithms: Practical Issues and Common Pitfalls, Journal of Information and Software Technology 43: 817–831, 2001.
4. K. Deb, S. Agrawal, Understanding Interactions Among Genetic Algorithm Parameters, 1998.
5. Авторский сайт Ю. Цоя (http://www.qai.narod.ru/).
6. Исаев С.А. Популярно о генетических алгоритмах (http://algolist.manual.ru/ai/ga/ga1.php).
7. Darrel Whitley "A Genetic Algorithm Tutorial", 1993.
8. Koza, John R. Genetic programming: on the programming of computers by means of natural selection, A Bradford book, The MIT Press, London, 1992
9. Кононюк А. Е Дискретно-непрерывная математика. (Алгоритмы). —В 12-и кн. Кн. 10,Ч.1— К.: 2017.

БЕЗСМЕРТНИЙ Денис Русланович – студент кафедри прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: gerbyboyz@gmail.com; ORCID 0000-0002-2362-644X.

Наукові інтереси:

- *системний аналіз;*
- *методи машинного навчання.*

КОНДРАТЮК Діана Сергіївна – студент кафедри прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: kondrdi2000@gmail.com; ORCID 0000-0003-3740-4220.

Наукові інтереси:

- *методи машинного навчання.*
- *методи штучного інтелекту.*

УДК 004.032.26 **СЕКЦІЯ 7**

BELKOV D.O.

COMPUTER MODEL OF THE SYSTEM OF ANALYSIS AND FORECASTING OF FINANCIAL TIME SERIES USING NEURAL NETWORK MODELS

Supervised learning

Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labeled training data consisting of a set of training examples. In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal). A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances. This requires the learning algorithm to generalize from the training data to unseen situations in a "reasonable" way. This statistical quality of an algorithm is measured through the so-called generalization error.

Advantages and disadvantages of supervised learning

Supervised learning in comparison with other learning paradigms has following advantages:

1. Supervised learning is a simple process to understand.
2. Knowledge of how many classes are there before giving the data for training.
3. It is possible to be very specific about the definition of the classes, possibility to train the classifier in a way which has a perfect decision boundary to distinguish different classes accurately.
4. After the entire training is completed, it is not necessary to keep the training data in your memory, instead of this keep the decision boundary as a mathematical formula.
5. Supervised learning can be very helpful in classification problems.
6. Another typical task of supervised machine learning is to predict a numerical target value from some given data and labels.

However it has several disadvantages, such as

1. Limited in a variety of sense. It can't handle some of the complex tasks in machine learning.
2. Don't give unknown information from the training data like unsupervised learning do.
3. Cannot cluster or classify data by discovering its features on its own.
4. In the case of classification, giving an input that is not from any of the classes in the training data, then the output may be a wrong class label.
5. If, training set does not include some examples that you want to have in a class, then after training, you might not get the correct class label as the output.
6. With vad examples the accuracy of the model will be very bad.
7. Training needs a lot of computation time.
8. Hard to find the information. A lot of the time, the machine needs to learn by itself from the training data.

Reinforcement learning

Reinforcement learning (RL) is an area of machine learning concerned with how intelligent agents ought to take actions in an environment in order to maximize the notion of cumulative reward. Reinforcement learning is one of three basic machine learning paradigms, alongside supervised learning and unsupervised learning.

Reinforcement learning differs from supervised learning in not needing labelled input/output pairs be presented, and in not needing sub-optimal actions to be explicitly corrected. Instead the focus is on finding a balance between exploration (of uncharted territory) and exploitation (of current knowledge).

© BELKOV D.O., 2021

The environment is typically stated in the form of a Markov decision process (MDP), because many reinforcement learning algorithms for this context use dynamic programming techniques. The main difference between the classical dynamic programming methods and reinforcement learning algorithms is that the latter do not assume knowledge of an exact mathematical model of the MDP and they target large MDPs where exact methods become infeasible.

Reinforcement Learning has following advantages:

1. Can be used to solve very complex problems that cannot be solved by conventional techniques.
2. Preferred to achieve long-term results, which are very difficult to achieve.
3. Is very similar to the learning of human beings.
4. Can correct the errors that occurred during the training process.
5. Once an error is corrected by the model, the chances of occurring the same error are very less.
6. Create the perfect model to solve a particular problem.
7. Robots can implement reinforcement learning algorithms to learn how to walk.
8. In the absence of a training dataset, it is bound to learn from its experience.
9. Can outperform humans in many tasks. DeepMind's AlphaGo program, a reinforcement learning model.
10. Intended to achieve the ideal behavior of a model within a specific context, to maximize its performance.
11. Useful when the only way to collect information about the environment is to interact with it.
12. Maintain a balance between exploration and exploitation. Other learning algorithms do not perform this balance.

Reinforcement Learning in comparison with other learning paradigms has following disadvantages:

1. Framework is wrong in many different ways, but it is precisely this quality that makes it useful.
2. Too much reinforcement learning can lead to an overload of states, which can diminish the results.
3. Not preferable to use for solving simple problems.
4. Needs a lot of data and a lot of computation. It is data-hungry.
5. Assumes the world is Markovian, which it is not. The Markovian model describes a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.
6. The curse of dimensionality limits reinforcement learning heavily for real physical systems. According to Wikipedia, the curse of dimensionality refers to various phenomena that arise when analyzing and organizing data in high-dimensional spaces that do not occur in low-dimensional settings such as the three-dimensional physical space of everyday experience.
7. Curse of real-world samples. For example, consider the case of learning by robots. The robot hardware is usually very expensive, suffers from wear and tear, and requires careful maintenance. Repairing a robot system costs a lot.
8. To solve many problems of reinforcement learning, use a combination of reinforcement learning with other techniques rather than leaving it altogether. One popular combination is Reinforcement learning with Deep Learning.

Conclusion

In case of all pros of reinforcement learning and all cons of supervised learning, the best paradigm to create a computer model of the system of analysis and forecasting of financial time series is to choose reinforcement learning.

Python modules which is the best combination with reinforcement learning:

- TensorFlow - an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications. [https://www.tensorflow.org/about]

- NumPy - brings the computational power of languages like C and Fortran to Python, a language much easier to learn and use to work with big data. [https://numpy.org/doc/stable/user/whatisnumpy.html]
- Matplotlib - a comprehensive library for creating static, animated, and interactive visualizations in Python.[https://matplotlib.org/stable/tutorials/index.html#introductory]

REFERENCES

1. Стеценко І.В. Моделювання систем: навч. посіб. Черкаси: ЧДТУ, 2010. 399 с.
2. Олифер В., Олифер Н. Компьютерные сети. Принципы, технологии, протоколы : Учебник для вузов. Питер: 5-е изд. СПб, 2016. 992 с.
3. Лосев Ю. І., Руккас К. М., Шматков С. І. Комп'ютерні мережі: навч. посіб. / за редакцією Ю. І. Лосева. Харків : ХНУ імені В. Н. Каразіна, 2013. 248 с.
4. Ляшенко В.П., Кобильська О.Б., Дям'янченко О.П. Математичні моделі теплообміну з умовами імпедансного типу у багатошарових областях. Вісник Кременчуцького національного університету імені Михайла Остроградського. 2017. Вип. 6/2017 (106). С. 37–43.
5. Боков І. П., Бондаренко Н. С., Стрельнікова О. О. Дослідження поведінки узагальнених переміщень, отриманих із використанням теорії {m,n}-апроксимації. Вісник Харківського національного університету ім. В.Н. Каразіна серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління». 2018. Вип. 38. С. 14–24.
6. Bulanchuk G., Ostapenko A. Modeling of the viscous fluid flow around rotating circular cylinders with the lattice Boltzmann method at moderate Reynolds numbers. Bulletin of V. Karazin Kharkiv National University, series Mathematical Modelling. Information Technology. Automated Control Systems. 2017. Issue 36. P. 27–37. DOI: 10.26565/2304-6201-2019-41-09. URL: https://periodicals.karazin.ua/mia/article/view/10086/9614 (Last accessed: 20.11.2018).
7. Зайцев, А. Тенденции в области искусственного интеллекта. Современные методы машинного обучения / А. Зайцев // Видеонаука: сетевой журн. - 2018. - No1(9). –URL: https://videonauka.ru/stati/32-vystavki- konferentsii-seminary/182-tendentsii-v-oblasti-iskusstvennogo-intellekta-sovremennye-metody- mashinnogo-obucheniya
8. Исхакова, А. Ф. Применение искусственного интеллекта / А. Ф. Исхакова // Вестник современных исследований. - 2018. - No 9.3 (24) .- С. 261-262. – https://elibrary.ru/item.asp?id=36275758

BELKOV Dmytro – student of engineering, V.N. Karazin Kharkiv National University, Svobody Square, 4, Kharkiv, Kharkiv Oblast, 61000; akirinoshi@gmail.com ORCID*: 0000-0002-8870-8606

Science interests:

– *neural networks*

UDC 004.415.2.031.43 **СЕКЦІЯ 7**

BENHADDOU M., KHRUSLOV M.

ROBUST ESTIMATION FOR MODEL CONSUMPTION ENERGY CARS

Abstract

In this period statistical analysis are becoming more important, LSM method still continue to be one of the most used methods among regression parameters estimation techniques. However, when a data set has an outlier, using LSM method by excluding these outliers from the data or including them as they are may give wrong results.

Introduction

Robust estimation is an important method for analyzing data that are contaminated with outliers, robust estimation is a form of regression analysis which is designed to circumvent some limitations of traditional parametric and non-parametric methods, Robust estimation methods are designed to be not overly affected by violations of assumptions by the underlying data-generating process. Throw this work, I will emphasize on optimization/way of consumption battery energy cars by using robust estimation method and model.

This graph EXPLAINS drive cycle for drivers and ITS relation with consumption of battery (speed), there is a data and this figure from which I got some information to make the statistical model below:

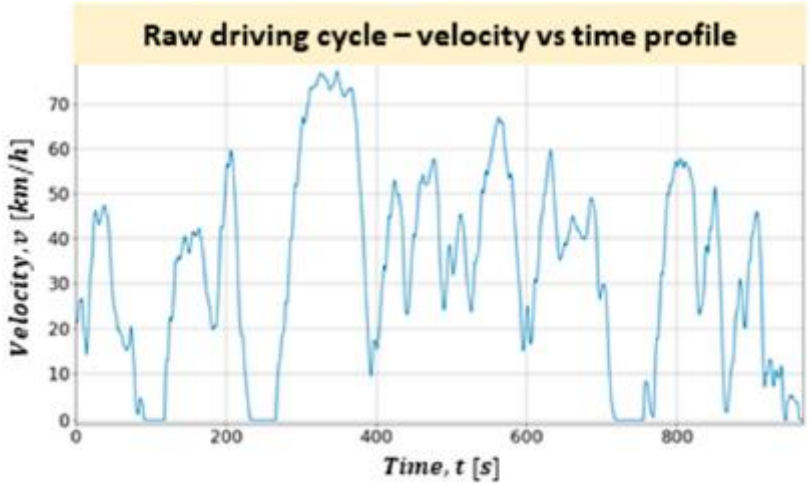


figure 1: Drive cycle related to speed

Methods of robust estimation:

the method of robust estimation is used for data consumption of energy cars, which can be described by this equation:

$$Y = X\beta + \varepsilon.$$

there is a relation between real consumption and speed, to make it happen it is important to introduce some robust estimation estimators.

M-estimator:

The most common general method of robust regression is M-estimation, introduced by Huber (1964), the method of M-estimation is a generalization of maximum likelihood estimation, to demonstrate this relation mentioned before, there is a function:

$$\hat{\beta}_M = min \sum_{i=1}^{n} \rho\left(y_i - \sum_{j=0}^{k} x'_{ij}\beta_j\right). \qquad T = \arg min_t \left\{ J = \sum_{i=1}^{N} \rho(x_i, t) \right\}$$

© BENHADDOU M., KHRUSLOV M., 2021

and
M estimators are solutions" t", which minimize:

$$\sum_{i=1}^{N} \rho(x_i, t)$$

,

This minimization can always be done directly, often it is simpler to differentiate it with *t* and solve the problem, here is the model that represent the regression linear using the parameters of estimation (equation) for our case:

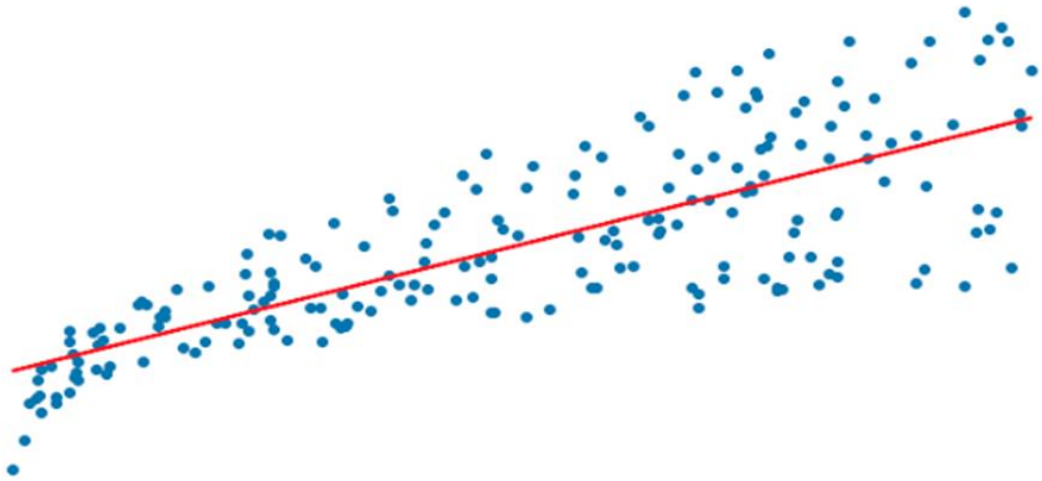


figure2: Description real consumption battery for energy cars

This is the equation that i got it from the model y = 2.01467487X-3.9057602 which we can see the relation between consumption and speed. There is an accuracy of 88% , it's an evaluation of classification for the model and show good results.

Conclusion

I can conclude that this simple demonstration of consumption battery energy cars by using statistical model represent for me good way to make more research in this topic, which I will introduce neural networks (Artificial Intelligence) and give some recommendations to drivers.

REFERENCES:

1. Alma, Ö. G. (2011). Comparison of Robust Regression Methods in Linear Regression. International Journal of Contemporary Mathematical Sciences, 6(9), 409-421.
2. Baltagi, Badi H. (2013) Econometric Analysis of Panel Data, 5th ed., John Wiley and Sons.
3. Dodge, Y. (2008). The Concise Encyclopedia of Statistics. Springer Science & Business Media.
4. Doornik, J. A. (2011). Robust Estimation Using Least Trimmed Squares, Institute for Economic Modelling, Oxford Martin School, and Economics Department, University of Oxford, UK.
5. Fox, J. (2002). Robust Regression. An R and S-Plus Companion to Applied Regression.
6. Fox, J. (2008) Applied Regression Analysis and Generalized Linear Models, Second Edition. Sage.

BENHADDOU M. – student, V. N. Karazin Kharkiv National University, Svobody Sq.4, Kharkiv, 61022, Ukraine, benhaddou2020ki51@student.karazin.ua, ORCID*: 0000-0003-1931-6489.

KHRUSLOV Maksym – PhD, docent, V. N. Karazin Kharkiv National University, Svobody Sq. 4, Kharkiv, 61022, Ukraine, maksym.khruslov@karazin.ua; ORCID*: 0000000196399340.

УДК 004.4`2 **СЕКЦІЯ 6**

БІЛЬСЬКИЙ Г.М., ЛАБЕНКО Д.П.

СТРУКТУРА КОМПЛЕКСУ ПЗ ДЛЯ УПРАВЛІННЯ ПРОГРАМНИМ ПРОДУКТОМ ТА ЙОГО РОЗРОБКОЮ НА ПРИКЛАДІ ІНТЕРАКТИВНОЇ СИСТЕМИ РОЗКЛАДУ ЗАНЯТЬ ДЛЯ ЗАКЛАДУ ВИЩОЇ ОСВІТИ

Вступ

Розвиток сервісних послуг та вирішення задач у інформаційному просторі у даний час відзначається великим поширенням різних програмних та апаратних платформ, через які користувачі мають доступ до тої чи іншої інформації. Смартфони, планшети, персональні комп`ютери, навіть розумні годинники та фітнес-браслети - у сукупності всі ці платформи можуть дозволяти людям вирішувати різні проблеми вне залежності від їх типу та місцезнаходження. Але головним чинником такої диверсифікації та доступності додатків є поширення інформаційної, хмарної інфраструктури, завдяки якій працює значна частина сервісних систем, у свою чергу віддаючи користувачам лише "готове зображення" та інтерактивні можливості роботи з даною інформацією.

Компанії, що надають такі інфраструктурні послуги, загалом поділили категорії на апаратні та програмні сервіси. В апаратні можуть входити аренда серверного обладнання, або ко-локація (аренда місць для серверів у обладнаних кімнатах, саме серверне обладнання встановлюється вже замовником. Головною темою даної доповіді є якраз програмні сервіси, їх структура, та розробка об'єднаного комплексу для розробки, обслуговування та контролю за роботою програмного продукту, а також з інструментами для автоматизації даних процесів на прикладі проекту інтерактивної системи розкладу навчальних занять.

Актуальність та необхідність роботи

При розгляді питання щодо побудови таких сервісів та комплексу зв'язного ПЗ, які будуть проміжним ланцюгом між інфраструктурою, ідеями та готовим продуктом, потрібно розглянути доцільність та необхідність проектування такої системи.

Порівняно з наявними рішеннями від провідних міжнародних корпорацій, проект даної роботи має достатньо переваг, що можуть зумовити створення і використання саме такого рішення, створеного майже з нуля, для побудови систем, подібних до розглядаємого прикладу (інтерактивної системи розкладу учбових занять):

- Фінансова незалежність - єдині фінансові витрати, пов'язані з даним проектом, відносяться в основному до плати за споживання ресурсів апаратним забезпеченням, та закупівлею самого апаратного забезпечення (сервери та комплектуючі для них). Враховуючи широкий асортимент пропозицій та можливість точного визначення необхідних потужностей для роботи, можна вивести можливість економії засобів, порівняно з використанням готових рішень, які можуть бути не тільки менш відкритими для глибокого налаштування, а й також залежати від регулярної оплати (помісячної, або річної) за надані послуги з використання цих рішень;
- Функціональні та архітектурні можливості - порівняно з використанням готових сервісів від великих компаній, розробка свого рішення з використанням супутніх крос-платформних інструментів може дозволити застосування ПЗ не лише на одній виділеній архітектурі та операційній системі, а й на декількох одночасно, що дозволяє розв'язувати такі вузли у єдиний комплекс незалежно від наявних апаратних чи програмних ресурсів. У додаток до цього, за рахунок прямого доступу до програмного коду також можливо проводити розширення функціоналу для своїх потреб, не орієнтуючись на бізнес-потреби, які є головними у великих компаніях, що реалізують подібні послуги та сервіси;

© БІЛЬСЬКИЙ Г.М., ЛАБЕНКО Д.П., 2021

- Більш гнучке масштабування в залежності від необхідності - за рахунок більш широкого доступу до конфігурації системи та програмного коду, розробник та/або адміністратор має можливість налаштувати та адаптувати сервіси до роботи з різними рівнями навантаження, а також взаємодії з іншими сервісами та компонентами системи. Є можливість для створення будь-яких механізмів взаємодії з зовнішнім середовищем, що не є доступним для всіх продуктів-сервісів у корпорацій, або надається за додаткову плату.

Необхідність у розробці саме такої системи у даному випадку обумовлена наведеними вище факторами, зважаючи на сам кінцевий продукт та його характеристику (обмежений бюджет, відсутність необхідності у географічній доступності у декількох точках світу, а також специфічний функціонал, що потребує спеціального налаштування та роботи з сервісами). Зважаючи також на те, що при розробці продукту, а саме інтерактивної системи розкладу занять, потребується застосування засобів, які дозволять системі працювати і у локальній мережі без підключення до Інтернету, постає питання про проектування сервісів, що майже не залежать від публічно-доступних продуктів, що потребують таке підключення.

Постановка задачі

Розробляємий проект повинен представляти собою зв’язний комплекс програмного забезпечення, основною метою яких є поліпшення умов проектування, розробки, тестування та обслуговування інших програмних продуктів.

- Універсальність - забезпечення можливості розробки та підтримки кінцевого продукту незалежно від впроваджених змін у його роботу, дизайн, або належність;
- Уніфікація - всі інструменти, документація та інтерактивні можливості взаємодії з ними повинні бути доступними на якомога великій кількості платформ та дозволяти користувачам системи отримувати доступ через єдиний інтерфейс;
- Собівартість - розробка даної системи повинна бути якомога дешевшою та використовувати, по можливості, безкоштовні компоненти, або ті, що мають відкритий програмний код;
- Інтеграція та розподіленість - система повинна дозволяти за потреби змінювати інструменти та деякий функціонал, який не потрібен для того чи іншого кінцевого програмного продукту. Видалення, або відключення, чи додавання інструментів до системи повинно мати стандартизований вид, щоб була можливість у майбутньому адаптувати до роботи з нею новий функціонал. У свою чергу, інструменти, що не є критичними для її роботи, повинні бути незалежно розміщені (включаючи використовувані ресурси, дані та програмний код) та мати мінімум залежностей з критичною частиною в плані взаємодії для недопущення проблем з усією системою коли один компонент виходить з ладу;
- Шаблонізація - система повинна підтримувати принцип “шаблонів” для різних інструментів та засобів проектування, розробки та обслуговування, що дозволить економити час у процесі підготовки різних частин кінцевого продукту до розробки або застосування. Такими шаблонами можуть бути шаблони сервісів, репозитаріїв програмного коду, документів, проектів з програмним кодом, проектів кінцевих продуктів загалом, тощо;
- Легкість у обслуговуванні - система повинна дозволяти її адміністраторам можливість зміни налаштувань без необхідності перезавантаження всієї системи або її компонентів, якщо ті зміни не є критичними та впливають на роботу платформи (апаратної чи програмної), де розташована ця система. Компоненти системи повинні мати функціонал щодо забезпечення оновлення конфігурації автоматично та у фоновому режимі після прийняття змін від користувача, а оновлення системи чи її компонентів до нової версії повинні мати мінімальний вплив на її доступність та забирати мінімум часу на проведення самого процесу оновлення;

Застосування принципів та технологій DevOps у системі

DevOps, або Development Operations - це набір методологій, загальних рекомендацій та технологій, розроблених у сфері, що дозволяють стандартизувати та автоматизувати процеси у розробці та підтримці ПЗ під час його використання. Від написання документації, до створення і тестування програмних компонентів в окремій або інтегрованій формах, та насамперед - зведення основних повторюваних процесів, що виконуються вручну до мінімальної взаємодії з системою, або повної автоматизації.

Основні категорії робіт та інструментів DevOps, для підтримки яких розробляються компоненти даного комплексу ПЗ:

- Розробка та аналіз коду, інструменти контролю версій;
- Інструменти безперервної інтеграції (Continuous integration);
- Тестування, інструменти безперервного тестування та аналізу ризиків;
- Збірка програмного забезпечення, репозиторій артефактів збірки;
- Управління параметрами випуску продукту, затвердження випуску та його автоматизація;
- Конфігурація та управління інфраструктурою, інфраструктура як код;
- Вимір продуктивності роботи додатків, взаємодія з кінцевим користувачем;

Проектування дизайну та функціональних можливостей компонентів системи полягає в основному на модель процесів безперервної ітерації у DevOps.

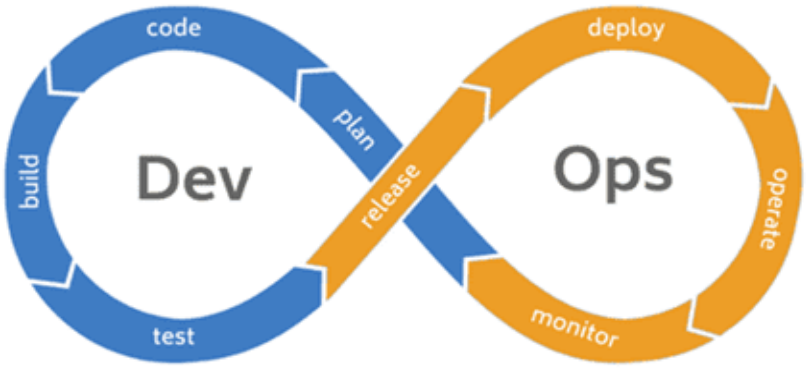


Рис. 1 модель процесів DevOps

Централізована та поетапна взаємодія розробників з різними компонентами та іншими людьми у команді (якщо така є) є основою, що береться до уваги при проектуванні даної системи:

1. Створення основної ідеї проекту, документації, проектування архітектури;
2. Вибір шаблонів для сервісів та інших частин продукту;
3. Розробка основної бізнес-логіки продукту, його тестування та конфігурація компонентів продукту;
4. Компонування за рахунок доступних інструментів та сервісів у єдиний продукт та затвердження випуску;
5. Проектування, розробка та інтеграція змін та/або нового функціоналу у продукт за рахунок інструментів системи;
6. Моніторинг роботи різних частин продукту, коригування параметрів, підготовка та випуск нових версій з виправленнями.

Структура та основні можливості комплексу

Система складається з набору програмного забезпечення, що об’єднаний у комплекс взаємопов’язаних та інтегрованих компонентів, кожний з яких має окремий функціонал. Деякі компоненти можуть мати контроль над іншим, або групою компонентів. Загалом, вся система представляє собою клієнт-серверну архітектуру, де інтерактивна частина, з якою взаємодіють користувачі, відноситься до клієнтської, а основна логіка та виконання поставлених завдань від користувача відбувається на стороні сервера.

У практичному сенсі, для даної системи серверна частина не є єдиною, а навпаки - це розподілений набір сервісів, що можуть бути як на одній фізичній машині (з використанням систем віртуалізації, таких як контейнери або віртуальні машини), або на різних. Ці сервіси

пов’язані між собою у єдину виділену мережу, через яку відбувається взаємодія та обмін даними.

Основні компоненти системи:

- Ключові, або критичні сервіси, що забезпечують повноцінну роботу усього комплексу. Вони пов’язані між собою більш щільно, ніж з компонентами з інших двох груп. До них належать аутентифікація користувачів та адміністраторів, підтримка налаштувань, моніторинг, взаємодія зі сховищами даних, такі як файлова система та бази даних, надання основних методів API для інтеграції інших компонентів;
- Функціональні сервіси, що забезпечують роботу частин системи, наприклад роботу з управлінням проектом, або функції серверу репозитаріїв програмного коду. Вони є незалежними один від одного, а при відсутності/відключення від системи, основна частина зробить цей функціонал недоступним для користувача у інтерфейсі. Кожен сервіс з даної категорії може бути змінений, або оновлений незалежно від ключових, що забезпечує можливість розширення функціоналу без переривання роботи усього комплексу;
- Клієнтська частина - надає користувачам та адміністраторам доступ до функціоналу системи, у тому числі сервісам, що відносяться до функціональних, через єдиний графічний інтерфейс. Він може бути розташований як у веб-браузері, у вигляді виділеного додатка на одній з популярних платформ (iOS, Android, Windows, Linux, тощо), так і у якості додатка для інших інструментів, що не є частиною даної системи (редактори коду, графічні редактори, та інші).

Висновки

Стандартизація та автоматизація процесів під час розробки та підтримки мульти-компонентних програмних продуктів є наявним завданням на сьогоднішний день. Зважаючи на це, великі компанії будують інструменти та сервіси, що можуть допомогти подолати труднощі, пов’язані з цим. Але для розробки та застосування клієнт-серверної системи, що розглядається у даному випадку, є необхідність застосування власного рішення, що може не лише зберегти переваги доступних сервісів, а й бути більш адаптованими до потреб та обмежень, включаючи фінансові, функціональні та обмеження платформ.

Використання модульного підходу при проектуванні даного комплексу сервісів також дозволить змінювати наявні можливості зважаючи на динамічні показники як під часу роботи системи, так і для майбутніх розробок, що робить даний проект унікальною розробкою для обслуговування кінцевих продуктів малого масштабу.

ЛІТЕРАТУРА

1. DevOps - Википедия
   URL: https://ru.wikipedia.org/wiki/DevOps
2. Что такое DevOps и зачем нужен этот Agile-подход: всё о девопс
   URL: https://www.bigdataschool.ru/wiki/devops
3. What are microservices?
   URL: https://microservices.io/
4. Pricing Overview | Google Cloud
   URL: https://cloud.google.com/pricing

БІЛЬСЬКИЙ Георгій Миколайович – студент групи КУ-51 факультету комп'ютерних наук; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків, Україна, 61022; email – mastermangg@gmail.com; ORCID - 0000-0002-3475-5650.

Наукові інтереси:

- *Програмування та архітектура прикладного ПО та систем,*
- *Управління проектами,*
- *Розробка відеоігор*
- *Відео- та 3D-дизайн.*

УДК 519.6: 621.382.233 **СЕКЦІЯ 4**

БОМБА А.Я, МОРОЗ І.П.

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ СТАЦІОНАРНИХ ПРОЦЕСІВ В АКТИВНІЙ ОБЛАСТІ ПОВЕРХНЕВО-ОРІЄНТОВАНИХ *P-I-N* СТРУКТУР МЕТОДАМИ ТЕОРІЇ ЗБУРЕНЬ

Вступ. Постановка задачі

P-i-n структури (плазмові діоди) – нелінійні активні елементи електронної техніки (рис.1). Робота такого пристрою грунтується на зміні провідності активної області діода (*i*-області) шляхом формування електронно-діркової плазми під дією струму управління. Базові характеристики *p-i-n*-діода – згасання, що вноситься у лінію передач, швидкодію – в основному визначають накопичений в *i*-області заряд та геометрія структурних елементів діода [1-3].

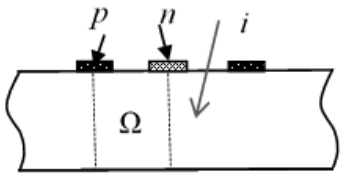


Рис.1 Схематичне зображення перерізу поверхнево-орієнтованої p-i-n структури

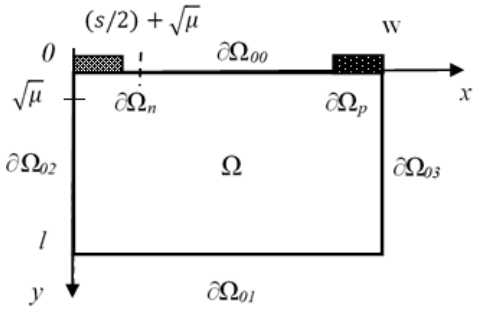


Рис.2 Елемент поверхнево-орієнтованої p-i-n структури

Розглянемо двовимірну просторову модель елемента поверхнево-орієнтованої *p-i-n* структури (область $\Omega = \{(x, y): 0 < x < w, 0 < y < l\}$, рис.2). В основу математичної моделі, яка описує розподіл концентрації дірок (n), електронів (p) та потенціалу (φ) в активній області *p-i-n*-діодів Ω у стаціонарному режимі роботи пристрою, покладемо (напр., як у [1,3]) рівняння неперервності струмів дірок, електронів та рівняння Пуассона (подано у безрозмірному вигляді):

$$\begin{cases} \mu\Delta\varphi = -(p - n), \\ \Delta n = \nabla n \nabla\varphi + n\Delta\varphi + A_n n, \\ \Delta p = -\nabla p \nabla\varphi - p\Delta\varphi + A_p p, \end{cases} \tag{1}$$

де $\mu = \dfrac{\varepsilon\varepsilon_0 kT}{e^2 w^2 N_i}$ (малий параметр задачі, μ~10^{-8}), $A_n = \dfrac{w^2}{D_n \tau_n^*}$, $A_p = \dfrac{w^2}{D_p \tau_p^*}$ τ_p^* , τ_n^* – характерні релаксаційні часи життя дірок та електронів в *i*-області; D_p , D_n – коефіцієнти дифузії відповідно дірок та електронів; e – заряд електрона; ε, ε_0 – відносна діелектрична проникність напівпровідника та діелектрична стала; T – температура (300 ºК); k – стала Больцмана; N_i - стала, визначає концентрацію електронів у власному напівпровіднику. Система рівнянь (1) доповнюється умовами на межі області $\partial\Omega = \partial\Omega_0 \cup \partial\Omega_n \cup \partial\Omega_p$ (також записані у безрозмірному вигляді):

$$nE_\nu + \frac{1}{w}\frac{\partial n}{\partial \nu} - 2\gamma_n n\Big|_{\partial\Omega_n} = -\frac{J}{eD_n N_i}, \quad pE_\nu - \frac{1}{w}\frac{\partial p}{\partial \nu}\Big|_{\partial\Omega_n} = 0, \quad \varphi|_{\partial\Omega_n} = 0, \tag{2}$$

$$pE_\nu - \frac{1}{w}\frac{\partial p}{\partial \nu} - 2\gamma_p p\Big|_{\partial\Omega_p} = \frac{J}{eD_p N_i}, \quad nE_\nu + \frac{1}{w}\frac{\partial n}{\partial \nu}\Big|_{\partial\Omega_p} = 0, \quad \varphi|_{\partial\Omega_p} = U,$$

© БОМБА А.Я, МОРОЗ І.П., 2021

$$nE_\nu + \frac{1}{w}\frac{\partial n}{\partial \nu} - 2\gamma_n^* n\Big|_{\partial\Omega_{00}} = 0, \quad pE_\nu - \frac{1}{w}\frac{\partial p}{\partial \nu} - 2\gamma_p^* p\Big|_{\partial\Omega_{00}} = 0,$$

$$-nE_\nu - \frac{1}{w}\frac{\partial n}{\partial \nu} - 2\gamma_n^* n\Big|_{\partial\Omega_{01}} = 0, \quad -pE_\nu + \frac{1}{w}\frac{\partial p}{\partial \nu} - 2\gamma_p^* p\Big|_{\partial\Omega_{01}} = 0, \quad \frac{\partial\varphi}{\partial\nu}\Big|_{\partial\Omega_0} = 0,$$

де $\vec{E} = -\nabla\varphi$, $\gamma_n = \dfrac{\alpha_n}{2D_n}$, $\gamma_p = \dfrac{\alpha_p}{2D_p}$, $\alpha_{n,p}$ - швидкості поверхневої рекомбінації відповідно електронів і дірок, U – напруга на контактах структури, $\vec{\nu}$ - вектор нормалі до поверхні. На межах області $\partial\Omega_{02}$, $\partial\Omega_{03}$ використовується умови періодичності *p-i-n* структури.

Нелінійна система диференціальних рівнянь в частинних похідних (1) є сингулярно збуреною. Скористаємось асимптотичними методами [4-9] для пошуку розв'язків відповідної задачі.

Побудова асимптотичного розв'язку крайової задачі (1)-(2)

Беручи до уваги те, що на структуру розв'язку в основному впливають умови на контактних ділянках $\partial\Omega_n$ та $\partial\Omega_p$, через які здійснюється наповнення *i*-області носіями заряду, пропонуємо шукати розв'язок у вигляді наступних асимптотичних рядів (за степенями малого параметра μ):

$$\begin{pmatrix}\varphi\\ n\\ p\end{pmatrix} = \begin{pmatrix}\varphi(x,y,\mu)\\ n(x,y,\mu)\\ p(x,y,\mu)\end{pmatrix} = \begin{pmatrix}\Phi_{(m)}(x,y,\mu)\\ N_{(m)}(x,y,\mu)\\ P_{(m)}(x,y,\mu)\end{pmatrix} + \begin{pmatrix}\underline{\Phi}_{(m)}(\underline{\xi},\mu)\\ \underline{N}_{(m)}(\underline{\xi},\mu)\\ \underline{P}_{(m)}(\underline{\xi},\mu)\end{pmatrix} + \begin{pmatrix}\overline{\Phi}_{(m)}(\overline{\xi},\mu)\\ \overline{N}_{(m)}(\overline{\xi},\mu)\\ \overline{P}_{(m)}(\overline{\xi},\mu)\end{pmatrix} + \begin{pmatrix}R_{\varphi(m)}(x,y,\mu)\\ R_{n(m)}(x,y,\mu)\\ R_{p(m)}(x,y,\mu)\end{pmatrix} = \tag{3}$$

$$= \begin{pmatrix}\sum_{i=0}^{m}\mu^i\varphi_i(x,y)\\ \sum_{i=0}^{m}\mu^i n_i(x,y)\\ \sum_{i=0}^{m}\mu^i p_i(x,y)\end{pmatrix} + \begin{pmatrix}\sum_{i=0}^{m}\mu^i\underline{\Phi}_i(\underline{\xi})\\ \sum_{i=0}^{m}\mu^i\underline{N}_i(\underline{\xi})\\ \sum_{i=0}^{m}\mu^i\underline{P}_i(\underline{\xi})\end{pmatrix} + \begin{pmatrix}\sum_{i=0}^{m}\mu^i\overline{\Phi}_i(\overline{\xi})\\ \sum_{i=0}^{m}\mu^i\overline{N}_i(\overline{\xi})\\ \sum_{i=0}^{m}\mu^i\overline{P}_i(\overline{\xi})\end{pmatrix} + \begin{pmatrix}R_{\varphi(m)}(x,y,\mu)\\ R_{n(m)}(x,y,\mu)\\ R_{p(m)}(x,y,\mu)\end{pmatrix},$$

де $\Phi_{(m)}, N_{(m)}, P_{(m)}$ - регулярні частини асимптотик; $\underline{\Phi}_{(m)}, \underline{N}_{(m)}$, $\underline{P}_{(m)}$, $\overline{\Phi}_{(m)}$, $\overline{N}_{(m)}$, $\overline{P}_{(m)}$ - примежові поправки асимптотик відповідно в околах точок $y=0$, $0 \le x \le s/2w$ і $y=0$, $\dfrac{s}{2w} \le x \le 1-\dfrac{s}{2w}$ ($\underline{\xi} = \dfrac{y}{\sqrt{\mu}}$ - регуляризуючий розтяг); $R_{\varphi(m)}, R_{n(m)}, R_{p(m)}$ - залишкові члени.

Підставляючи (3) у рівняння системи (1) та умови (2) та застосувавши стандартну процедуру "прирівнювання" [4-7] формуємо послідовність задач для визначення *i*-х компонент асимптотичних рядів. Головну увагу зосереджуємо на пошуку головних членів регулярних рядів (3), оскільки ці функції містять основну інформацію про поведінку плазми. Отримуємо задачі:

$$\begin{cases} n_0 = p_0, \\ \Delta n_0 - \nabla(n_0\nabla\varphi_0) - A_n n_0 = 0, \\ \Delta p_0 + \nabla(p_0\nabla\varphi_0) - A_p p_0 = 0. \end{cases} \qquad \begin{cases} \dfrac{\partial^2\underline{\Phi}_0}{\partial\underline{\xi}^2} = -(\underline{P}_0 - \underline{N}_0), \\ \dfrac{\partial^2\underline{N}_0}{\partial\underline{\xi}^2} - \dfrac{\partial}{\partial\underline{\xi}}\left(\underline{N}_0\dfrac{\partial\underline{\Phi}_0}{\partial\underline{\xi}}\right) = 0, \\ \dfrac{\partial^2\underline{P}_0}{\partial\underline{\xi}^2} + \dfrac{\partial}{\partial\underline{\xi}}\left(\underline{P}_0\dfrac{\partial\underline{\Phi}_0}{\partial\underline{\xi}}\right) = 0. \end{cases} \tag{4), (5}$$

Для визначення $\overline{\Phi}_0(\overline{\xi})$, $\overline{N}_0(\overline{\xi})$, $\overline{P}_0(\overline{\xi})$ використовуємо систему рівнянь аналогічну (5).

При цьому граничні умови (2) для системи рівнянь (4)-(5) набувають наступного виду:

$$\frac{\partial n_0}{\partial y}-\gamma_n w n_0-\gamma_n w\underline{N}_0\Big|_{y=0}=-\frac{J}{2eD_n}\frac{w}{N_i}\ \text{при}\ 0\le x\le\frac{s}{2w},\ \ \varphi_0+\underline{\Phi}_0\big|_{0\le x\le\frac{s}{2w}}^{y=0}=0, \tag{6}$$

$$\frac{\partial n_0}{\partial y}+\gamma_p w n_0+\gamma_p w\overline{P}_0\Big|_{y=0}=-\frac{J}{2eD_p}\frac{w}{N_i}\ \text{при}\ 1-\frac{s}{2w}\le x\le 1,\ \varphi_0+\overline{\Phi}_0\big|_{1-\frac{s}{2w}\le x\le 1}^{y=0}=U,$$

$$\frac{\partial n_0}{\partial y}-\beta w n_0\Big|_{y=0}=0\ \text{при}\ \frac{s}{2w}\le x\le 1-\frac{s}{2w},\ \ \frac{\partial n_0}{\partial y}+\beta w n_0\Big|_{y=\frac{l}{w}}=0,\ \frac{\partial\varphi_0}{\partial y}\Big|_{y=0,y=\frac{l}{w}}=0,$$

$$n_0(0,y)=n_0(x\pm 2i,y),\quad \frac{\partial\underline{N}_0}{\partial\underline{\xi}}\Big|_{\underline{\xi}=0}=0,\quad \frac{\partial\underline{P}_0}{\partial\underline{\xi}}\Big|_{\underline{\xi}=0}=0,\quad \frac{\partial\overline{P}_0}{\partial\overline{\xi}}\Big|_{\overline{\xi}=0}=0,\quad \frac{\partial\overline{N}_0}{\partial\overline{\xi}}\Big|_{\overline{\xi}=0}=0.$$

Також використовуємо характерні для примежових функцій додаткові умови [4-7]:

$$\lim_{\underline{\xi}\to\infty}\underline{\Phi}_0(\underline{\xi})=\lim_{\underline{\xi}\to\infty}\underline{N}_0(\underline{\xi})=\lim_{\underline{\xi}\to\infty}\underline{P}_0(\underline{\xi})=\lim_{\overline{\xi}\to\infty}\overline{\Phi}_0(\overline{\xi})=\lim_{\overline{\xi}\to\infty}\overline{N}_0(\overline{\xi})=\lim_{\overline{\xi}\to\infty}\overline{P}_0(\overline{\xi})=0.$$

Зауважимо, що система (4) приводиться до лінійного диференціального рівняння виду:

$$\Delta n_0-\frac{1}{2}(A_n+A_p)n_0=0, \tag{7}$$

яке також отримують і в рамках застосування наближення амбіполярної дифузії [1,3].

Встановлюємо оцінки залишкових членів асимптотичних рядів як і в [4-7]:

$$R_{\varphi(m)}(x,y,\mu)=O(\mu^{m+1}),\ \ R_{n(m)}(x,y,\mu)=O(\mu^{m+1}),\ \ R_{p(m)}(x,y,\mu)=O(\mu^{m+1}).$$

Загальний розв'язок (7) з відповідними граничними умовами (6) шукаємо у вигляді ряду:

$$n_0(x,y)=\sum_{i=0}^{M-1}(C_{1i}\exp(-\eta_i y)+C_{2i}\exp(\eta_i y))\cos\alpha_i x,\quad \text{де}\ \alpha_i=\pi i. \tag{8}$$

Сталі C_{1i} , C_{2i} визначаються в результаті розв'язання системи лінійних алгебраїчних рівнянь, що отримана на основі застосування умов (6) на межах області Ω при $y=0$ та $y=l/w$ та проектування на систему базисних функцій $\cos\alpha_j x$:

$$\int_0^1\sum_{i=0}^{M-1}(-\eta_i C_{1i}+\eta_i C_{2i})\cos\alpha_i x\cos\alpha_j x dx=\int_0^{\frac{s}{2w}}\left(-\frac{J}{2eD_n}\frac{w}{N_i}+\gamma_n w\tilde{\Pi}_\infty^2\right)\cos\alpha_j x dx+$$

$$+\gamma_n w\int_0^{\frac{s}{2w}}\sum_{i=0}^{M-1}(C_{1i}+C_{2i})\cos\alpha_i x\cos\alpha_j x dx+\beta w\int_{\frac{s}{2w}}^{1-\frac{s}{2w}}\sum_{i=0}^{M-1}(C_{1i}+C_{2i})\cos\alpha_i x\cos\alpha_j x dx-$$

$$-\int_{1-\frac{s}{2w}}^1\left(\frac{J}{2eD_p}\frac{w}{N_i}+\gamma_p w\tilde{\Pi}_\infty^2\right)\cos\alpha_j x dx-\gamma_p w\int_{1-\frac{s}{2w}}^1\sum_{i=0}^{M-1}(C_{1i}+C_{2i})\cos\alpha_i x\cos\alpha_j x dx,$$

$$C_{2i}=\frac{(\eta_i-\beta w)}{(\eta_i+\beta w)}\exp(-2\eta_i)C_{1i},\quad j=\overline{0,(M-1)}. \tag{9}$$

Примежові поправки знаходимо шляхом розв'язання задачі (5)-(6) із використанням підходів, що описані у роботі [9]. Результат отримуємо у наступному вигляді:

$$\underline{P}_0=0,\quad \underline{N}_0(\underline{\xi})=\Pi_\infty{}^2\left(1-th^2\left(\frac{\Pi_\infty\underline{\xi}}{2}\right)\right),\ \overline{P}_0(\overline{\xi})=\Pi_\infty{}^2\left(1-th^2\left(\frac{\Pi_\infty\overline{\xi}}{2}\right)\right),\quad \overline{N}_0=0. \tag{10}$$

Для встановлення значення сталої інтегрування Π_∞ отримано трансцендентне рівняння виду (аналогічно [9]): $\int_{\frac{s}{2w}}^{1-\frac{s}{2w}}\left(th\left(\frac{\Pi_\infty t}{2\sqrt{\mu}}\right)-th\left(\frac{\Pi_\infty(1-t)}{2\sqrt{\mu}}\right)\right)dt=\frac{U}{\Pi_\infty}$.

Таким чином, розподіл концентрації електронно-діркової плазми в активній області інтегральної поверхнево-орієнтованої *p-i-n* структури визначається функцією виду:

$$n(x,y)=\sum_{i=0}^{M-1}\left(C_{1i}\exp(-\eta_i y)+C_{2i}\exp(\eta_i y)\right)\sin\alpha_i x+$$
$$+\begin{cases}\Pi_\infty{}^2\left(1-th^2\left(\dfrac{\Pi_\infty y}{2\sqrt{\mu}}\right)\right), & 0\le x\le\dfrac{s}{2w},1-\dfrac{s}{2w}\le x\le 1\\ 0, & \dfrac{s}{2w}<x<1-\dfrac{s}{2w}\end{cases}.$$

Висновки

Побудовано математичну модель стаціонарного розподілу електронно-діркової плазми в активній області інтегральних поверхнево-орієнтованих *p-i-n* структур у вигляді нелінійної сингулярно збуреної крайової задачі для системи нелінійних рівнянь неперервності струму носіїв заряду та Пуассона. Знайдено наближений розв'язок відповідної крайової задачі у вигляді головних членів асимптотичних рядів за степенями малого параметра. Показано, що класичні постановки задач моделювання характеристик *p-i-n* структур [1-4] автоматично включаються у рамки запропонованої схеми пошуку розв'язків поставленої задачі, а представлений метод дозволяє внести суттєві поправки у розв'язок.

ЛІТЕРАТУРА

1. Sze S., Kwok K. Physics of Semiconductor Devices. New York:Wiley-Interscience, 2006. 815 p.
2. Grimalsky V.V., Kishenko Ya.I., Koshevaya S.V., Moroz I.P. The Interaction of Powerful Electromagnetic Waves With Integrated p-i-n-structures. *Doc. of Int. Symp. "Physics and Engineering of Mm and Submm Waves", June 7-10, 1994.* Kharkov, Ukraine. Vol.1. P. 238-239.
3. Адирович Э. И., Карагеоргий-Алкалаев П. М., Лейдерман А. Ю. Токи двойной инжекции в полупроводниках. Под ред. Гальперина. М.: Советское радио, 1978. 320 с.
4. Тихонов А.Н. Системы дифференциальных уравнений, содержащие малые параметры. *Математический сборник.* 1952. т. 31(73). № 3. С. 575-586.
5. Васильева А.Б., Бутузов В.Ф. Асимптотические методы в теории сингулярных возмущений. М.: Высш. шк., 1990. 208 с.
6. Бомба А.Я., Присяжнюк І.М., Присяжнюк О.В. Методи теорії збурень прогнозування процесів тепломасоперенесення в пористих та мікропористих середовищах. Рівне: О.Зень, 2017. 291 с.
7. Бомба А.Я. Про асимптотичний метод наближеного розв'язання однієї задачі масопереносу при фільтрації в пористому середовищі. *Український математичний журнал.* 1982. Т.34. № 4. С.37-40.
8. Васильева А.Б., Стельмах В.Г. Сингулярно возмущенные системы теории полупроводниковых приборов. *Вычислительная математика и математическая физика.* 1977. Т.17. №2. С.339-348.
9. Бомба А.Я., Мороз І.П. Чисельно-асимптотичний метод розв'язання сингулярно збурених модельних задач про стаціонарний розподіл носіїв заряду в активній області p-i-n-діодів. *Вісник Національного університету водного господарства та природокористування. Серія: Технічні науки.* Рівне. 2020. №1. С. 14-25.

БОМБА Андрій Ярославович – д.т.н.,проф.; професор кафедри комп'ютерних наук та прикладної математики; Національний університет водного господарства та природокористування, Рівне, Україна; e-mail: abomba@ukr.net; ORCID: 0000-0001-5528-4192.

Наукові інтереси: *методи теорії збурень і комплексного аналізу моделювання керовних процесів та систем за умов ідентифікації та оптимізації параметрів.*

МОРОЗ Ігор Петрович – к.ф.-м.н., доц.; докторант кафедри комп'ютерних наук та прикладної математики; Національний університет водного господарства та природокористування, Рівне, Україна; e-mail: igor_moroz@yahoo.com; ORCID: 0000-0001-6381-2266.

Наукові інтереси: *математичне моделювання фізичних процесів у елементах техніки надвисоких частот.*

УДК 004.93 **СЕКЦІЯ 7**

БОНДАРЕНКО Є.В.

МЕТОД ВІДНОВЛЕННЯ ДАНИХ В АВТОМАТИЗОВАНИХ СИСТЕМАХ КОНТРОЛЮ ТА ДІАГНОСТИКИ СТАНУ ОБ'ЄКТІВ УПРАВЛІННЯ

Чимало технічних галузей, що мають відношення до отримання, обробки, зберігання та передачі інформації, в значній мірі орієнтуються на розвиток систем, в яких інформація має характер зображень. Обробка зображень дозволяє вирішувати широке коло прикладних задач, окремим прикладом яких є відновлення втрачених або пошкоджених ділянок зображень.

В автоматизованих системах, в яких інформація має характер зображень, пошкодження або втрата ділянок може привести до зниження ефективності і стабільності роботи. В даному випадку найбільш доцільним було б використання пакетної обробки зображень, яка дозволяє мінімізувати взаємодії з користувачем, але вимагає високоефективних алгоритмів високої якості. Для забезпечення даного підходу варто розглянути методи автоматизації процесу відновлення загублених або пошкоджених ділянок з використанням глибокого навчання.

Глибоке навчання - це алгоритми машинного навчання для моделювання високорівневих абстракцій із застосуванням численних нелінійних перетворень. Глибоке навчання дозволяє навчати модель передбачати результат по набору вхідних даних. Для навчання мережі можна використовувати як контрольоване, так і неконтрольоване навчання.

Глибоке навчання характеризується як клас алгоритмів машинного навчання, яке:

- використовує багатошарову систему нелінійних фільтрів для вилучення ознак з перетвореннями. Кожен наступний шар отримує на вході вихідні дані попереднього шару. Система глибокого навчання може поєднувати алгоритми навчання з вчителем і без вчителя, при цьому аналіз зразка являє собою навчання без вчителя, а класифікація - навчання з вчителем;
- володіє декількома шарами виявлення ознак або параметрів уявлення даних (навчання без вчителя). При цьому ознаки організовані ієрархічно, а ознаки більш високого рівня є похідними від ознак нижчого рівня;
- формує в процесі навчання шари на декількох рівнях представлень, які відповідають різним рівням абстракції; шари утворюють ієрархію понять;

Всі визначення констатують:

1. наявність декількох шарів нелінійної обробки;
2. навчання з вчителем або без вчителя ознак кожного шару, формуючи ієрархію від низького до високого рівня;

В першу чергу до глибокого навчання відносяться такі методи і їх варіації:

- певні системи навчання без вчителя, такі як обмежена машина Больцмана для попереднього навчання, автоенкодер, глибока мережа довіри, генеративно-змагальна мережа;
- певні системи навчання з вчителем, такі як згорткова нейронна мережа, яка вивела на новий рівень технології розпізнавання образів;
- рекурентні нейронні мережі, що дозволяють навчатися на процесах в часі,
- рекурсивні нейронні мережі, що дозволяють включати зворотний зв'язок між елементами схеми і ланцюгами.

Постановка задачі дослідження

Постановкою задачі є дослідження роботи автоенкодера як конкретної найпростішої архітектури глибокого навчання.

Аналітичний огляд моделей та методів, згідно поставленої задачі

Автоенкодер - нейронна мережа, яка копіює вхідні дані на вихід. За архітектурою схожий на персептрон. Автоенкодери стискають вхідні дані для представлення їх в прихованому просторі(latent-space), а потім відновлюють з цього представлення вихідні дані (output). Мета - отримати на вихідному шарі зображення, найбільш близьке до вхідного.

© БОНДАРЕНКО Є.В., 2021

Відмінна особливість автоенкодерів - кількість нейронів на вході та на виході збігається.

Автоенкодер складається з двох частин:

• енкодер - відповідає за стиснення входу в прихованому просторі. Представлений функцією кодування $h = f(x)$;

• декодер - призначений для відновлення введення з прихованого простору. Представлений функцією декодування $h = f(x)$.

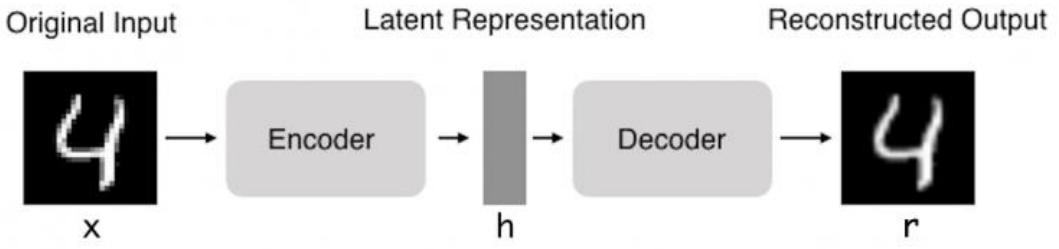


Рис. 1 Архітектура автоенкодера

Таким чином, автоенкодер описують функцією$g\big(f(x)\big) = r$, де *r* збігається з початковим *x* на вході.

Де застосовуються автоенкодери?

Два основних практичних застосування автоенкодерів для візуалізації даних:

• згладжування шуму;

• зниження розмірності;

З відповідними обмеженнями по розмірності і розрідженості автоенкодери можуть вивчати data projections, які більш цікаві, ніж PCA (метод головних компонент) або інші базові техніки.

Автоенкодери навчені як зберігати інформацію, так і надавати новим представленням різні властивості. Для цього використовують різні типи автоенкодерів:

• Undercomplete Autoencoder - має менший розмір прихованого шару в порівнянні з вхідним шаром. Це допомагає виділити залежності із даних. Цей автоенкодер мінімізує функцію втрат, штрафуючи g(f(x)) за відмінності від вхідних даних *x*. Не потребує регуляризації, оскільки максимізує ймовірність даних, а не копіює вхідні дані у вихідні.

$$L = |x - f(g(x))|$$

• Sparse Autoencoder - розмірність прихованого шару більше, ніж вхідного. На прихований шар накладається штраф за розрідженість $\Omega(h)$, значення, близьке до нуля, на додаток до помилки відновлення. Це запобігає перенавчанню.

$$L = |x - f(g(x))| + \Omega(h)$$

• Denoising Autoencoder (DAE) - створюється пошкоджена копія введення, яка вносить певний шум. Це допомагає навчити прибирати шум з даних. DAE мінімізує функцію втрат між вихідним вузлом і пошкодженим вводом.

$$L = |x' - f(g(x))|$$

• Contractive Autoencoder(CAE) - метою CAE є отримання надійного представлення даних, яке менш чутливе до невеликих змін. Надійність представлення даних досягається шляхом регуляризації. Покарання в CAE це норма Фробеніуса матриці Якобі, яка обчислюється для прихованого шару щодо вхідних даних. Фробеніусовською нормою матриці Якобі є сума квадратів всіх елементів.

• Stacked Denoising Autoencoder - це нейронна мережа з декількома шарами Sparse Autoencoder. Коли в автоенкодер додають більше прихованих шарів, це допомагає зменшити об'ємний код, отриманий кодувальником.

• Deep Autoencoder- складається з двох ідентичних глибоких мереж. Одна мережа для кодування, інша для декодування. Зазвичай мають від 4 до 5 рівнів для кодування і наступні від 4 до 5 рівнів для декодування.

• Variational Autoencoder(VAE) - має одну унікальну властивість, яка відрізняє його від стандартного автоенкодера. Саме ця властивість робить варіаційні автоенкодери настільки корисними при генерації даних: їх прихований простір для побудови є безперервним, дозволяючи виконувати випадкові перетворення та інтерполяцію.

Безперервність прихованого простору досягається несподіваним способом: енкодер видає не один вектор розміру n, а два вектора розміру n - вектор середніх значень μ і вектор стандартних відхилень σ.

Варіаційні автоенкодери формують параметри вектора довжини n з випадкових величин X_i, причому i - елементи векторів μ і σ є середнім і стандартним відхиленням i -ї випадкової величини X_i. Разом ці величини утворюють n -мірний випадковий вектор, який надсилається на декодер для відновлення даних. Ця так звана стохастична генерація означає, що навіть для однакових вхідних даних результат кодування буде різним внаслідок випадковості вибору вектора кодування.

Формалізація та представлення вихідних даних

База даних MNIST (Modified National Institute of Standards and Technology) - об'ємна база даних зразків рукописного написання цифр. База даних є стандартом, запропонованим Національним інститутом стандартів і технологій США з метою калібрації і зіставлення методів розпізнавання зображень за допомогою машинного навчання в першу чергу на основі нейронних мереж. Дані складаються з заздалегідь підготовлених прикладів зображень, на основі яких проводиться навчання та тестування систем. База даних була створена після переробки оригінального набору чорно-білих зразків розміром 20x20 пікселів NIST

Творці бази даних NIST, в свою чергу, використовували набір зразків з Бюро перепису населення США, до якого були додані ще тестові зразки, написані студентами американських університетів. Зразки з набору NIST були нормалізовані, пройшли згладжування і приведені до розміру 28x28 пікселів.

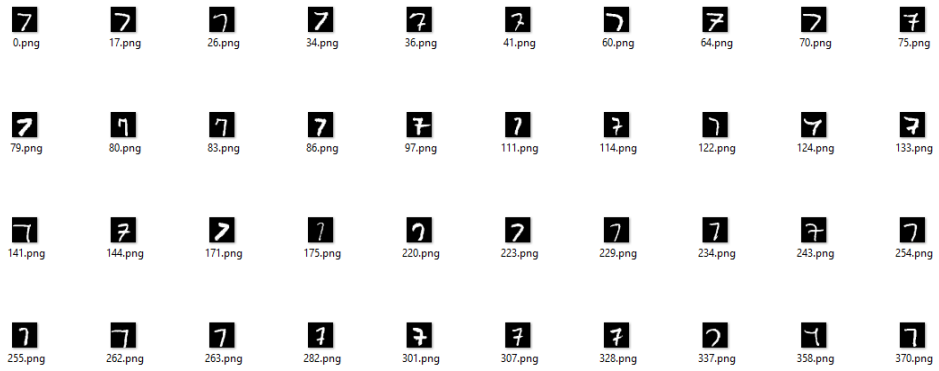


Рис. 2. Фрагмент набору NIST для тестування

Опис обраного методу розв'язання задачі

Властивістю автоенкодерів є отримуваня прихованої параметричної моделі даних, а математичні наслідки з цього дають можливість зв'язати їх з Байєсовськими методами прийняття рішення. Навчаючий набір даних визначається щільністю їх розподілу. Чим вище щільність навчальних прикладів навколо локальної точки у вхідному просторі, тим краще автоенкодер реконструює вхідний вектор в цьому місці простору. Крім того, всередині автоенкодера є вектор латентного представлення вхідних даних (як правило, низької розмірності) і, якщо дані проектуються в латентному просторі в область, що раніше не була задіяна при навчанні, то, значить, і в навчальній вибірці не було нічого схожого.

Опис обраного обчислювального алгоритму розв'язання задачі

У випадку автоенкодера на мережу накладається додаткове обмеження - наявність «вузького горла» в одному з проміжних (прихованих) шарів. Важливим критерієм є те що до

"вузького горла" відбувається навчання за допомогою фільтра Хебба, а після - навчання методом зворотного поширення помилки.

Результати обчислень

Нижче наведено результат роботи простого автоенкодера, який використовує набір MNIST в якості формалізація та представлення вихідних даних, а також результати роботи за допомогою згорткового автоенкодера:

Рис. 3. Набір MNIST,верхній ряд- вихідні цифри, нижній ряд - відновлені цифри

Рис. 4. Вхід і вихід при згорткованому автоенкодері

Дивлячись на рисунок, можна побачити, що втрачено деякі важливі деталі в цьому базовому прикладі. Оскільки наші вхідні дані є зображеннями, має сенс використовувати згорткові нейронні мережі в якості кодеров і декодерів. В практичних умовах автоенкодери, застосовані до зображення, завжди є згортковими автоенкодерами, оскільки вони просто працюють набагато краще.

ЛІТЕРАТУРА

1. Саймон Хайкін. Нейронні мережі: повний курс, 2-е видання, 2006. - с. 95-104,523-530,537-543.
2. Hongyu Shen, Daniel George, E. A. Huerta and Zhizhen Zhao, " Denoising Gravitational Waves using Deep Learning with Recurrent Denoising Autoencoders ", 2012 IEEE International Work on (IEEE, 2012) pp. 1-6.
3. https://habr.com/ru/post/198268/
4. https://neurohive.io/ru/osnovy-data-science/avtojenkoder-tipy-arhitektur-i-primenenie/
5. https://neurohive.io/ru/osnovy-data-science/glubokoe-obuchenie-deep-learning-kratkij-tutorial/

БОНДАРЕНКО Євгеній Володимирович – студент; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: evgeniy.bondarenko1999@gmail.com; ORCID 0000-0001-7592-4783.

Наукові інтереси:

– методи машинного навчання.

УДК 004.942 **СЕКЦІЯ 2**

БРАТУСЬ М.А.

ІМОВІРНІСНІ МЕТОДИ ПОБУДОВИ ПРОБНИХ ВИБІРОК ПО ЗАДАНОМУ ЗАКОНУ РОЗПОДІЛУ

Формування вихідних тестових вибірок даних є однією з головних проблем, що виникають у момент побудови алгоритмів стохастичної оптимізації. Враховуючи зростаючу тенденцію до використання алгоритмів аналізу даних, які ґрунтуються на універсальній структурі, при рішенні складних практичних задач стохастичної оптимізації, етап підготовки та організації тестових вибірок даних є достатньо важливим та потребує чітких злагоджених дій за мінімально короткий проміжок часу. Звертаючи увагу на масштабність застосування машинного навчання у різних сферах життя сучасної людини, динаміку ускладнення вирішуваних задач в області аналізу даних, створення систем штучного інтелекту, потреби в ефективності, якості, чіткості будуть рости.

Також, ґрунтуючись на приналежності тестових вибірок даних до різних класів застосування їх отримання утруднено з об'єктивних причин, у першу чергу це практичне застосування вибірок в експериментальних дослідженнях. При проведенні подібних досліджень в медицині часто об'єктивно присутні обмеження, пов'язані з необхідністю отримання тестових даних для діагностики захворювань; при цьому підготовка таких даних може займати місяці і роки.

До даної проблеми має також пряме відношення проблема незбалансованості навчальної вибірки в задачах класифікації. Дана проблема виникає в ситуаціях нерівномірного розподілу різних класів, коли деякі класи образів представлені істотно меншими за обсягом даними, ніж інші.

Звідси виникає потреба в дослідженні, порівнянні та аналізі імовірнісних методів генерації пробних вибірок, що мають заданий розподіл.

В умовах сьогодення, наукові дослідження у сфері імовірнісні методи генерації пробних вибірок, що мають заданий розподіл є доволі масштабними. Автори підходять до розгляду теми як з теоретичного так і з практичного аспектів, формуючи коло вирішених задач та спонукаючи до появи нових.

Постановка задачі дослідження

Аналітично усічений нормальний розподіл з використання методу Латинського Гіперкуба, згенерувати багатовимірну вибірку репрезентативна заданому закону розподілу.

Аналітичний огляд моделей та методів, згідно поставленої задачі

Метод латинського гіперкуба

Вибірка латинського гіперкуба - це статистичний метод генерації майже випадкової вибірки значень параметрів з багатовимірного розподілу. Метод вибірки часто використовується для побудови комп'ютерних експериментів або для інтеграції Монте-Карло.

При вибірці функції змінних діапазон кожної змінної ділиться на рівноімовірні інтервали. потім розміщуються точки вибірки, щоб задовольнити вимогам латинського гіперкуба; це змушує кількість поділів, бути рівним для кожної змінної. Ця схема вибірки не вимагає більшої кількості вибірок для більшої кількості вимірів (змінних); ця незалежність - одне з головних переваг даної схеми вибірки. Ще одна перевага полягає в тому, що випадкові проби можна брати по одній, запам'ятовуючи, які проби були взяті на даний момент.

У двох розмірах, відмінність між випадковою вибіркою, латинської вибіркою гіперкуби і ортогональної вибіркою може бути пояснено таким чином:

- У **випадкових** нових типових пунктах зроблені, не беручи до уваги раніше вироблені типові пункти. Потрібно не обов'язково знати заздалегідь, скільки типових пунктів необхідно.

© БРАТУСЬ М.А., 2021

- У **латинському гіперкуби**, потрібно спочатку вирішити, скільки зразок вказує на використання, і для кожного типового пункту пам'ятають, в якому ряді і колонці був узятий типовий пункт.
- У **ортогональній вибірці** типовий простір поділений на однакові ймовірні підмести. Всі типові пункти тоді обрані, одночасно переконавшись, що повний ансамбль типових пунктів - латинський зразок гіперкуби і що кожний підпростір вибрано з тією ж самою щільністю.

Таким чином **ортогональна вибірка** гарантує, що ансамбль випадкових чисел - дуже хороший представник реальної мінливості, **латинський гіперкуб** гарантує, що ансамбль випадкових чисел представлений для реальної мінливості, тоді як **традиційна випадкова вибірка** є просто ансамблем випадкових чисел без будь-яких гарантій.

В даному випадку всі області простору вибірки представлені вхідними значеннями і проводиться генерація вибірки розміру N із K змінних $x = [x_1 \dots x_k]^T$ з функцією густини розподілу ймовірності $f(x)$. Розподіл густини ймовірностей, випадкової величини, що цікавить, спочатку ділиться на N діапазонів з рівною ймовірністю $\frac{1}{N}$, а величина з вибірки кожного рівного діапазону ймовірності береться тільки один раз.

Для відбору існуючих допоміжних даних ми не можемо безпосередньо застосувати МЛГ(Метод Латинського Гіперкуба) до багатовимірного розподілу. Зразки відбираються звичайними МЛГ представляють собою комбінацію багатоваріантних варіантів, які можуть не існувати в реальному світі. На рисунку 1 показано ділянку розсіювання кумулятивної вірогідності даних висоти та схилу у вигляді сірих точок. Ми використовували алгоритм МЛГ, щоб зробити 10 зразків латинських гіперкубів. Отримані 10 зразків відображаються як кола на рис. 1, ми бачимо, що лише 4 з 10 зразки збігаються з точками даних, а інші 6 зразків не існують у реальному світі. Ця проблема буде більш виражена при роботі з великою кількістю змінних та зразкових сайтів.

Це стає проблемою оптимізації: дано N публікацій з допоміжними даними (X), виділяємо n зразкових публікацій 5N, так що вибіркові с публікацій x утворюють латинський гіперкуб, або багатоваріантний розподіл X максимально стратифіковано. Це вимагає процедури пошуку, в даній роботі ми пропонуємо МЛГ з пошуковим алгоритмом на основі евристичних правил у поєднанні з графіком відпалу. Цей алгоритм працює як на безперервних, так і на категоричних даних.

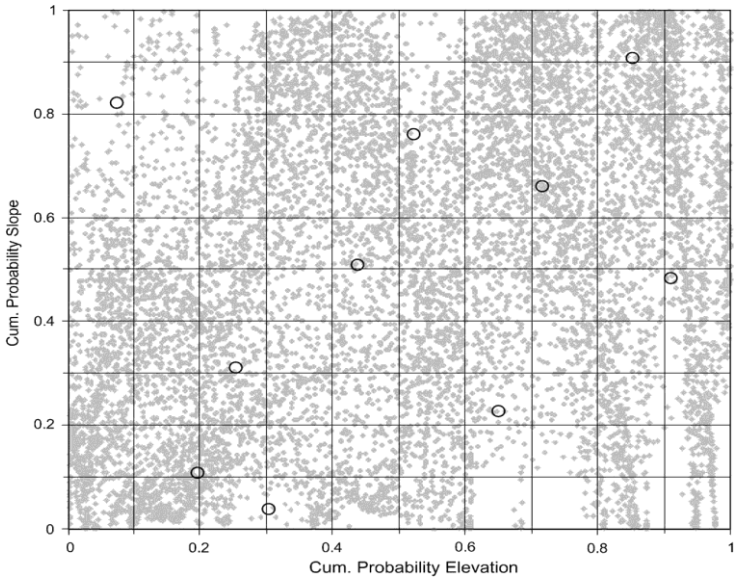


Рисунок - 1 Сірі точки представляють дані в області, а чорні кола представляють зразки, зроблені з використанням звичайних МЛГ.

Поки значення цільової функції не виходить за межі заданого критерію зупинки або для певного числа ітерацій.

Остаточний зразок буде представляти собою вірний або приблизний латинський гіперкуб функціонального простору, за якого збережеться розподіл та багатовимірна кореляція.

Усічений нормальний розподіл

При нормальному (гауссовому) розподілі випадкової величини вона може приймати будь-які значення від -∞ до + ∞. Оскільки можливі значення випадкової напрацювання до відмови Т можуть бути тільки позитивними, розподіл Т може бути лише усіченим нормальним.

Усіченим нормальним розподілом випадкової величини називається розподіл, що отримується з нормального при обмеженні інтервалу можливих значень цієї величини. Так як можливі значення випадкової величини Т обмежені інтервалом (t1, t2), то щільність усіченого розподілу

$$f(t) = cf(t)$$

Де

$$f(t) = \frac{1}{\sigma_t\sqrt{2\pi}} exp\left[-\frac{(t-m_t)^2}{2\sigma_t^2}\right]$$

f (t) - щільність неусіченого розподілу; с - нормуючий множник, що знаходяться за умови, що площа під кривою розподілу дорівнює одиниці:

$$c = \int_{t_1}^{t_2} f(t)dt = 1$$

Знайдемо формули для числових характеристик усіченого нормального розподілу: математичного очікування напрацювання до відмови усіченого нормального розподілу m_t, відкрита онлайн бібліотека та дисперсії напрацювання до відмови усіченого нормальний розподіл σ_t^2відкрита онлайн бібліотека. Згідно з визначеннями цих характеристик маємо:

$$m_t = \frac{c}{\sigma_t\sqrt{2\pi}}\int_{t_1}^{t_2} texp\left[-\frac{(t-m_t)^2}{2\sigma_t^2}\right]dt;$$

$$\sigma_t^2 = \frac{c}{\sigma_t\sqrt{2\pi}}\int_{t_1}^{t_2}(t-m_t)^2 exp\left[-\frac{(t-m_t)^2}{2\sigma_t^2}\right]dt;$$

Провівши перетворення, отримаємо:

$$m_t = m_t - \sigma_t k;$$

$$\sigma_t^2 = \sigma_t^2\left\{1 - k^2 - \frac{c}{\sqrt{2\pi}}\left[u_2 exp\left(-\frac{u_2^2}{2}\right) - u_1 exp\left(-\frac{u_1^2}{2}\right)\right]\right\},$$

Де

$$k = \frac{c}{\sqrt{2\pi}}\left[u_2 exp\left(-\frac{u_2^2}{2}\right) - u_1 exp\left(-\frac{u_1^2}{2}\right)\right]$$

На рис. 2 наведені залежності відносин числових характеристик усіченого і неусіченого нормального розподілів і значення нормує множники C0 від відносини mt / σt.

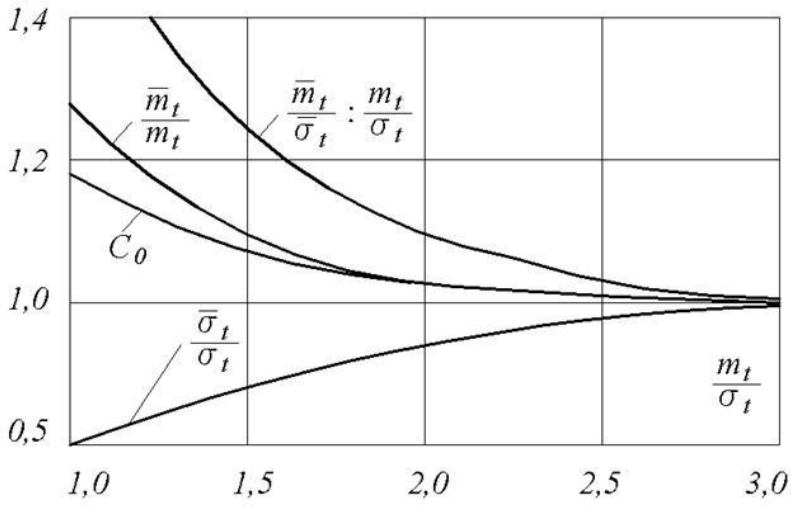


Рис. 2. Залежність відносин числових характеристик усіченого і неусіченого нормального розподілу і нормує множники C0 від відносини mt / σt. (Лінія над m_t , σt означає, що ці характеристики відносяться до усіченому розподілу)

Зважаючи на велику теоретичного і прикладного значення нормального розподілу його намагаються іноді застосувати і при явно несиметричних розподілах напрацювання до відмови. Для цього підбирають деяку функцію випадкової напрацювання до відмови, наприклад lg T, T2 тощо. Приблизно таку нормальному закону. Наприклад, досить часто використовується логарифмічно нормальний розподіл втомної довговічності, при якому передбачається, що логарифм числа циклів навантаження до руйнування зразка розподілений за нормальним законом.

ЛІТЕРАТУРА

1. The generalization of Latin hypercube sampling / Michael Shields, Jiaxin Zhang.
2. Latin hypercube sampling with inequality constraints. / Brendan Tracey, David H. Wolpert
3. Estimates of the coverage of parameter space by Latin Hypercube and Orthogonal sampling: connections between Populations of Models and Experimental Designs / Diane Donovan, Kevin Burrage, Pamela Burrage, Thomas A. Mccourt.

БРАТУСЬ Максим Андрійович – студент; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: xa11867822@student.karazin.ua; ORCID 0000-0002-3062-4258.

Наукові інтереси:

– *імовірнісні методи;*

УДК 004.7 **СЕКЦІЯ 2**

БУБЕР Д.И. ПАВЛОВ А.Н.

МЕТОДИКА РАСЧЕТА НАДЕЖНОСТИ ДЛЯ ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ НА БАЗЕ "ТОЛСТЫХ" КЛИЕНТОВ

Актуальность темы обусловлена тем, что в настоящее время сетевые технологии составляют основу большинства предприятий, которые требуют надежную и непрерывную работу сетевых технологий. Появление сбоев и отказов в технических и программных средствах локально-вычислительных системах способствуют нарушению работы пользователей системы, компонентов АСУ ТП, которые требуют подключения к ЛВС, а также нарушения в информационных процессах хранения данных. В свою очередь это приводит к большим убыткам на предприятии которые вызваны задержками в выполнении работ. Поэтому необходимо уделять достаточное внимание обеспечению надежности в работе ЛВС. Обеспечение надежности будет рассмотрена на примере клиент-серверной архитектуры на базе «толстого» клиента.

Клиент-серверная архитектура

Термин «клиент-серверная архитектура» – сборное понятие, состоящее из двух взаимодополняющих компонентов: сервера и, собственно, клиента.

Клиент – локальный компьютер на стороне виртуального пользователя, который выполняет отправку запроса к серверу для возможности предоставления данных или выполнения определенной группы системных действий.

Сервер – очень мощный компьютер или специальное системное оборудование, которое предназначается для разрешения определенного круга задач по процессу выполнения программных кодов. Он выполняет работы сервисного обслуживания по клиентским запросам, предоставляет пользователям доступ к определенным системным ресурсам, сохраняет данные или БД.

Рассмотрим клиент-серверную архитектуру, построенную на базе «толстого» клиента. Особенности такой модели заключаются в том, что пользователь отправляет определенный запрос на сервер, где тот системно обрабатывается и конечный результат отсылается клиенту (Рис 1). В возможности сервера входит одновременное обслуживание сразу нескольких клиентов.[1]

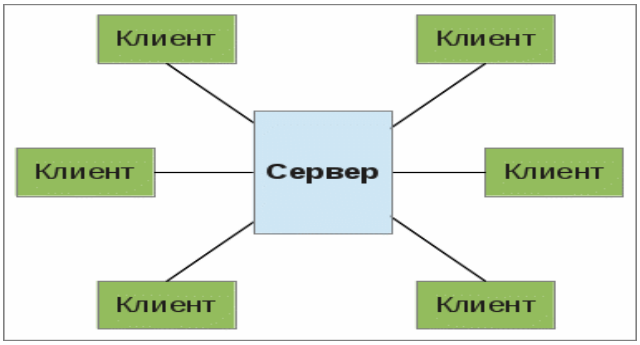


Рис. 1 Структура сети на основе выделенного сервера

Главная концепция архитектуры «клиент-сервер» заключается в разделении сетевого приложения на некоторое количество компонентов, каждый из которых реализует определённый набор сервисов. Эти компоненты могут выполняться на разных компьютерах, при этом выполняя серверные либо клиентские функции

За решение каждой задачи отвечает соответствующий компонент клиент-серверного приложения. Выделяют два основных вида клиент-серверных архитектур. Наиболее распространённой является двухзвенная архитектура (Рис. 2). Двухзвенной она называется из-за необходимости распределения трех базовых компонентов между двумя узлами (клиентом и сервером) [2]

© БУБЕР Д.И. ПАВЛОВ А.Н., 2021

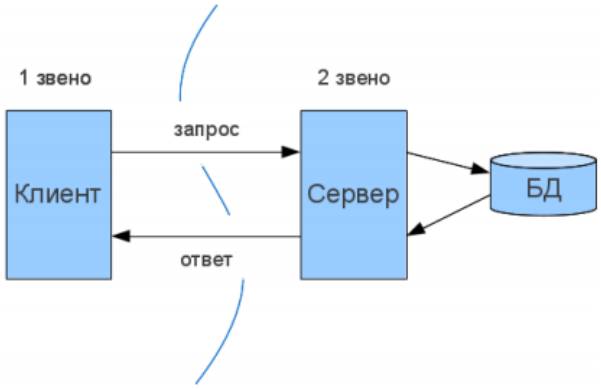


Рис. 2 Двухзвенная клиент-серверная архитектура

Расположение компонентов на стороне клиента или сервера определяет основные модели их взаимодействия в рамках двухзвенной архитектуры (Рис 3):

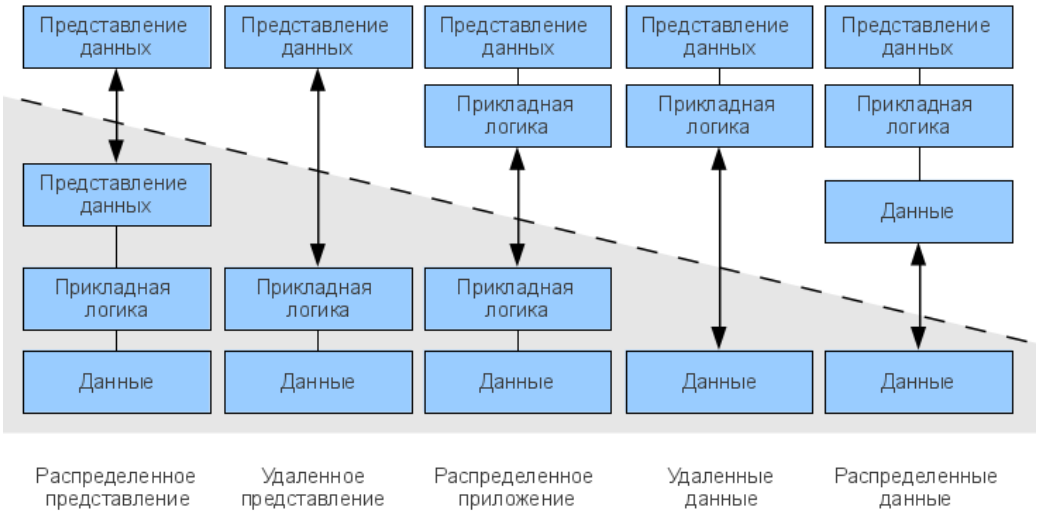


Рис. 3 Модели клиент-серверного взаимодействия

Постановка задачи

Для проведения анализа критерия надежности необходимо спроектировать схему ЛВС, состоящую из «толстых» клиентов, построенную по топологии «звезда». Для удобства расчетов достаточно смоделировать небольшую корпоративную сеть отдела, состоящую из 15 типичных устройств. Выработанные критерии отказов по схеме ЛВС позволят создать структурную схему расчета надежности и приступить к расчетам. В ходе расчетов необходимо определить коэффициенты готовности, наработки на отказ и времени восстановления. Первым этапом исследования является построение ЛВС. Допустим, что исследуемый отдел состоит из 12-и рабочих мест, используя соответствующее программное обеспечение (ПО). Таким образом, для построения ЛВС из «толстых клиентах» понадобятся 12 рабочих станций, сервер, принтер. Все устройства объединяются в сеть через коммутатор (см. рис. 4). Следующим этапом исследования является построение структурной схемы расчета надежности для разных уровней разукрупнения ЛВС по ОСТ 4Г 0.012.242-84 [2]. Для этого необходимо определить, неисправность каких элементов является критичной для работы системы в целом. Допустим, загрузка отдела позволяет оставить 2 рабочих места в резерве. Остальные 10 рабочих мест используются непрерывно в течение всего рабочего дня. Соответственно, отказ более, чем 2-ух рабочих станций/терминальных станций скажется на отказе системы в целом. Так же, недопустим отказ сервера, недопустим отказ коммутатора.

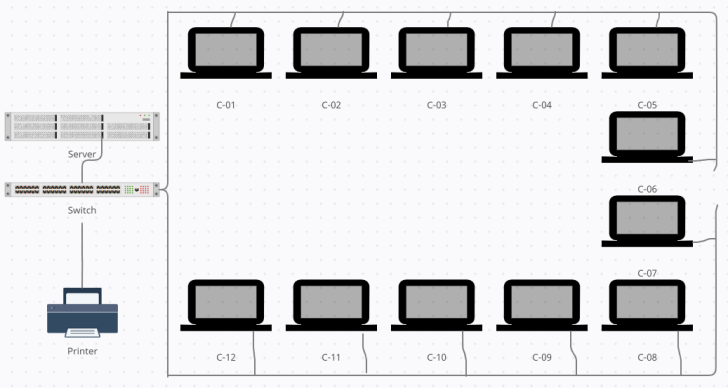


Рис. 4 Схема соединения устройств для ЛВС на базе «толстых» клиентов

Методика расчета надежности

Расчёт надежности ЛВС производится в два этапа. Сначала рассчитывается (определяется) надёжность элементов в отдельности, затем проводится расчёт надёжности ЛВС в целом. После чего, определяются комплекты запасных частей, инструментов, принадлежностей (ЗИП) для каждой из ЛВС. Схема последовательности действий при расчёте надёжности представлена на рис. 5.

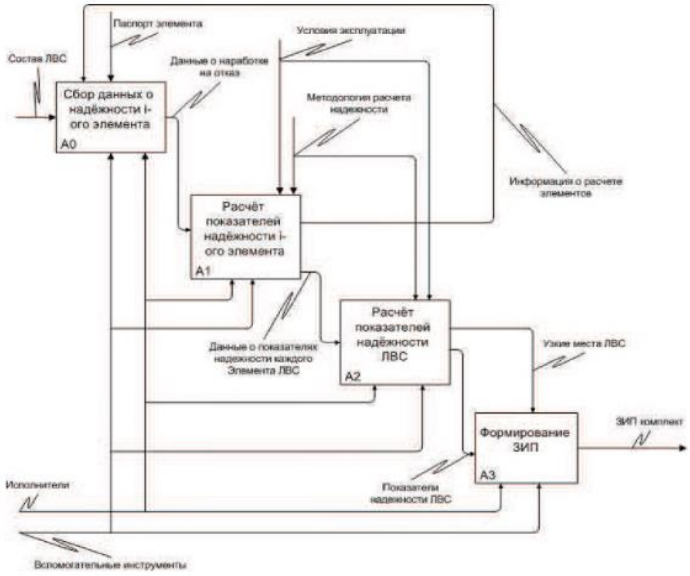


Рис. 5 Схема последовательности расчетов.

Расчёт наработки на отказ системы проводится по следующей формуле:

$$T_O = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{oi}}} \quad (1)$$

где T_{oi} – наработка на отказ i-го отдельного элемента.

Коэффициент готовности рассчитывается по следующей формуле:

$$K_r = \frac{T_O}{T_O + T_\text{В}} \quad (2)$$

Наработка на отказ совокупности рабочих станций, представляющих собой схему с нагруженным резервированием, рассчитывается по следующей формуле:

$$T_O = \frac{1}{\lambda} \frac{\sum_{i=0}^{m} C_N^i (\lambda T_\text{ВЭ})^i}{C_N^m (\lambda T_\text{ВЭ})^m} \quad (3)$$

где: m – количество резервных элементов, N – общее количество элементов, λ – интенсивность отказов, $T_\text{ВЭ}$ – время восстановления одного элемента:

при этом время восстановления такой системы рассчитывается по формуле:

$$T_{\text{ВЭ}} = \frac{T_{\text{ВЭ}}}{m+1} \quad (4)$$

Расчет наработки на отказ сервера и всей системы в целом приводится по формуле:

$$T_O = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{oi}}} \quad (5)$$

Пример расчета

Расчетные значения наработки на отказ для каждого устройства (элемента ЛВС) следующие:

В состав рабочей станции входят:

- монитор с показателем наработки на отказ 750000 часов,
- мышь и клавиатура по 100000 часов,
- системная плата 100000 часов,
- блок питания 100000 часов
- система охлаждения 55000 часов,
- процессор 550000 часов,
- оперативная память 500000 часов,
- видеокарта 55000 часов,
- жесткий диск 500000 часов.

Сервер от 45000 до 55000 часов.

Коммутатор 98690 часов.

Для подсчета времени восстановления достаточно взять оценочные (экспертные) данные. Так, например, на восстановление рабочей станции могут потребоваться целые сутки, с учетом диагностики, замены частей, перенастройки системы, либо полной переустановки системы. Для восстановления терминальной станции требуется существенно меньше времени из-за меньшего количество компонент и малого времени установки системы (допустим, 3 часа). На восстановление коммутатора требуется в среднем 1 час, на восстановление сервера 4 часа.

Рассчитаем значение наработки на отказ для одной рабочей станции с учетом наработки на отказ составных элементов:

$$T_O = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{oi}}} = 9643 \text{ часов}$$

Далее проводим расчет наработки на отказ для системы рабочих станций (10 основных, 2 резервных):

$$T_O = \frac{1}{\lambda} \frac{\sum_{i=0}^{m} C_N^i (\lambda T_{\text{ВЭ}})^i}{C_N^m (\lambda T_{\text{ВЭ}})^m} = 2.32 * 10^4$$

Время восстановления для такой системы рассчитываем следующим образом:

$$T_{\text{ВЭ}} = \frac{T_{\text{ВЭ}}}{m+1} = 8 \text{ часов}$$

Далее рассчитывается коэффициент готовности для системы из рабочих станций:

$$K_r = \frac{T_O}{T_O + T_{\text{В}}} = 0{,}999655$$

Проводим расчет времени наработки на отказ для одного сервера:

$$T_O serv = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{oi}}} = 43662 \text{ часов}$$

Дальнейшим шагом является проведение расчета времени наработки на отказ для схемы, представленной на рис. 3:

$$T_O 1 = \frac{1}{\sum_{i=1}^{n} \frac{1}{T_{oi}}} = 13130 \text{ часов}$$

Спецификация ЛВС на базе «толстых» клиентов с указанием рассчитанного времени наработки на отказ, а также оценочного времени восстановления представлена в таблице 1

Табл. 1 Показатели надежности ЛВС

Наименование	На базе «толстых» клиентов (кол-во шт., $T_O/T_В$)
Сервер	1, 43662/ 3
Коммутатор	1, 98690/ 1
Рабочая станция	12, 9643/ 24

Коэффициент готовности для ЛВС на базе «толстых» клиентов:

$$К_Г = \prod_{i=1}^{n} K_{Гi} = 0{,}99955$$

Показатель времени восстановления для ЛВС на базе «толстых» клиентов:

$$T_В = T_O 1 \frac{1 - K_Г}{К_Г} = 6{,}03 \text{ часов}$$

Выводы

Для анализа, рассмотрим ЛВС, состоящую из разного количества станций. На рисунке 6 приведен график, показывающий соотношение коэффициента готовности к количеству станций в системе.

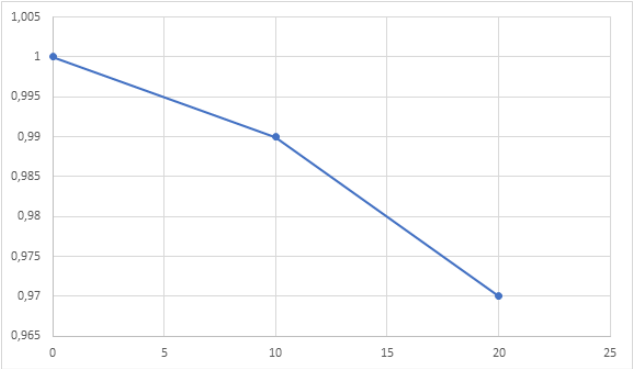


Рис. 6 Отношение коэффициента готовности к числу станций в сети.

Из рисунка видно, что с увеличением числа станций, готовность ЛВС на базе «толстых» клиентов существенно падает.

ЛИТЕРАТУРА

1. Основные понятия и особенности клиент-серверной архитектуры URL: https://testmatick.com/ru/osnovnye-ponyatiya-i-osobennosti-klient-servernoj-arhitektury/
2. Архитектура информационной системы, 2014, URL: http://studopedia.su/10_142448_arhitektura-informatsionnoy-sistemi.html
3. Гавриленко А.А., Полесский С.Н. Анализ построения локально вычислительной сети по критерию надёжности на базе «тонких» и «толстых» клиентов: науч. ст., г. Москва: Московский институт электроники и математики Национального исследовательского университета «Высшая школа экономики», 12 с.

БУБЕР Дмитрий Игоревич – студент 1 курса магистратуры факультета компьютерных наук специальности «Автоматизация та компьютерно-интегрированные технологии», Харьковский национальный университет имени В. Н. Каразина, площадь Свободы, 4, Харьков-22, Украина, 61022; e-mail: buber2020ku51@student.karazin.ua; ORCID: 0000-0001-8199-5916

Научные интересы:

- - *математическое моделирование сложных систем.*

ПАВЛОВ Анатолій Миколайович – старший викладач факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: anatolijpavlov1331@gmail.com,; ORCID: 1331-1331-1331-1331

Наукові інтереси:

- *Комп'ютерні мережі, моделювання комп'ютерних мереж.*

УДК 680.3 **СЕКЦІЯ 2**

БУДЬКО В.В., ТОЛСТОЛУЗЬКА О.Г.

АНАЛІЗ КОМП'ЮТЕРНИХ ЗАСОБІВ ІНКЛЮЗИВНОГО НАВЧАННЯ ДЛЯ ЛЮДЕЙ З ВАДАМИ СЛУХУ

Щодня людина з вадами слуху стикається з безліччю проблем у повсякденному житті. Звичайна розмова з кимось, хто не знає про слуховий бар'єр стає складною. Дехто втрачає слух у випадку складних обставин, а інші мають проблеми з народження.

Навчання дітей з вадами розвитку слуху для їх подальшого майбутнього є довгим та тяжким процесом адаптації до навколишнього середовища. Дехто не чув ніяких звуків зовсім, а хтось чує лише невеликий відсоток. Тому процес навчання вимагає особливих умов та спеціальної підготовки викладачів.

Інклюзивне навчання[1] в Україні є досить складним, оскільки самі умови, що створені для дітей з особливими потребами, є незручними або зовсім непридатними. За Законом України діти з особливими потребами мають право на отримання освіти у спеціальних інклюзивних групах у загальноосвітніх школах, на адаптивні навчальні плани та програми. Існує можливість дистанційного або індивідуального навчання. Однак усі види мають свої недоліки. Якщо долучити до навчання спеціальні технічні засоби та комп'ютерні програми, можна полегшити освітній процес, пришвидшити засвоєння інформації та допомогти дітям з особливими потребами адаптуватися до умов навчання.

З якими проблемами зіштовхуються учні з вадами слуху? У класах, де навчається багато дітей, не всі можуть встигати за викладачем і таким чином втрачати якусь інформацію. Якщо уявити звичайну ситуацію, коли дитина конспектує предмет і одночасно читає інформацію по губах або за допомогою жестів, то досить нахилитися на секунду, щоб втратити частину сенсу. До того ж читання по губах або по жестам займає більше часу, ніж прослуховування звичайного діалогу. Тому занадто швидка подача інформації також несе за собою втрату її цілісності. Ще одним таким фактором є чіткість подання, тобто вимова та рухи губами.

Наступна проблема – це мова. Людина з проблемами слуху не чує звуків та мови, вона може бачити контури мови по губах або за допомогою жестів. Коли викладач говорить українською і переходить на російську або іншу мову змінюється сприйняття, контури та форми губ. Зараз в українських школах процес викладання є виключно українською мовою, але, якщо брати до прикладу Харківську область, багато людей спілкується російською. Мовний бар'єр відчутний, оскільки у родинах або навіть на вулиці дитина бачить контури мови, які кардинально відрізняються від тих, до яких вона звикла на уроках у школі.

Більшість людей вважають, що слухові апарати вирішують всі проблеми, але це далеко не так. В основному вони обмежені за діапазоном частот, тобто розпізнають лише невелику кількість звуків, і досить часто, це лише механічні звуки, а не голос. Також апарати обмежені відстанню сприйняття звуків. І, найголовніше, вони підходять лише людям, які мають якийсь відсоток слуху.

У наших силах поліпшити процес навчання. Важливо забезпечити учнів правильним та повним донесенням інформації під час уроку. За допомогою технологій розпізнавання мови можна створити програмне забезпечення спеціально налаштоване для процесу інклюзивного навчання.

Які технічні засоби існують зараз? Наприклад, Яндекс.Розмова[2]. Додаток має вигляд звичайної переписки, де співрозмовник може ввести своє повідомлення. Також у програмі є вже готові фрази, на які досить натиснути, щоб відтворити їх. В умовах написано, що існує функція розпізнавання мови, але для цього навколо має бути максимально тихо і співрозмовник має говорити голосно і чітко, а також знаходитися близько. За схожим принципом працює Deaf Pad Pro[3]. Додаток виступає у ролі розумного блокноту. Але, нажаль, ці засоби не зовсім підходять для інклюзивного навчання, оскільки вони складні для сприйняття і не завжди працюють коректно. В більшій мірі вони направлені на подачу інформації у навколишній світ, а не на

© БУДЬКО В.В., ТОЛСТОЛУЗЬКА О.Г., 2021

отримання її для людини з вадою слуху, що є найважливішим для навчального процесу. Зручнішим для цих цілей був би голосовий ввід, але, нажаль спеціалізованих програм немає.

Що має бути у програмному рішенні для представленої проблеми? Чітке розпізнавання мови, можливість конспектування та збереження текстових файлів, можливість перекладу на різні мови, зручний та зрозумілий інтерфейс, який не вимагатиме багато часу для ввімкнення та вимкнення запису та конспектування. Було б великою перевагою, якби програмний додаток мав певний набір голосових команд для викладачів. Наприклад, щоб привернути увагу до важливих рядків або запустити автоматично режим конспектування.

Таке програмне рішення можна реалізувати за допомогою Web Speech API[4] у вигляді веб-додатку. Створення рішення у форматі веб-додатку допоможе зробити його доступним на різних пристроях та не прив'язуватися до одного девайсу. Хмарне збереження даних допоможе дати доступ до конспектів як і в класі, так і з дому. Також додаток відкритий на смартфоні допоможе вирішити проблему і за межами школи або дому.

ПЕРЕЛІК ДЖЕРЕЛ

1. http://ispukr.org.ua/?page_id=331#.YGYMOq8zZEY
2. https://te-st.ru/entries/yandex-talk-app/
3. https://www.mobileaction.co/app/android/us/deaf-pad-pro/canalrun.apps.deafpad
4. https://developer.mozilla.org/ru/docs/Web/API/Web_Speech_API

БУДЬКО Владислава Віталіївна – студентка кафедри штучного інтелекту та програмного забезпечення; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: xa11867778@student.karazin.ua; ORCID: 0000-0003-1344-3153.

Наукові інтереси:

– *моделювання складних комп'ютерних систем*

ТОЛСТОЛУЗЬКА Олена Генадіївна – д. т. н., с. н. с.; професор кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: elena.tolstoluzka@karazin.ua; ORCID: 0000-0001-2741-180.

Наукові інтереси:

– *технології розподіленого і паралельного програмування*

УДК 004.78:519.72 **СЕКЦІЯ 2**

БУЄВИЧ-СИСОЄВ В.М., ШМАТКОВ С.І.

ІНФОРМАТИВНІСТЬ ПАРАМЕТРІВ КОМП'ЮТЕРНИХ СИСТЕМ

Актуальність

Моніторинг мережі відіграє важливу роль в управлінні мережею. Завдяки аналізу мережевих параметрів (наприклад, пропускної здатності) оператори можуть спостерігати за поведінкою мережі та приймати рішення на їх основі. Вибір параметрів мережі повинен бути актуальним для кожної конкретної мети. У цій роботі увага зосереджується на аналізі інформативності мережевих параметрів, які мають значення для управління мережею. У цій статті використовувався метод статистичного аналізу потенційних параметрів мережі, як оцінювачів потоку. Основним внеском цієї роботи є запровадження статистичної методології для прогнозування підтвердження того, що деякі параметри мережі можна розглядати, як інформативні. Тому ці прогнози є актуальними для використання в класифікації стану комп'ютерних систем.

Вступ

На сьогоднішній день рішення вибору параметрів для моніторингу мережі все ще залежить від операторів мережі. Однак це рішення не є тривіальним. Операторам доводиться справлятися з кількома параметрами мережі (наприклад, пропускна здатність потоку, тривалість потоку і т.д.), щоб проводити збалансований моніторинг. Тому вибір належного набору параметрів мережі є вирішальним для цього завдання. Крім того, оператори мережі повинні своєчасно вибирати потоки, оскільки набагато дешевше переправити трафік на оптичному рівні, ніж на рівні IP, на що вказує дослідження Cees de Laat [1].

Тому пропонується використовувати статистичні методи, такі як кореляція та класифікація дерев, як методологія для оцінки, які параметри мережі є інформативними для моніторингу стану комп'ютерних систем. Отже, така методологія є головним внеском цієї роботи.

Методологія

Параметри мережі надають цінну інформацію про стан мережевого трафіку та пристроїв (наприклад, маршрутизаторів, комутаторів). Але не всі параметри змінюються при різній інтенсивності трафіку, тому у контексті нашого підходу важливо прогнозувати параметри мережі тільки інформативні. Цей підхід використовується для вирішення задачі класифікації стану комп'ютерної системи.

Щоб визначити набір параметрів, було проведено дослідження літератури для переліку параметрів кандидатів. Такі параметри кандидатів були взяті з наступних джерел: (a) модулі MIB MIB-II [2], RMON-2 [3] та SMON [4]; (b) інформаційна модель для протоколу IP Flow Information eXport (IPFIX) [5]. Оскільки ці джерела мають справу з інформативністю. Результатом цього дослідження є визначення параметрів мережі, які відповідають інформативним параметрам мережі[6]. В ході роботи були відкинуті параметри, які не змінювались при різній інтенсивності навантаження та вибрані основні параметри, які відповідають інформативним параметрам мережі: джиттер, швидкість передачі інформації, затримка передачі пакета по мережі і втрати пакетів (рис. 1). Даний набір параметрів часто використовують при оцінці якості обслуговування мережі (QoS), що дозволяє описати стан комп'ютерної системи.

© БУЄВИЧ-СИСОЄВ В.М., ШМАТКОВ С.І., 2021

Total Packets	Minimum delay(s)	Maximum delay(s)	Average delay(s)	Average jitter(s)	Delay standard deviation(s)	Bytes received	Average bitrate(Kbit/s)	Average packet rate(pkt/s)	Packets dropped	Average loss-burst size(pkt)	Error lines
5680	0,137528	0,151015	0,142838	0,004436	0,002834	2908160	1163,23	283,99	0	0	0
5679	0,137496	0,147758	0,142808	0,004437	0,002831	2907648	1162,92	283,92	0	0	0
5678	0,137435	0,148422	0,142838	0,004431	0,002829	2907136	1163,26	284,00	0	0	0
5679	0,137461	0,148064	0,142597	0,004432	0,00283	2907648	1162,97	283,93	0	0	0
5681	0,137468	0,149377	0,142817	0,004434	0,002836	2908672	1163,81	284,13	0	0	0
5681	0,137474	0,147807	0,142594	0,004428	0,002825	2908672	1163,39	284,03	0	0	0
5679	0,137494	0,147638	0,142579	0,004427	0,002827	2907648	1163,34	284,02	0	0	0
5683	0,137415	0,150235	0,142552	0,004425	0,002829	2909696	1163,87	284,15	0	0	0
5682	0,137397	0,148095	0,142583	0,004421	0,002825	2909184	1163,74	284,12	0	0	0
5674	0,13745	0,149121	0,142614	0,004394	0,002808	2905088	1162,30	283,77	0	0	0
5677	0,137424	0,150926	0,142639	0,004418	0,002821	2906624	1162,95	283,92	0	0	0
5679	0,137361	0,147713	0,142531	0,004421	0,002823	2907648	1163,27	284,00	0	0	0
5680	0,137311	0,147633	0,142513	0,004426	0,002827	2908160	1163,66	284,10	0	0	0
5681	0,137355	0,148515	0,142511	0,004423	0,002827	2908672	1163,94	283,01	0	0	0
5681	0,13738	0,147608	0,14246	0,004431	0,002827	2908672	1163,30	284,01	0	0	0
5678	0,13735	0,14966	0,142494	0,004421	0,002825	2907136	1163,13	283,97	0	0	0
5678	0,137311	0,15985	0,142451	0,00444	0,002847	2908112	1163,07	283,95	0	0	0
5677	0,137322	0,1477	0,142471	0,004428	0,002826	2906624	1163,07	283,95	0	0	0
5680	0,137315	0,147924	0,142431	0,004429	0,002827	2908160	1163,52	284,06	0	0	0
5678	0,137276	0,147645	0,142411	0,004421	0,002824	2907136	1162,79	283,88	0	0	0

Рис. 1 – параметри мережі

Збір мережевих метрик

Процес полягав у зборі даних з маршрутизатора з підтримкою NetFlow, що працює з версією 9. Цей маршрутизатор експортував записи NetFlow у колектор потоків. Метрики від мережі збиралися протягом 500 натурних експериментів з різною завантаженістю мережі в залежності до класу стану комп'ютерної системи (рис. 2).

При малому навантаженні мережі було вирішено використовувати швидкість передачі даних в 300 пакетів/сек, для середньої - 3000 пакетів/сек, а для високої - 30000 пакетів/сек. Об'єм кожного пакету мережі становив 512 байт та використовувався 1 потік передачі даних.

	Total Packets	Minimum del	Maximum del	Average dela	Average jitter	Delay standa	Bytes receive	Average bitra	Average pacl	Packets drop	Average loss	Error lines
2	30902	0,399867	0,480456	0,463712	0,001132	0,004921	15821824	6309,45	1540,39	246472(88.86	127,90	0
3	30900	0,40333	0,477192	0,463727	0,001138	0,004951	15820800	6309,93	1540,51	255694(89.22	132,76	0
4	30916	0,4033	0,470282	0,463655	0,001138	0,004938	15828992	6312,26	1541,08	247672(88.90	128,53	0
5	30528	0,404322	0,471396	0,4637	0,001134	0,00487	15836672	6315,27	1541,81	253740(89.13	131,54	0
6	30932	0,403109	0,470122	0,463667	0,001134	0,004916	15837184	6316,31	1542,07	259125(89.34	134,40	0
7	30935	0,403231	0,474754	0,463642	0,001134	0,004918	15838720	6317,78	1542,43	253123(89.11	131,22	0
8	30916	0,402967	0,470269	0,463735	0,001133	0,004956	15828992	6312,13	1541,05	262605(89.47	136,35	0
9	30932	0,403386	0,472166	0,463647	0,001139	0,00494	15837184	6316,24	1542,05	259038(89.33	134,43	0
10	30908	0,402965	0,47323	0,463716	0,001137	0,004933	15824896	6311,98	1541,01	257786(89.29	133,71	0
11	30915	0,403768	0,470461	0,463679	0,001139	0,004921	15828480	6313,48	1541,38	254413(89.17	131,96	0
12	30900	0,40333	0,477192	0,463727	0,001138	0,004951	15820800	6309,93	1540,51	255694(89.22	132,76	0
13	30916	0,4033	0,470282	0,463655	0,001138	0,004938	15828992	6312,26	1541,08	247672(88.90	128,53	0
14	30931	0,404322	0,471396	0,4637	0,001134	0,00487	15836672	6315,27	1541,81	253740(89.13	131,54	0
15	30932	0,403109	0,470122	0,463667	0,001134	0,004916	15837184	6316,31	1542,07	259125(89.34	134,40	0
16	30935	0,403231	0,474754	0,463642	0,001134	0,004918	15838720	6317,78	1542,43	253123(89.11	131,22	0
17	30916	0,402967	0,470269	0,463735	0,001133	0,004956	15828992	6312,13	1541,05	262605(89.47	136,35	0
18	30932	0,399753	0,470575	0,463682	0,001138	0,004857	15837184	6313,36	1541,35	261476(89.42	135,48	0
19	30932	0,403771	0,471795	0,463656	0,001139	0,004927	15837184	6314,59	1541,65	256507(89.24	133,04	0
20	30931	0,403417	0,470117	0,463669	0,00114	0,004929	15836672	6314,00	1541,50	260556(89.39	135,14	0
21	30918	0,403166	0,474473	0,463666	0,001136	0,00493	15830016	6314,26	1541,57	255057(89.19	132,22	0
22	30915	0,403859	0,472821	0,463676	0,001139	0,004914	15828480	6312,99	1541,26	261761(89.44	135,84	0
23	30915	0,403175	0,473466	0,463677	0,001138	0,004936	15828480	6312,16	1541,05	260157(89.38	135,01	0
24	30889	0,403247	0,511809	0,46378	0,001137	0,005121	15815168	6307,79	1539,99	253318(89.13	131,59	0
25	30931	0,403433	0,472042	0,463669	0,001135	0,004897	15836672	6315,08	1541,77	260121(89.37	134,92	0
26	30916	0,40385	0,471775	0,463694	0,001139	0,004916	15828992	6312,92	1541,24	261537(89.43	135,72	0

Класс1 | Класс2 | Класс3 | Класс4 | Класс5

Рис. 2 – класи стану комп'ютерних систем

Висновок

У цій роботі представлена статистична оцінка потенційних параметрів мережі для класифікації стану комп'ютерних систем. Було піднято два наукові запитання:

Дослідницьке питання 1: Які параметри мережі можна використовувати для прогнозування стану комп'ютерних систем? Кількість потенційних параметрів мережі дуже широкий. Тому вибір параметрів здійснювався залежно від визначеної мети. Оскільки мета зосереджувалась на передбаченні стану комп'ютерних систем, було звужено вибір параметрів мережі, дослідивши які з параметрів відповідають інформативності. Проведений аналіз літератури дозволив звузити параметри кандидатів з наступних джерел: (a) модулі MIB MIB-II [2], RMON-2 [3] та SMON [4]; (b) інформаційна модель для протоколу IP Flow Information eXport (IPFIX) [5]. З цих параметрів в ході роботи були відкинуті параметри, які не змінювались при різній інтенсивності навантаження та вибрані основні параметри, які відповідають інформативності: джиттер, швидкість передачі інформації, затримка передачі пакета по мережі і втрати пакетів.

Питання дослідження 2: Як можна оцінити інформативність цих параметрів? Ці показники оцінювались за допомогою методів статистики. Процес полягав у зборі даних з маршрутизатора з підтримкою NetFlow. Метрики від мережі збиралися протягом 500 натурних експериментів з різною завантаженістю мережі в залежності до класу стану комп'ютерної системи (Клас 1, Клас 2, Клас 3, Клас 4, Клас 5). Таким чином, оскільки всі параметри мають

певний вплив на об'єм потоку, то їх слід використовувати не окремо, а в зв'язці. Тоді було використано техніку CHAID (дерево рішень) та [6, 101-104 с.] для аналізу інформативності параметрів. За аналізом CHAID, джиттер, швидкість передачі інформації, затримка передачі пакета по мережі і втрати пакетів є інформативними параметрами.

Як майбутнє дослідження, ми націлені на подальше використання цих параметрів в аналізі трафіку та його класифікації в комп'ютерних системах в розробленій моделі станів.

ЛІТЕРАТУРА

1. de Laat, C., Radius, E., Wallace, S.: The Rationale of the Current Optical Networking Initia- tives. Future Gener. Comput. Syst. 19(6), 999–1008 (2003)
2. Raghunarayan, R.: Management Information Base for the Transmission Control Protocol (TCP). RFC 4022 (Proposed Standard) (March 2005)
3. Waldbusser, S.: Remote Network Monitoring Management Information Base Version 2. RFC 4502 (Draft Standard) (May 2006)
4. Waterman, R., Lahaye, B., Romascanu, D., Waldbusser, S.: Remote Network Monitoring MIB Extensions for Switched Networks Version 1.0. RFC 2613 (Draft Standard) (June 1999)
5. Quittek, J., Bryant, S., Claise, B., Aitken, P., Meyer, J.: Information Model for IP Flow In- formation Export. RFC 5102 (Proposed Standard) (January 2008)
6. Стрілець В. Є., Шматков С. І., Угрюмов М. Л., Мєняйлов Є. С., Черниш С. В., Угрюмова К. М. Методи машинного навчання у задачах системного аналізу і прийняття рішень. *Видавець і виготовлювач - Харківський національний університет імені В. Н. Каразіна, 61022, Харків, майдан Свободи, 4, Свідоцтво про державну реєстрацію ДК No 3367 від 13.01.2009,* 101-104 с.

БУЄВИЧ-СИСОЄВ Владислав Миколайович – ст. викладач; кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, 61022; e-mail: buievych.sysoiev@karazin.ua; ORCID*: https://orcid.org/0000-0002-4195-0526.

Наукові інтереси:

– Програмування, структури даних.

ШМАТКОВ Сергій Ігорович – зав. каф. кафедри теоретичної та прикладної системотехніки, д. т. н., професор; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, Харків-22, Україна, 61022; e-mail: s.shmatkov@karazin.ua; ORCID: 0000-0002-6328-988X.

Наукові інтереси:

– методи управління, організація інформаційних процесів і моделювання складних адаптивних розподілених систем, системний аналіз.

UDC 004.056.55 **СЕКЦІЯ 5**

BUZKO K.S., NARIEZHNII O.P., GRINENKO T.O.

ANALYSIS OF CYBER THREATS IN THE LOCAL SYSTEM OF DIFFERENTIAL CORRECTION OF GPS SIGNALS

Introduction. Over the last decade, considerable attention has been paid abroad to the creation of local (tactical) differential correction systems (DCD) using pseudo-satellites (PS) [1, 2]. Therefore, a promising coordinate-temporal navigation and information system of operational and tactical purposes should be based on the use of broadband signals with pseudo-random modulation, which allows to realize the latency of radiation, crypto-security, noise immunity to significant and artificial interference and multicast data transmission. In addition, a promising tactical DCD must:

1. Apply the L-frequency range to minimize weight and size characteristics and simultaneously emit C/A (clear acquisition code) code and broadband WB (P) (wideband (protected) code) - code with a pseudo-random sequence (PRN 33-37).

2. Create influence on signals of global navigation satellite systems (GNSS) of irresistible artificial interference or physical destruction of its elements in a given local zone.

3. Provide, if necessary, interference with the systems of a probable enemy.

Main part. The differential mode of DCD measurements is based on the spatio-temporal correlation of the main measurement errors in the navigation equipment of consumers and the control and correction station (CCS). In tactical DCD, the accuracy of the method of using relative coordinates in the first approximation is equivalent to the accuracy of the standard differential GNSS mode. However, this type of differential mode is created when the signals transmitted by the differential station (CCS) are tied to a local time scale. Because as a result of taking into account the differential corrections, the time scale of the consumer also becomes tied to the time scale of the differential station, the signals received by the consumer are a source of information about the mutual distance and speed between the consumer and CCS. In the case of a tactical DCD, the differential stations (CCS) shall emit a signal similar to the GNSS signal. The issue of optimal location of aircraft for differential GNSS subsystems is devoted to [1]. The use of the aircraft makes it possible to correct the bad geometrical factors of GLONASS and GPS, if they appear.

Deliberate distortion of CCS state vector data is the main task of a cyber-attack on tactical DCD. This type of hacker attack refers to modification attacks [2, 3]. RTCM SC-104 standards define messages that contain reference station (or CCS) information and system data. Using the concept of a Virtual Reference Station (VRS) and making errors in the calculation of the pseudo-range correction for each CCS, it is possible to significantly distort the consumer's information about its actual location. The NTRIP (Networked Transport of RTCM via Internet Protocol) can be used to transmit RTCM data to the VRS. The RTCM correction information is obtained for the virtual point corresponding to the exemplary location of the user, based on data from a number of reference stations (or CCS).

Thus, the easiest way to attack a tactical CCS is to create a VRS. To do this, carry out the following attack scheme:

1. Identify the coordinate, frequency-time parameters of the CCS and modify them when transmitting over the Internet using the NTRIP protocol.

2. Make an obstacle setting and block GNSS signals in the tactical CCS receiver.

3. Emulate "raw" data from the blocked CCS to hide the modification of data for users of the local CCS when processing them in the data collection center of the regional DCD.

Modification of the data of the vector of states of CCS at such type of attack will allow seizing completely control of issue of corrective information in tactical DCD. To eliminate this threat, it is necessary to apply the procedure of authentication of corrective information in the CCC, which is a procedure to verify that the data obtained came from the specified CCC and were not modified. At the same time, the authentication of corrective information should also involve checking the order of passage and timeliness of delivery of messages.

© BUZKO K.S., NARIEZHNII O.P., GRINENKO T.O., 2021

The technique of data authentication in CCS involves attaching to the corrective information created using a secret key a small block of data of a fixed size, called a cryptographic checksum or message authentication code (Message Authentication Code, MAC). It is assumed that the two parties involved in the exchange of data, the CCC and the consumer, use a shared secret key.

The following results were obtained during the research:

1. Cyber threats are a major challenge for the future use of GNSS [2]. Cyber-attacks can affect both onboard and ground systems, and methods of influence can range from simple asymmetric attacks, such as jamming, to highly sophisticated electronic systems and software attacks.

2. In order to eliminate the modification attack on the CCS server, the information (internal information) about the height of the phase center of the antenna of the consumer navigation equipment and the parameters of the reference generator must be encrypted.

3. As an example, with the help of a simulation complex was simulated the work of CCS in real time. It is shown that the use of MAC code makes an attack impossible based on the application of the VRS concept.

4. The simplest cyber-attack on a tactical DCD is based on the impossibility of implementing a reliable determination of the validity of corrective information (CI) in terms of "one-way" communication channel. Because in this situation the well-known hacker attack by repeating the signal (replay attack) works well. Therefore, to control the probability of CI necessarily requires a duplicate radio data channel, which is quite difficult to suppress by means of electronic warfare.

Conclusions. The paper considers the issues of building a local autonomous dual-purpose SDU. An important aspect of the security of dual-purpose DCD is the means of cryptographic protection of CI. It is shown that intentional distortion of GPS signal state vector data and CCS differential corrections is the main task of a cyber-attack on A-GPS or GPS algorithms of mobile devices. To eliminate the threat of violation of the integrity of differential corrections, the procedure of CI authentication broadcast by CCS is used. At the same time, the data transmitted to the CCS is checked for modification, the order of passage and timeliness of delivery of CI messages.

REFERENCES

1. Parkinson B.W., Fitzgibbon K. T. Optimal Locations of Pseudolites for Differential GPS. Navigation. 1986. Vol. 33, No 4. P. 259-283.
2. GPS Software Attacks / T. Nighswander, B. Ledvina, J. Diamond, R. Brumley, D. Brumley // Proceedings of CCS 2012. - pp. 450-461. DOI: 10.1145/2382196.2382245.
3. Grinenko T.O., Narezhniy O.P. Message authentication codes application for data modification detection in regional systems of differential correction of GPS/GLONASS systems navigation signals. Applied Radio Electronics: Sci. Journ. 2014. Vol. 13. № 3. P. 301–310 (in Ukrainian).

Kseniia S. BUZKO – student specializing in computer science of the V. N. Karazin Kharkiv National University.

Areas of scientific interests: *mathematical models of cyber attacks based on A-GPS algorithms, GPS mobile attachments.*

Oleksii P. NARIEZHNII – Candidate of Sciences (Engineering), Associate Professor of the Department security information systems and technologies of the V. N. Karazin Kharkiv National University.

Areas of scientific interests: *cybersecurity, security information systems and technologies.*

Tetiana O. GRINENKO – Candidate of Sciences (Engineering), Associate Professor of the Department of information technology security of the Kharkiv National University of Radio Electronics.

Areas of scientific interests: *cybersecurity, security information systems and technologies.*

УДК 658.5 **СЕКЦІЯ 2**

BULAVIN D. O.

QUALITY CONTROL AND PROJECT MANAGER ROLE IN OUTSOURCING/ OUTSTAFFING COMPANIES

What is your system approach to project quality management?

In today's world, to achieve success in competition, it is necessary to ensure the optimal combination of well-established business processes in management structures and dynamic and result-oriented design approaches. It is the project quality management role that currently plays a special relevance in project management because only a high-quality product can win out the competition in today's realities.

Each project manager ultimately has the task of optimizing the management process to achieve the project goal more effectively. Project quality management includes the processes required by the quality assurance organization regarding IT project planning, management, and control. Project quality management includes the following processes:

- Quality management planning;
- Quality management;
- Quality control.

At the start of the planning process it is necessary to obtain all necessary information for starting the project: contents of the project, the technical specification of the project which needs to be approved together with the client, standards and requirements to software quality and realization of all processes, and documentation on the quality system.

Now in modern IT companies, quality assessment is considered from the standpoint of an individual customer. It is with him or her at the stage of forming requirements that the quality criteria of the developed software product are determined. This is because every product and every project is unique. That is why quality indicators may not differ significantly from one project to another.

When we analyze the indicators that affect the quality of the project, it becomes necessary to select the most influential ones. Some can qualitatively change the final result of the project in the company. The solution to the task and the achievement of the necessary results comes from the selection of the most important indicators for the analysis of impact and determination of the quality component as a result of the analysis of the quality level of the project under study. This approach allows us to use information visualization tools – Ishikawa diagrams.

The ranking procedure consists of an expert having to arrange the indicators of the IT-project quality in a rational order and assign each of them the numbers in a natural series – ranks. As rank is understood as an indicator (number) characterizing the ordinal position of the evaluated object of research in a group of objects that have the properties necessary for evaluation.

The rating system allows us to manage through coordinated planning, setting control over the quality level of IT projects. By tracking deviations from the set goals and strategic objectives, we can increase the efficiency of the process of managing the quality level of IT projects and, thus, improve the results of enterprises in general.

Based on my experience working with custom engineers from different countries on projects of different sizes, the use of mathematical methods for quality management is advisable and successfully financed only on large projects of 2 years or more and with a team of 15-20 people. On smaller projects, the use of such techniques is labor-intensive and seldom customers agree to finance this type of work. My suggestion for projects like this is to implement clear milestones and quick discussions of the results after each one.

What are the main features of customers from Holland, USA, Israel, or England?

The work of an IT company in Ukraine in most cases is aimed at foreign customers in the form of companies of different sizes. At the same time, the success of cooperation largely depends on whether the management of an IT company understands the key differences that exist between customers from different countries of the world.

Consider the key areas with which our company is currently cooperating. It is necessary to identify key differences, as well as highlight common similarities that can be used to achieve success.

© BULAVIN D. O., 2021

The Israeli market is now growing very fast, and the number of projects that come to us from there is really large.However, many of them are startups. This means that the client wants to make an MVP of the product, and in the future receive its full funding. In this regard, the customer does not want to spend his financial resources at this stage on connecting a business analyst to create a full-fledged technical task, and the project description often consists of a small document describing key details. Besides, some of them, in an attempt to save money, create a back-end on their side and give the front-end to outsourcing. In this case, quite often a lot of time is spent on implementation, and this leads to client dissatisfaction, as well as possible financial losses. That is why the assessment of time costs on the side of the company must be approached with special care, and a T&M contract should be made. Another rather interesting feature of Israeli clients is jokes/anecdotes – if the PM is on good terms with the customer, then they love informal communication and can continue to communicate with the messenger on completely different topics.

Dutch clients are always very polite and pleasant to talk to. They often order large systems, the development of which is completely outsourced. The technical documentation of such projects for many of them is compiled at a high level, and the design is either outsourced and approved or provided by the client in the form of sketches / full design. Provided that good contact with the client is established, and a clear record of all tasks and time is maintained, work with Dutch clients is successful and well-tuned, without elements of chaos.

American clients see each of their projects as special, and certainly one that will turn the world upside down. An unconditional feature is that they require the constant attention of developers, PMs – even if the team is not sold full-time. It is impossible not to mention the dependence of technical documentation on an American client. There can be both an ideal technical specification and a short description up to the moment of MVP, and then we will come up with it as the project progresses. The ideal work with American clients involves a T&M or outstaff team and displays the interest of the entire team in the client's ideas.

Great Britain has very pedantic clients and the terms of reference for their projects can be very huge, while there will also be a lot of water in it. Working with such clients is very dependent on how much you will be on the same wavelength with them and comply with all their terms and conditions. They really do not like to use American services, so offering them Amazon can cause you to completely lose the project. Drawing up clear plans, working towards a result, and conducting high-quality demos is the key to successful cooperation with British customers.

What is the PM's role in Outsourcing / Outstaffing?

In the modern IT industry there are many types of cooperation, and one of the most commonly used in our country are Outsourcing and Outstaffing models. The key difference between these models is that in Outsourcing, a customer company concludes a contract with another executing company to perform some work. And in Outstaffing, the customer company buys out the entire team – or a certain number of employees of the executing company to work full time.

When analyzing the tasks of the PM, which he or she performs in these models, one can note both similarities and differences. Of course, in both models, the PM is responsible for project analysis, phase identification, assessment, and distribution within the team. Also, the PM is obliged to monitor the results of the team both in small phases and throughout the entire project distance, assess risks, manage the quality of work, and use soft skills to maintain the microclimate within the team and when communicating with the customer.

However, in most cases, in the Outsourcing model, the PM manages a certain set of projects (from 3 and higher), and in the Outstaffing model, the use of a full-time PM within the framework of work on 1 project prevails. Most often, in this case, it is a long-term project with a large team (10 or more team members). In such cases, the PM is obliged to work with large volumes of incoming data, take into account the peculiarities of the system being developed and the impact of changes introduced, carry out mandatory version control, and, in some cases, perform the duties of a business analyst.

In the Outsourcing model, the PM participates in presale activities, carries out classical PM activities on several projects, prepares upselling plans for current projects, and manages teams in terms of software and hard skills growth.

To sum up, we can say that both models are effective and contribute to the growth of PM qualifications from different sides, and to single out only one as the most correct would not be a correct decision.

How do you properly build effective communication with the customer?

Every time a new project starts and a presale meeting occurs, I think about what the new customer will be like, whether our communication will be productive and mutually interesting, and how to correctly build a negotiation strategy at the initial and subsequent stages of working with him or her. I think everyone will agree that effective communication is not mathematics or physics, there is no single formula or strategy for the PM to adhere to a certain rule and be successful with all clients. Our second section, where we discussed the specifics of customers from different countries, only confirms this. The eternal question arises: "How do you properly build effective communication with the customer?

– In my opinion, it is very important at the first meeting with a customer to show oneself as honest, open, and most importantly interested in the project and its details. A good PM should always remember that for a customer a project is a brainchild that he or she wants to bring to the world. That is why, as one well-known trainer used to say: "The eyes must burn."

The second important aspect of the conversation is to clarify the interests of the client in addition to the project.his helps to reduce the degree of tension at a time when there is a delay on the project. However, you should not use this if there is a crisis on the project because while you can certainly communicate with the client on topics that are interesting to both, during a crisis it is essential to be professional and serious. In such moments, the customer wants to see the maximum involvement of both PM and the whole team in solving difficulties. The client should feel confident that the PM will find words of motivation for the team and all controversial points will be eliminated.

Summarizing all of the above, let us note the key qualities that the Project Manager should possess in the negotiation process:

- Honesty and openness;
- Analytical skills;
- Delivered competent speech;
- Ability to instill confidence in the client;
- Punctuality;
- A broad outlook for dialogue.

One of the best cases of the negotiation process in my experience was on a project that had already lasted about 2 years and was close to failure. The key mistake of previous PMs was the strict adherence to customer documents and the lack of attempts to analyze the application to the market. In the process of communicating with the customer, which was on average 3-4 hours a day, we discussed that the key feature of applications should be an orientation to the web with mandatory responsiveness. As a result, the size of the team grew to 8 people, the development of the project started from scratch, and the company's income grew by 300%.

Three tips for beginner Project Managers

- At the start of communicating with the customer – try to win him or her over to you and show that you are a professional;
- In the process of working with a client – be honest, but highlight the achievements of your team, as he or she must know that he is working with the best;
- In each phase of the project – consider any upselling possibilities, but first, clearly articulate the details and all benefits for the customer.

REFERENCES

1. Schwaber Ken Agile Project Management with Scrum, Microsoft Press 2004 – 163 p.
2. International Standards Organization. 2015. ISO 9000:2015 Quality Management Systems—Fundamentals and Vocabulary. Geneva: Author
3. Project Management Institute. 2014. Implementing Organizational Project Management: A Practice Guide. Newtown Square, PA: Author
4. Project Management Institute. 2017. The Standard for Program Management – Fourth Edition. Newtown Square, PA: Author.

БУЛАВІН Дмитро Олексійович – к. т. н., доцент; доцент кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: d.bulavin@karazin.ua; ORCID: 0000-0002-4840-4763.

Наукові інтереси: – *управління IT-проектами, методи оптимізації.*

УДК 519.2 **СЕКЦІЯ 2**

БУРЛАЙ І.С., БАКУМЕНКО Н.С.

КОМП'ЮТЕРНА МОДЕЛЬ СИСТЕМИ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ЦІН НА ЕНЕРГОНОСІЇ ЗА ДОПОМОГОЮ МЕТОДУ ARIMA (p,q,k)

Ще до початку постіндустріальної або інформаційної епохи, значно виросли потреби у обробці інформації, обсяги даних стали настільки великими, що це спричинило розвиток нової галузі – data mining. Нині аналіз даних є однією з провідних галузей. Одним з основних напрямків цієї дисципліни є прогнозування, яке засноване на методах обробки часових рядів. В роботі розглядаються нестаціонарні часові ряди та методи їх аналізу.

Спершу розглянемо модель авторегресії-проінтегрованного ковзного середнього (ARIMA(p, q, k)-модель). Ковзне середнє – це метод, заснонований на обчисленні середнього підмножини значень.

ARIMA(p, q, k)-модель була розроблена Джорджем Боксом і Гвілімом Дженкінсоном. Вона використовується для опису нестаціонарних часових рядів x_t, якщо ці ряди мають деякі властивості:

1) у складі ряду адитивно присутня складова f(t), у вигляді алгебраїчного поліному деякої степені k ($k > 1$). Поліноміальні коефіцієнти можуть мати і стохастичне, і не стохастичне походження [1];
2) ряд x_t , над яким застосували k-кратну процедуру послідовних різниць і отримали ряд x_t^k, де $t = 1, \dots, T - k$. може бути описаний моделлю ARMA(p, q)

ARIMA(p, k, q) - модель процесу x_t , який ми аналізуємо, можна представити у вигляді:

$$x_t^k = \alpha_1 x_{t-1}^k + \alpha_2 x_{t-2}^k + \cdots + \alpha_p x_{t-p}^k + \delta - \theta_1 \delta_{t-1} - \cdots - \theta_q \delta_{t-q}\,, \quad (1.1)$$

де $x_t^k = \Delta^k x_t = x_t - C_k^1 x_{t-1} + C_k^2 x_{t-2} - \cdots + (-1)^k x_{t-k}\,,$ (1.2)

де C_k - число поєднань, Δ - різницевий оператор.

Для ідентифікації ARIMA-моделей спочатку визначається порядок моделі k, тут існує 2 типи критеріїв підбору:

1) опирається на відслідковування поведінки величини $\hat{\sigma}^2(k)$ – оцінки дисперсії k-і різниці, в залежності від k;
2) опирається на аналіз поведінки автокореляційних функцій процесів $\Delta x_t, \Delta^2 x_t, \dots$, послідовні перетворення нашого процесу x_t за допомогою операторів $\Delta_t, \Delta^2, \dots$ націлені на ліквідацію його нестаціонарності

Після підбору, починають аналізувати k-ті різниці ряду x_t . Ідентифікація такого ряду зводиться до ідентифікації ARMA(p, q) моделей [2]. А саме, визначення загального вигляду поточної моделі, що проявляється у вигляді найменшого числа параметрів, відносно інших можливих варіантів, при цьому, на точність опису вихідного процесу, це ніяк не впливає.

Тепер розглянемо моделі часових рядів, що містять сезонну компоненту. Це процеси, при формуванні значень завжди присутні сезонні чи циклічні фактори.

Один із найбільш використовуваних підходів прогнозування полягає в наступних діях. Ряд розділяють на частини: довготривала, сезонна(циклічна) і випадкова складові. Далі довготривалу підганяють поліномом, а сезонну – рядом Фур'є. Наступним кроком, відбувається прогноз, який відбувається методом екстраполяції оцих даних. Нажаль, цей підхід не застрахований від помилок. А саме: відбуваються спотворення на коротких відрізках, через імітацію частин поліноміальних або гармонійних функцій; або якщо ряд містить невипадкові поліноміальні і гармонійні складові, їх формальна апроксимація може бути жадібною, тобто потребувати велику кількість параметрів, що не є економічним.

Також, є ще один поширений підхід, в основі якого модифікація ARIMA-моделей за допомогою «спрощуванних операторів». Наприклад, візьмемо ARIMA-конструкції модифікованих за допомогою спрощуванних операторів $\nabla_T = 1 - F^T$ і процедуру побудови сезонних моделей на їх основі, можна описати:

© БУРЛАЙ І.С., БАКУМЕНКО Н.С., 2021

1) застосовуємо до спостерігаємого ряду x_t , оператори Δ , ∇_T;
2) опираючись на вид авто кореляційної функції перетвореного ряду $X_{k,K}^{(T)}(t)$ підбираємо пробну модель ARMA(чи модифікованих в правій частині ARMA-моделей);
3) по отриманим значення автокореляції, отримуємо методом моментів оцінки параметрів пробної моделі;
4) діагностична перевірка моделі-результату або підтверджує коректність роботи, або вказує на недоліки, після чого необхідно повторити цикл процедури.

ARIMA-моделі працюють з великою частиною часових рядів, у тому числі і з сезонними часовими рядами, для яких використовують невеликі модифікації ARIMA [3]. Розглянемо методи, які засновані на використанні ARIMA-моделей. Зазвичай, сюди входять і окремі випадки AR-, MA-, ARMA-моделі.

Нехай ми вже здійснили вибір потрібної моделі для нашого ряду, включаючи ідентифікацію моделі. І тому, надалі будемо вважати, що усі параметри моделі вже оцінені. Отже, ми будемо прогнозувати невідоме значення $x_{t+l}, l \geq 1$, при умові, що x_t - останнє спостереження аналізованого ряду на поточний момент, умовно позначимо його як $\hat{x}_t^l$.

Слід відмітити, що хоча $\hat{x}_t^l$ і $\hat{x}_{t-1}^{l+1}$ обидва обозначають прогноз одного й того ж значення x_{t+l}, проте рахуються абсолютно різним чином, бо відповідно є рішеннями різних задач [1].

Ряд x_τ , в рамках ARIMA(p, k, q)-моделі, представимо при $\tau > k$,
у вигляді:

$$\left(1 - \alpha_1 L - \cdots - \alpha_p L^p\right) \sum_{j=0}^{k} (-1)^k C_k^j x_{\tau-1} = \delta_\tau - \theta_1 \delta_{\tau-1} - \cdots - \theta_q \delta_{\tau-q} \quad (1.3)$$

де L – це оператор зсуву значення часового ряду один часовий такт назад.

Із цього можна виразити x_τ для будь-якого $\tau = t - q, \ldots, t-1, t, t+1, t++1$. Отримаємо:

$$x_\tau = \left(\sum_{j=1}^{p} \alpha_j L^j\right) x_\tau + \left(1 - \sum_{j=1}^{p} \alpha_j L^j\right)\left(\sum_{i=0}^{k} (-1)^i C_k^j x_{\tau-i}\right) + \left(1 - \sum_{j=1}^{q} \theta_j L^j\right) \delta_\tau \quad (1.4)$$

Праві частини цих співвідношень являють собою лінійні комбінацій $p + k$ попередніх(відносно лівої частини) значень нашого процесу x_t , доповнених лінійними комбінаціями поточного і q попередніх значень випадкових залишків δ_t [4]. Причому коефіцієнти, за допомогою котрих ці лінійні комбінації рахуються, відомі, оскільки виражаються в термінах уже оцінених параметрів моделі.

Цей факт і дозволяє використати останнє отримане відношення для того, щоб побудувати прогнозні значення часового ряду на l тактів часу вперед. Відповідно до теоретичної бази, найкращим лінійним прогнозом в момент часу t з випередженням l є умовне математичне очікування випадкової величини x_{t+l} , обчислене при умові, що всі значення x_t до моменту часу t. Отриманий результат є окремим випадком загальної теорії прогнозування.

В роботі [1] показано, що умовне математичне очікування $E(x_{t+1}|x_1, \ldots, x_t)$ можна отримати шляхом застосування операцій усереднення до обидвих частин, при $\tau = t + 1$ з урахуванням відношень:

$$E(x_{t-j} \mid x_1, \ldots, x_t) = x_{t-j} \text{ при усіх } j = 0, 1, 2, \ldots, t-1 \quad (1.5)$$

$$E(x_{t+j} \mid x_1, \ldots, x_t) = \hat{x}_t^j \text{ при усіх } j = 1, 2, \ldots; \quad (1.6)$$

$$E(x_{t+j} \mid x_1, \ldots, x_t) = 0 \text{ при усіх } j = 1, 2, \ldots; \quad (1.7)$$

$$E(x_{t-j} \mid x_1, \ldots, x_t) = x_{t-j} - \hat{x}_{t-j-1}^{jl} \text{ при усіх } j = 0, 1, 2, \ldots, t-1; \quad (1.8)$$

Таким чином, для побудови прогнозу можна використовувати наступну процедуру: по формулам (1.4), розраховується ретроспективні прогнози $\hat{x}^{l}_{t-q-1}$, $\hat{x}^{l}_{t-q}$, …, $\hat{x}^{l}_{t-1}$ по попереднім значенням часового ряду. При цьому при обчисленні $\hat{x}^{l}_{t-q+m-1}$ для x_{t-q+m}, $m = 0,1,\ldots$, по формулам (1.4) замість умовних середніх $E(\delta_{t-q+m-j}|x_1,\ldots,x_{t-q+m})$, які в загальному випадку необхідно було би рахувати по формулам (1.8), підставляючи їх безумовні значення, рівню нулю [5]. Використовуючи формули для $\tau > t$ і правила (1.5-1.8) підраховуються умовні математичні очікування для обчислень прогнозних значень.

В доповіді було розглянуто метод побудови ARIMA-моделей для аналізу та прогнозування часових рядів. Дані методи пристосовані для вирішення широкого спектру задач (зокрема, для нестаціонарних часових рядів) й мають високу ефективність роботи.

ЛІТЕРАТУРА

1. Э.Е.Тихонов. Методы прогнозирования в условиях рынка: учебное пособие. Невинномысск: 2006. 221 с.
2. Подкорытова О. А., Соколов М. В. Анализ временных рядов : учеб. пособие для бакалавриата и магистратуры. Юрайт: 2-е изд., 2017. 267 с.
3. Садовникова Н.А., Шмойлова Р.А. Анализ временных рядов и прогнозирование. Московский финансово-промышленный университет «Синергия»: 2016. 152 с.
4. Рунова Л.П., Рунов И.Л. Анализ временных рядов и прогнозирование. Учебно-методические материалы. Ростов-на-Дону: РГУ, 2006.
5. Садовникова Н.А., Шмойлова Р.А. Анализ временных рядов и прогнозирование. Вып. 3: Учебно-методический комплекс. ЕАОИ: 2016. 264 с.

БУРЛАЙ Іскандер Сергійович – студент; Харківський національний університет імені В.Н. Каразіна, проспект Незалежності, 6, Харків, Харківська область, 61000; e-mail: xa11867825@student.karazin.ua; ORCID: 0000000258028183.

Наукові інтереси:

– *аналіз та прогнозування часових рядів.*

БАКУМЕНКО Ніна Станіславівна – доцент кафедри теоретичної та прикладної системотехніки факультету комп'ютерних наук, Харківський національний університет імені В.Н. Каразіна, проспект Незалежності, 6, Харків, Харківська область, 61000; e-mail: n.bakumenko@karazin.ua; ORCID: 0000-0002-5802-8183.

Наукові інтереси:

– *інтелектуальний аналіз даних.*

УДК 321.311 **СЕКЦІЯ 1**

ВАНІН В.А., КРУГОЛ М.М., ЛАЗУРЕНКО О.П.

МАТЕМАТИЧНІ МОДЕЛІ ТА РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ ГІДРАВЛІЧНИХ СИСТЕМ ТЕС

Актуальність задачі

При вироблені електричної енергії на теплових електричних станцій (ТЕС) для забезпечення роботи її основних агрегатів використовуються механізми власних потреб. В основному це відцентрові механізми (ВМ) – насоси та вентилятори. Для приводу таких механізмів використовують в основному асинхронні двигуни (АД) напругою 0,4 та 6 кВ. Частка споживання електричної енергії на власні потреби залежить від типу електричної станції та палива, що використовують на ній, та може досягати 15%, з яких до 90% споживання електричної енергії припадає саме та ВМ.

ВМ забезпечують перекачування робочого середовища – води, повітря, газів, реагентів, тощо – по розгалуженим гідравлічним системам ТЕС. Робота основного устаткування ТЕС у неномінальних режимах в наслідок участі станції в добовому графіку регулювання навантаження, або при роботі ТЕЦ за тепловим графіком навантаження призводить до необхідності регулювання основних параметрів роботи ВМ – їх продуктивності та напору. В свою чергу продуктивність з якою працює ВМ, напір, який він розвиває, його ККД визначають навантаження на валу електропривода, та електричну потужність яку електричний двигун споживає з електричної мережі. Аналіз характеристик асинхронних двигунів та ВМ показує, що дані механізми працюють з максимальним ККД при навантаженнях близьких до номінальних. При регулюванні параметрів ВМ класичними способами, зміною кута відкриття направляючого апарату та дроселюванням, значно знижується ефективність їх роботи, в наслідок чого частка споживання електричної енергії на власні потреби значно зростає.

Для підвищення енергоефективності роботи ТЕС необхідне детально дослідити режими роботи їх гідравлічних мереж та з'ясувати можливості оптимального регулювання роботи механізмів власних потреб. Для вирішення такої задачі необхідно побудувати математичні моделі гідравлічних мереж та ВМ в них.

Математична модель відцентрового механізму, що працює на гідравлічну мережу

На сьогодні широко використовуються математичні моделі ВМ різної складності (диференціальні, алгебраїчні, стаціонарні, нестаціонарні). Кожна з таких математичних моделей використовується для конкретних прикладних задач. В роботі пропонується використовувати закони подібності ВМ для перерахунку їх характеристик при різних режимах роботи. Більш складні математичні моделі ВМ базуються на фундаментальних законах термодинаміки. Використання таких моделей не є доцільним при аналізі статичних режимів роботи ВМ та аналізі ефективності їх регулювання, у зв'язку з їх складністю [1].

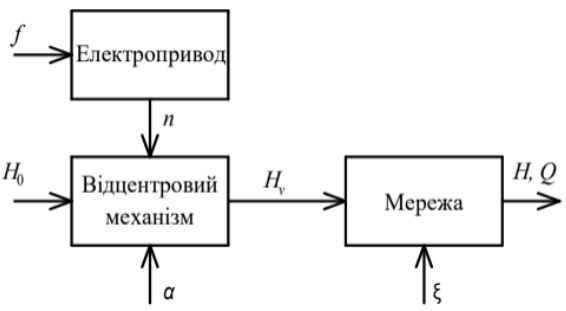


Рис. 1 Схема системи забезпечення власних потреб ТЕС із еквівалентним механізмом

Запропонована математична модель ВМ, що приводиться в дію АД, та працює на гідравлічну мережу показана на рис.1. ВМ та мережа, на яку він працює, утворюють єдину технічну систему «ВМ-гідравлічна мережа», стан якої визначається наступними величинами:

Q – продуктивність механізму, м3/год;

H - напір, що розвиває механізм, м;

n – швидкість обертання робочого колеса, об/хв;

α - кут відкриття направляючого апарату;

© ВАНІН В.А., КРУГОЛ М.М., ЛАЗУРЕНКО О.П., 2021

ξ– еквівалентний гідравлічний опір мережі [2].

В загальному вигляді роботу ВМ на мережу можна описати наступною залежністю

$$\Psi(\alpha,Q,n,\xi)=H(Q,\alpha,n)-H_{\text{M}}(Q,\xi)=0.$$

Математична модель асинхронного двигуна, як споживача електричної енергії, що враховує можливість регулювання частоти живильної напруги побудована з використанням його ККД та потужності, що споживається ним з електричної мережі. ККД асинхронного двигуна при регулюванні частоти має вигляд [3]

$$\eta_{АД}=\frac{P_{мех}}{P_{мех}+\Delta P_{з.м.ном}K^2+\Delta P_{ст.ном}\beta^{\frac{n}{2}+q}+\Delta P_{мх.ном}\beta^2+\Delta P_{\mu ном}\beta^n},$$

де $P_{мех}$ - потужність на валу АД; $\Delta P_{з.м.ном}$ - змінні втрати потужності в АД, Вт; $\Delta P_{мх.ном}$ - номінальні механічні втрати в АД, Вт; $\Delta P_{ст.ном}$ - номінальні втрати в сталі АД, Вт; $\Delta P_{\mu ном}$ - втрати в обмотці статора АД від протікання номінального намагнічувального струму, Вт.

Потужність на валу АД (механічну) та потужність, яку він споживає з електричної мережі можна розрахувати за наступними співвідношеннями

$$P_{\text{мех}}=\frac{Q\rho gH}{1000\,\eta_{\text{н}}},\ P_{АД}=\frac{P_{мех}}{\eta_{АД}(P_{мех})}.$$

Технологічний процес виробництва електричної та теплової енергії на ТЕС є складним, а тому для нормального функціонування її основного устаткування використовується велика кількість допоміжних ВМ, які з'єднані за різними схемами. Основними схемами з'єднання є послідовне, паралельне та змішане.

При моделюванні ВМ, вихідними даними слугують їх паспортні характеристики. Для вентиляторів це залежності напору та ККД механізму від продуктивності та кута відкриття направляючого апарату, а для насосів – залежність напору та ККД від продуктивності при номінальній швидкості обертання, що відповідає частоті живильної напруги електричної мережі – 50 Гц. За паспортними даними механізмів знімаються таблиці (*N*x*M*) значень виду (a_i,Q'_j,H_{ij}) та (a_i,Q'_j,η_{ij}) де $i=1\ldots N,\ j=1\ldots M$, та за допомогою методу найменших квадратів визначається функція інтерполяції напору та ККД механізму у вигляді:

$$\tilde{H}(\alpha,Q',n(f_0))=h_0+h_1\alpha+h_2Q'+h_3\alpha Q'+h_4\alpha^2+h_5Q'^2, \quad (1)$$

$$\tilde{\eta}(\alpha,Q',n(f_0))=k_0+k_1\alpha+k_2Q'+k_3\alpha Q'+k_4\alpha^2+k_5Q'^2. \quad (2)$$

Для відцентрових механізмів характерні наступні закони подібності:

$$\frac{Q}{Q'}=\frac{n}{n_0},\ \frac{\tilde{H}}{H'}=\left(\frac{n}{n_0}\right)^2, \quad (3)$$

де Q – актуальна продуктивність ВМ, м^3/год; Q' - продуктивність ВМ при номінальній частоті живильної напруги, м^3/год; $\tilde{H}$ - актуальний напір, м; H' - напір ВМ при номінальній частоті живильної напруги, м.

З урахування (1) – (3) відтворені вирази для знаходження, відповідно, напору та ККД відцентрового механізму як функції продуктивності, кута відкриття направляючого апарату та частоти живильної напруги електричної мережі:

$$\tilde{H}(\alpha,Q,f)=\left(h_0+h_1\alpha+h_2Q\cdot\left(\frac{n(f_0)}{n(f)}\right)+h_3\alpha Q\cdot\left(\frac{n(f_0)}{n(f)}\right)+h_4\alpha^2+h_5\left[Q\cdot\left(\frac{n(f_0)}{n(f)}\right)\right]^2\right)\left(\frac{n(f)}{n(f_0)}\right)^2,$$

$$\tilde{\eta}(\alpha,Q,f)=k_0+k_1\alpha+k_2Q\cdot\left(\frac{n(f_0)}{n(f)}\right)+k_3\alpha Q\cdot\left(\frac{n(f_0)}{n(f)}\right)+k_4\alpha^2+k_5\left[Q\cdot\left(\frac{n(f_0)}{n(f)}\right)\right]^2.$$

Для побудови рівняння стану групи ВМ на мережу зі складною розгалуженою схемою використовують закони Кірхгофа. Приклад такої системи показаний на рис. 2. Невідомими є витрати через гілки, по яким можна буде розрахувати витрати напорів. Число невідомих приймається по числу гілок схеми, що розглядається [4].

Число рівнянь по першому та другому законах Кірхгофа

$$m_I = m_в - 1,\ m_{II} = m_г - m_в + 1,$$

де $m_в$ – кількість вузлів складної гідравлічної схеми;

$m_г$ – кількість гілок складної гідравлічної схеми.

Розглянемо гідравлічну мережу представлену на рис. 2. Така схема підключень має 2 вузли та 3 гілки, а тому необхідно скласти систему з 3 рівнянь, одне з яких на основі I закону Кірхгофа, та 2 на основі II закону Кірхгофа.

Система рівнянь, що описують роботу складної гідравлічної системи матиме вигляд

$$\omega\left(\vec{f},\vec{\alpha},\vec{Q},\xi\right)=\begin{pmatrix} H_1(Q_1,\alpha_1,f_1)+H_2(Q_2,\alpha_2,f_2)-H_M(Q_1,\xi) \\ H_1(Q_1,\alpha_1,f_1)+H_3(Q_3,\alpha_3,f_3)-H_M(Q_1,\xi) \\ Q_1-Q_2-Q_3 \end{pmatrix}=0. \quad (4)$$

Система рівнянь (4) виступає обмеженнями у задачі знаходження оптимальних параметрів керування ВМ, що працюють на гідравлічну мережу. А саму задачу знаходження оптимальних параметрів керування можна записати у вигляді [5]:

$$\left(\vec{f}^{\,*},\vec{\alpha}^{\,*}\right)=\arg\max_{\substack{f_i\in[f_{i,\min},f_{i,\max}],i=1..N \\ \alpha_i\in[\alpha_{i,\min},\alpha_{i,\max}],i=1..N \\ \omega\left(\vec{f},\vec{\alpha},\vec{Q},\xi\right)=0 \\ Q_1=Q_\Sigma^*}}\frac{\sum_{i=1}^{N}P_i\eta_i\left(\alpha_i(f),Q_i,f\right)}{\sum_{i=1}^{N}P_i}. \quad (5)$$

Додатковими обмеженнями є технічні обмеження на кут відкриття направляючого апарату та частоти живильної напруги, що визначається технічними характеристиками системи.

Дослідження проводилися для 3 ВМ, що включені за схемою рис. 2. Дана група складається одного вентилятора ЦР-40 та двох вентиляторів ВДН-17-3. При заданій продуктивності групи механізмів в 90 тис.м^3/год, були побудовані їх еквівалентні характеристики – рис.3 та знайдені оптимальні параметри керування механізмами :

$$\vec{f}^{\,*}=(43{,}5\quad 45{,}5\quad 45{,}5),\ \vec{\alpha}^{\,*}=(8{,}8\quad 6{,}25\quad 6{,}25).$$

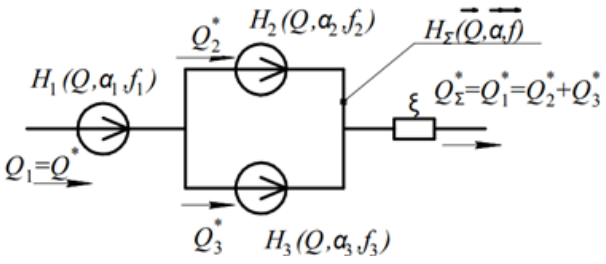


Рис. 2 – Приклад гідравлічної мережі

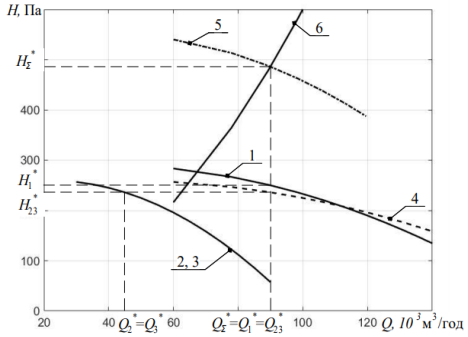


Рис. 3 Результати розрахунку (характеристики)
1,2,3 – характеристики ВМ; 4 – еквівалентна характеристика паралельно працюючих ВМ 2 та 3; 5 – еквівалентна характеристика групи ВМ; 6 – характеристика мережі.

Висновок

Запропоновані математичні моделі відцентрових механізмів та гідравлічних мереж доцільно використовувати при аналізі режимів роботи гідравлічних мереж ТЕС, та техніко-економічних розрахунках перспективних заходів, щодо підвищення енергоефективності роботи ТЕС.

ЛІТЕРАТУРА

1. Кругол Н. М. Математические модели систем обеспечения работы котлоагрегата ТЭС в задаче повышения его энергоэфективности / А. П. Лазуренко, В. А. Ванин, Н. М. Кругол // Вісник НТУ «ХПІ». Сер. : Математичне моделювання в техніці та технологіях. – 2019. – № 8(1333). – С. 41–48.
2. Kruhol M. Assessment of group regulation feasibility in thermal power plant auxiliaries capacity control / M. Kruhol, V. Vanin, O. Lasurenko // Eastern-European Journal of Enterprise Technologies. – 2020. – Vol. 6 (8 (108)). – P. 45–53.
3. Фираго Б. И. КПД асинхронного электродвигателя при частотном управлении с различными видами нагрузок / Б. И. Фираго, Л. Павлячик // Энергетика. Известия высших учебных заведений и энергетических объединений СНГ. – 2001. – № 3 – С. 52–59.
4. Ванін В. А. Математичне моделювання одногрупового газогідравлічного тракту допоміжних механізмів парового котла теплової електричної станції / В. А. Ванін, М. М. Кругол, О. П. Лазуренко // Вісник НТУ «ХПІ». Сер : Математичне моделювання в техніці та технологіях. – 2020. – № 1. – С. 3–14.
5. An Algebraic Model of Gas-Hydraulic Network of Mechanisms with Electric Drive in the Problem of Thermal Power Plant Auxiliaries Optimization / N. Kruhol, O. Lasurenko, V. Vanin, R. Tomashevskiy // 2020 IEEE KhPI Week on Advanced Technology (KhPIWeek). – Kharkiv, 2020.– P. 182–192.

ВАНІН Віктор Антонович – д.т.н., професор, професор кафедри вищої математики, Національний технічний університет «Харківський політехнічний інститут», вул. Кирпичова 2, Харків 61002, Україна, e-mail: vvaplb5256@gmail.com; ORCID: 0000-0002-3523-7505.

КРУГОЛ Микола Михайлович – аспірант кафедри електричних станцій, Національний технічний університет «Харківський політехнічний інститут», вул. Кирпичова 2, Харків 61002, Україна, e-mail: kruhgol@gmail.com; ORCID: 0000-0002-6090-4875.

ЛАЗУРЕНКО Олександр Павлович – к.т.н., доцент, завідувач кафедри електричних станцій, Національний технічний університет «Харківський політехнічний інститут», вул. Кирпичова 2, Харків 61002, Україна, e-mail: Lasurenko@kpi.kharkov.ua; ORCID: 0000-0002-4409-629X.

УДК 004.35: 004.9 **СЕКЦІЯ 7**

ВАРЛАМОВА Н.В., СТЄРВОЄДОВ М.Г.

ІНФОРМАЦІЙНО-ВИМІРЮВАЛЬНА ПСИХОДІАГНОСТИЧНА СИСТЕМА: ПРИНЦИПИ І МЕТОДИ ПОБУДОВИ

Введение

Сучасна концепція Інтернету речей спрямована на створення віддалено керованих систем і технічних засобів для збору, зберігання і обробки даних. Не дивлячись на це, розробки в області психодіагностичного обладнання представлені як стаціонарні системи, призначені для локальної роботи з даними. Така локальна особливість роботи системи значно обмежує можливості системи.

З огляду на, що метою роботи інформаційно-вимірювальної системи є психологічна діагностика людини (психофізіологічна, профвідбір, тренування когнітивних процесів і т.д.), то, відповідно, передбачається створення профілю людини. Профіль людини базується, з одного боку, на об'єктивних вимогах до досліджуваного явища (наприклад, вимоги до професії), а, з іншого боку, на його психологічних і психофізичних показниках. Передбачається розглядати профілі випробовуваних як класи з заданими загальними ознаками. Також передбачається існування проблеми нечіткості класів через варіативності як вимог, діагностичних методів, так і експертної оцінки. Тому необхідним є визначення належності об'єкта (випробуваного) до певного класу.

Тому, актуальною є розробка концептуально і принципово нових інформаційно-вимірювальних систем для віддалених психодіагностичних досліджень.

Аналіз літературних джерел

Існуючі на сьогодні розробки в області психодіагностичного приладобудування, з одного боку, мають ряд загальних принципів:

– використання моделей послідовної цілеспрямованої діяльності, просторово-часового прогнозування, діяльності людини в експериментальному середовищі;

- реалізація принципів системного підходу;

- реалізація набору методик.

Проте, такі розробки мають також і загальні проблеми. Серед таких загальних проблем можно зазначити наступні:

– опис інформаційно-вимірювальних засобів у вигляді їх архітектури, структурних схем або процедури при відсутній концептуальній моделі,

– відсутність повноцінної можливості віддаленого управління системою;

– практична відсутність можливості використання локальної мережі для передачі і збереження даних, тому як збереження даних реалізується локальною базою даних.

Тому раціонально запропонувати створення інформаційно-вимірювальної психодіагностичної системи, що базується на новых принципах. Для розробки даної системи були застосовані наступні принципи:

– віддалена взаємодія між компонентами системи завдяки мережі;

– застосування стандартних і нестандартних (та інтеграція нових) засобів діагностики;

– застосування технологій WiFi и GPRS для передачі даних;

– побудова профілю випробуваного нового типу;

– безпосередня присутність експерта не є обов'язковою вимогою;

– використання стандартних і нестандартних шкал.

Для проектування та реалізації інформаційно-вимірювальної системи необхідне застосування облачного серверу, сучасних технологій передачі даних, сучасних елементів мікроконтролерної техніки.

© ВАРЛАМОВА Н.В., СТЄРВОЄДОВ М.Г., 2021

Принципи і методи побудови інформаційно-вимірювальної системи

Була розроблена структура інформаційно-вимірювальної системи, завдяки сформованому переліку наступних вимог:

– наявність функціональних підсистем (блоків) всередині системи;

– відповідність елементів системи функціональним рівням моделі проекту Інтернету речей;

– можливість інтеграції нових елементів діагностики;

– резервування ресурсів системы;

– використання програмних засобів для обробки, представлення і візуалізації даних;

– наявність різних сценаріїв відповідності профіля заданим класам.

Структура загальної системи містить взаємодіючі підсистеми - блок управління, блок тестування, блок отримання і передачі даних, блок обробки і зберігання даних.

Розроблений блок управління призначений для використання експертом з метою проведення реєстрації випробовуваних, проведення процедури тестування, перегляду отриманих результатів та формування експертної оцінки, а також інженером - для контролю за проведення процедури тестування. Компонентами блоку управління є ПК, мобільний додатки і його інтерфейс.

Блок тестування містить дво- та тривимірні засоби представлення сигналів, мікрофон, клавіатуру.

Мікроконтролер STM32F4, як керуючий компонент системи, модулі WiFi і GSM і роутер здійснюють обмін даними завдяки AT-командам і протоколу TCP / IP. У випадках, що унеможливлюють роботу модуля WiFi, система використовує пакетну передачу GSM через станції, шлюз до глобальної мережі.

Система передбачає різноманітні режими і сценарії роботи, за вибір яких відповідає мобільний додаток, а за виконання – ПО мікроконтролера STM32F4. Первісні результати діагностування, а саме, факт натискання кнопки та час, фіксують спеціалізовані лічильники.

Для роботи із хмарним сервером система користується протоколом MQTT, а для візуалізації даних сервера можна застосовувати математичний пакет MATLAB. Робота з базою даних здійснюється за допомогою мобільного додатку і інтерфейсу СУБД.

З огляду на те, що інформаційно-вимірювальна система є проектом Інтернету речей, то відповідність елементів системи рівням концептуальної моделі Інтернету речей виглядає наступним чином :

- галузі «психологія», «ринок праці» відповідають рівню додатків;

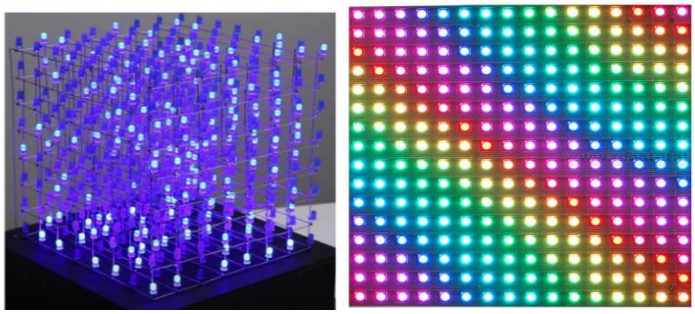
- хмарний сервер Thingspeak, протокол MQTT, СУБД можно віднести до рівня підтримки додатків і послуг;

- функції технологій GPRS і WiFi, LTE, Ethernet, мікроконтролера STM32F4, модуля SIM800L, модуля ESP8266, роутера відповідають вимогам мережевого рівня;

- рівню пристроїв - панелі для відображення двовимірних світлових, звукових, тактильних і комбінованих сигналів, 3D пристрої пред'явлення стимулів (LED-куб, LED-куля), мікрофон, клавіатура.

Пристрої подання світлових сигналів.

Макети пристроїв (рис. 1) для візуальної діагностики розроблені на у вигляді панелей з матриць триколірних адресних цифрових RGB світлодіодів з вбудованими контролерами WS2812B на гнучкій підкладці - для представлення лінійних і двовимірних стимулів, LED-куб і LED-куля - для об'ємного представлення стимулів.



а) *б)*

Рис. 1. Макети психодіагностичних пристроїв: а) LED-куб розміром 8 × 8 × 8 світлодіоді, б) панелі з адресних RGB світлодіодів WS2812B

Метод мониторинга психологического и психофизического состояния испытуемого

Моніторинг психологічного стану має 3 етапи: формування блоку даних, формування первинної експертної оцінки і формування остаточної експертної оцінки (рис. 2).

Для визначення профілю людини застосовуються методи кластерного аналізу, які дозволяють достатньо точно виділити профілі як кластер (класс) або типологічні групи. Наприклад, ієрархічний метод кластеризації застосовувався для визначення кількості профілів (кластерів) .

Одною із функціональних переваг інформаційно-вимірювальної системи є можливість системи оцінювати вплив різних видів навантажень на психологічний або психофізичний стан людини. Такий метод оцінки допомагає прогнозувати і дозувати робочі завантаження. Специфіка методу полягає в визначенні матриці перетворення *Mt.* Вважається, що результатом перетворень матриці початкових станів X є матриця кінцевих станів *Y. Тому,*

$$M_{t_{слух}} = B_{слух} \cdot A_{слух}^{-1},\ M_{t_{зор}} = B_{зор} \cdot A_{зор}^{-1},\ M_{t_{інші}} = B_{інші} \cdot A_{інші}^{-1} \quad (1)$$

де A – корінь кореляційної матриці початкових показників, а *B* – відповідно,кінцевих.

Сценарій 1 характеризується визначенням необхідних параметрів в групі досліджуваних і групі представників профілю, їх нормалізацією та ранжуванням експертами. Показники рівня відповідності параметрів профілю визначаються як

$$Y_I = \frac{I \cdot 100\%}{R_{max}} \quad (2)$$

$$Y_R = \frac{R \cdot 100\%}{R_{max}} \quad (3)$$

де Y - значення рівнів відповідності людини профілем в досліджуваних групах в процентному співвідношенні, де I, R - чисельні значення рівня в групах, Rmax - максимальне значення рівня відповідності.

Сценарій 2 передбачає застосування парних експертних порівнянь параметрів. Завдяки, таким парним порівнянням формується матриця, елементи якої нормуються та визначаються сума її елементів та її вага. Надалі формується матриця M, елементами якої є ступені приналежності параметрів

Висновки

У роботі запропоновано побудову інформаційно-вимірювальної психодіагностичної системи з функцією віддаленого управління, що базується на сучасних концептуально нових принципах. Такі принципи дозволяють проводити психодіагностичне дослідження в умовах непрацездатності для локальних психодіагностичних систем: відсутності кабельного зв’язку із

глобальною мережею, неможливості безпосередньої присутності експерта у момент дослідження та інших.

Запропоновані методи аналізу отриманих результатів дозволять розробити більш гнучку процедуру діагностування і створення особистого профілю людини.

ЛІТЕРАТУРА

1. Балин, В. Д., Гайда, В. К., Гербачевский, В. К. и др. Практикум по общей, экспериментальной и прикладной психологии: СПб.: Питер, 2003. 560 с.
2. Єна, А. І., Кальніш, В. В. Принципи професійного психофізіологічного відбору. *Гігієна праці*.2001. Вип. 32. С.131–144.
3. Варламова, Н.В., Лазурик, В.Т., Стєрвоєдов, М.Г. Модель та апаратно-програмна реалізація інформаційно-вимірювальної системи для психофізичних і психофізіологічних досліджень. *Вісник Харківського національного університету імені В.Н. Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. 2019. Вип. 44, С. 16-22.
4. Lazurik, V.T., Styervoyedov, N.G., Varlamova, N.V. Information processing system for psychophysical research with two- and three-dimensional presentation of test signals. *Control systems and computers. 2020.* Issue 4, P. 66-74.
5. Росляков, А.В. Р75 Интернет вещей: учебное пособие. Самара: ПГУТИ, 2015. 200 с.
6. Николаев, П. Л. Применение облачных технологий в системах умного дома. *Научный журнал «Молодой ученый»*. 2014. Вып. 13. С. 37-39.
7. Сайківська, Л.Ф. Розробка та використання інформаційної технології для оцінки функціонального стану оператора зорового профілю. *Технологический аудит и резервы производства*. 2015. Вып. 4(2). С. 45-49.
8. Чернявская, Е.В. Применение теории нечеткой логики для оценки профессиональной пригодности школьников. *Вестник НГУ. Серия: Педагогика*, *12*. 2011. Вып. 2. С. 66-71.
9. Бурилич, И.Н., Уварова, А.Г., Филист, С.А. Автоматизированная система диагностики шизофрении на основе нечеткой логики принятия решений. *Вестник новых медицинских технологий*. 2006. Вып. 13(2). С. 46-49.
10. Дьяконов, В. П. MATLAB и SIMULINK для радиоинженеров. М.: «ДМК-Пресс», 2011. 976 с.

СТЄРВОЄДОВ Микола Григорович – К. т. н., доцент, завідуючий кафедрою електроніки і управляючих систем; Харківський національний університет імені В. Н. Каразіна; майдан Свободи, 6, м. Харків, Україна, 61022; E-mail: keus@karazin.ua; ORCID: http://orcid.org/0000-0003-0136-6437.

Наукові інтереси:

– *проектування систем автоматичного управління, схемотехніка електронних обчислювальних машин.*

ВАРЛАМОВА Наталя Володимирівна – Викладач кафедри електроніки та управляючих систем; Харківський національний університет імені В. Н. Каразіна; майдан Свободи, 6, м. Харків, Україна, 61022; E-mail: natess123@gmail.com, ORCID: http://orcid.org/0000-0001-5117-7293**.**

Наукові інтереси:

– *Інтернет речей, психологія.*

UDC 539.3 **SECTION 1**

VIERUSHKIN I.A.

HYPERSINGULAR INTEGRAL EQUATION METHOD IN NUMERICAL SIMULATING FREQUENCIES AND MODES OF CIRCULAR PLATE IMMERSED INTO LIQUID

1 Introduction

Problems of hydroelastic interaction of structural elements with the environment are of great interest to many scientists in recent decades. These problems are conventionally divided into two classes. The first includes the task of determining the dynamic characteristics of such structures, inside which there is a liquid, i.e between the surfaces of the structural element is a one-way contact with the liquid. Works [1-2] are devoted to these issues, where the problems of liquid splashes for rigid shell systems are considered. In article [3], the oscillations of elastic reservoirs partially filled with liquid were analyzed, and in [4] the free oscillations of elastic lids of a hydroturbine during interaction with liquid were considered.

The second class of questions concerns the determination of the dynamic characteristics of elastic thin structures, and in which the bearing surfaces are in bilateral contact with the liquid. For this case, it is possible to determine the blades of the impellers Francis and Kaplan turbines, the blades of wind power stations, the wings of aircraft, etc.

Powerful computational methods have been developed to numerically solve such problems. Among them are the method of finite differences, the methods of finite and boundary elements [5], the method of R functions.

But each new design, working in conditions of intense power loads, in interaction with the environment, requires careful analysis of strength and dynamic characteristics. This will lead to the need of creation new and improving existing computational methods that would take into account the specific features of the structure under analysis.

In the article, a new effective method for solving hypersingular integral equations on a circular domain has been developed. It has also been used to determine the frequencies and shapes of free oscillations of a circular plate immersed in a liquid.

2 Formulation of the problem

The spring of a thin round plate which is immersed in liquid is analyzed. Kirchhoff-Love hypotheses should be used to model the motion of the plate [6].

We introduce the following notation: E is the Young's modulus, ν is the Poisson's ratio, and h is the plate thickness.

With a constant thickness of a homogeneous plate, the equation of motion of the plate [6] has the form

$$D\Delta\Delta w + \rho_p h \frac{\partial^2 w}{\partial t^2} = q(x, y.t), \tag{2.1}$$

where $w(x,y,t)$ - deflection of the plate;ρ_p is the density of the plate material; $D = \frac{Eh^3}{12(1-\nu^2)}$ - cylindrical rigidity; $q(x,y,t)$ - force acting on the plate.

If the plate is immersed in the liquid, then

$$q(x,y,t) = p(x,y,t)\mathbf{n} + q_0(x,y,t), \tag{2.2}$$

Where $p(x,y,t)$ - the pressure drop of the liquid on the plate; $\mathbf{n}$ - unit normal to the surface of the plate; $q_0(x,y,t)$ - a disturbing force.

If the plate is immersed in the liquid, then

$$q(x,y,t) = p(x,y,t)\mathbf{n} + q_0(x,y,t), \tag{2.2}$$

© VIERUSHKIN I.A., 2021

where $p(x,y,t)$ - the pressure drop of the liquid on the plate; $\mathbf{n}$ - unit normal to the surface of the plate; $q_0(x,y,t)$ - a disturbing force.

If in equation (2.2) the function $q_0(x,y,t)$=0, then we have the case of free hydroelastic oscillations.

To find the pressure $p(x,y,t)$, we make the following assumptions: the fluid is ideal and incompressible, and its motion is vortex-free. In this case, there is a velocity potential φ(*x*,*y*,*z*,*t*), the gradient of which is the velocity of the liquid. This potential satisfies the Laplace equation

The relationship between velocity potential and pressure is determined from the linearized Cauchy-Lagrange integral

$$p^{\pm} = -\rho_l \frac{\partial \varphi^{\pm}}{\partial t} + p_0$$

where ρ_l is the density of the liquid; p_0 - atmospheric pressure.

The limit values $\varphi^{\pm}$ correspond to the positive and negative values of the z coordinate, respectively.

Thus, for the pressure drop we have

$$p = p^{+} - p^{-} = -\rho_l \left(\frac{\partial \varphi^{+}}{\partial t} - \frac{\partial \varphi^{-}}{\partial t} \right). \tag{2.4}$$

We formulate the boundary conditions for the differential equation (2.3). According to the condition of non-penetration we obtain the following equations on the surfaces σ of the plate:

$$\left. \frac{\partial \varphi^{\pm}}{\partial \mathbf{n}} \right|_{\sigma} = \frac{\partial w}{\partial t}. \tag{2.5}$$

The system of differential equations (2.1), (2.3) with boundary condition (2.5) must also be supplemented by the conditions of fixation, in other words by the conditions with respect to the function $w(x,y,t)$ on the plate contour.

Thus, we arrive at a related problem with respect to two unknown functions $w(x,y,t)$ and φ(x,y,z,t), while the right-hand side of the differential equation $w(x,y,t)$ with respect to time derivative from φ(x,y,z,t), and in boundary conditions for the Laplace equation with respect to φ(x,y,z,t) is a time derivative of an unknown function $w(x,y,t)$.

3 The method of given shapes

To solve the problems of natural and forced oscillations of elastic plates immersed in a liquid, the method of given shapes was used. In the General case, we will look for the movement of the plate immersed in the liquid in the form of:

$$w(x,y,t) = \sum_{k=1}^{N} c_k(t) w_k(x,y), \tag{3.1}$$

where $c_k(t)$ - unknown coefficients that depend only on time; $w_k(x,y)$ - forms of oscillations of the plate without taking into account the interactions with the liquid; *N* - the number of forms retained in the calculations.

Given (2.4), (3.1) we see that for the velocity potential φ(x,y,z,t) it is necessary to choose the following image

$$\varphi(x,y,z,t) = \sum_{k=1}^{N} \dot{c}_k(t) \varphi_k(x,y,z), \tag{3.2}$$

in which functions $\varphi_k(x,y,z)$ are solutions of such boundary value problems

$$\Delta \varphi_k = 0, \quad \frac{\partial \varphi_k}{\partial \mathbf{n}} = w_k. \tag{3.3}$$

Note that in boundary value problems (3.3) the right-hand sides in boundary conditions are known.

Next, we describe the method of reducing the problem of determining the frequencies and forms of oscillations of the "plate-liquid" system to hypersingular equations.

4 Hypersingular equation to determine the velocity potential and fluid pressure

To find the fluid pressure drop on the surfaces of the plate, we have equation (2.4). According to equation (2.4), this function has a finite gap in the specified area.

As is known, such properties are inherent in the potential of the double layer. This potential has a form

$$\varphi(\mathbf{P}_0)=\frac{1}{4\pi}\iint_S \Gamma(\mathbf{P})\frac{\partial}{\partial \mathbf{n}}\frac{1}{|\mathbf{P}-\mathbf{P}_0|}dS,\quad \mathbf{P}\in S, \tag{4.1}$$

where S is the area occupied by the circular plate; $\mathbf{n}$ is the unit normal to the surface S; $\mathbf{P}$ and $\mathbf{P}_0$ are points of three-dimensional space with coordinates (x,y,z) and (x_0,y_0,z_0), respectively; $|\mathbf{P}-\mathbf{P}_0|=\sqrt{(x-x_0)^2+(y-y_0)^2+(z-z_0)^2}$ is a Cartesian distance between $\mathbf{P}$ and $\mathbf{P}_0$; $\Gamma(\mathbf{P})$ is a potential density, a function given on the surface S.

The function defined by formula (4.1) satisfies the Laplace equation (2.3), has a continuous normal derivative, and at the intersection of the surface S along the normal has a finite discontinuity, namely

$$\varphi^+(\mathbf{P}_0)-\varphi^-(\mathbf{P}_0)=\Gamma(\mathbf{P}_0),\quad \mathbf{P}_0\in S.$$

Therefore, we use representation (4.1) for the velocity potential. Note that to determine the pressure drop, it is necessary to find an unknown density $\Gamma(\mathbf{P})$ on the surface of the plate. From representation (3.2) we obtain that for this we need to find solutions of boundary value problems (3.3). For the functions φ_k we use the image (4.1). The Laplace equation with (3.3) is satisfied, and the boundary condition leads to the integral equation in the form

$$\frac{\partial}{\partial \mathbf{n}_0}\frac{1}{4\pi}\iint_S \Gamma_k(\mathbf{P})\frac{\partial}{\partial \mathbf{n}}\frac{1}{|\mathbf{P}-\mathbf{P}_0|}=w_k(\mathbf{P}_0),\quad \mathbf{P},\mathbf{P}_0\in S \tag{4.2}$$

5 Construction of the matrix of attached masses

After determining the forms of natural oscillations $w_m(\rho)$ of a circular plate by formulas we find solutions of hypersingular equations

$$\frac{1}{\pi}\int_0^R \Gamma_m(\rho)\frac{\rho \mathsf{E}(k)d\rho}{(\rho+\rho_0)(\rho-\rho_0)^2}=-w_m(\rho_0)$$

according to the above method of boundary elements.

We look for a solution to the problem of hydroelastic oscillations of the plate in the form of series (3.1), (3.2).

Consider equation (2.1) in the assumption that $q_0(x,y,t)$ = 0, and the pressure drop is from equation (2.4), and we have the following relations

$$D\Delta\Delta w+\rho_p h\frac{\partial^2 w}{\partial t^2}=-\rho_l\left[\frac{\partial\varphi^+(x,y,t)}{\partial t}-\frac{\partial\varphi^-(x,y,t)}{\partial t}\right],$$

$$D\sum_{m=1}^{N}c_m(t)\Delta\Delta w_m+\rho_p h\sum_{m=1}^{N}\ddot{c}_m(t)w_m=-\rho_l\sum_{m=1}^{N}\ddot{c}_m(t)\Gamma_m. \tag{5.1}$$

Since free hydroelastic oscillations will be considered, we believe that

$$c_m(t)=C_m\exp(i\omega t), \tag{5.2}$$

where ω is the frequency of natural hydroelastic oscillations; i - imaginary unit; C_m - became unknown.

$$D\sum_{m=1}^{N}C_m\alpha_m^4 w_m-\omega^2\left[\rho_p h\sum_{m=1}^{N}C_m w_m+\rho_l\sum_{m=1}^{N}C_m\Gamma_m\right]=0. \tag{5.3}$$

The next step lays in fining a scalar product of equation (5.3) on functions $w_n(\rho),\quad n=1,2,...N$ and obtaining a system of linear homogeneous algebraic equations with respect to constants

$$\sum_{m=1}^{N}\left\{\delta_{nm}A_m D\alpha_m^4-\omega^2\left[\rho_p h A_m\delta_{nm}+\rho_l(\Gamma_m,w_n)\right]\right\}C_m=0 \tag{5.4}$$

To find the nonzero solution of equations (8.4) we obtain the characteristic equation with respect to ω

$$\det\left\{\rho_p h\Omega_m^2 A_n\delta_{nm}-\omega^2\left[\rho_p hA_m\delta_{nm}+\rho_l\left(\Gamma_m,w_n\right)\right]\right\}=0 \qquad (5.5)$$

Thus, the problem of determining the frequencies and shapes of the natural hydroelastic oscillations of a round plate is reduced to solving the problem of natural values.

$$\left(\mathbf{A}-\omega^2\mathbf{B}\right)\mathbf{C}=0\,,$$

where the elements of the matrices are determined by formulas

$$a_{mn}=\Omega_m^2\rho_p hA_m\delta_{mn},\quad b_{mn}=\rho_p hA_m\delta_{mn}+\rho_l\left(\Gamma_m,w_n\right).$$

The elements of the matrix of the attached masses of the liquid are determined by the formulas

$$g_{mn}=\left(\Gamma_m,w_n\right)=\tau\sum_{i=1}^{N_1}\rho_{i0}\Gamma_m\left(\rho_{i0}\right)w_n\left(\rho_{i0}\right),$$

that is, we use the formula of central rectangles.

Thus, a method for studying the determination of frequencies and shapes of free oscillations of a circular plate immersed in a liquid based on the use of one-dimensional hypersingular equations has been developed.

REFERENCES

1. Ibrahim R. A., Liquid sloshing dynamics: theory and applications. *Cambridge University Press*, 2005. URL: https://www.researchgate.net/publication/259815818_Liquid_Sloshing_Dynamics_Theory_and_Applications_by_Raouf_A_Ibrahim
2. Gavrilyuk, I., M. Hermann, Lukovsky I., Solodun O., Timokha, A. Natural Sloshing frequencies in Truncated Conical Tanks. *Engineering Computations*, vol. 25, no. 6, pp.518 – 540, 2008. URL: https://www.researchgate.net/publication/245338809_Natural_sloshing_frequencies_in_rigid_truncated_conical_tanks
3. Degtyarev, K., Glushich, P., Gnitko, V., Strelnikova, E. Numerical Simulation of Free Liquid-Induced Vibrations in Elastic Shells. *International Journal of Modern Physics and Applications*. Vol. 1, No. 4, pp. 159-168, 2015. DOI: 10.13140/RG.2.1.1857.5209 URL: https://www.researchgate.net/publication/280728146_Numerical_Simulation_of_Free_Liquid-Induced_Vibrations_in_Elastic_Shells
4. T. Medvedovskaya, E. Strelnikova, K. Medvedyeva Free Hydroelastic Vibrations of Hydroturbine Head. *Intern. J. Eng. and Advanced Research Technology (IJEART)*, vol. 1, no 1. pp.45 – 50, 2015. DOI 10.13140/RG.2.1.3527.4961 URL: https://www.researchgate.net/publication/282868308_Free_Hydroelastic_Vibrations_of_Hydroturbine_Head
5. Brebbia, C.A, Telles, J.C.F & Wrobel, L.C., Boundary element techniques: theory and applications in engineering. Springer-Verlag: Berlin and New York, 1984. URL: https://studizba.com/files/show/djvu/1932-1-brebbiya-k-telles-zh-vroubel-l--metody.html
6. Strelnikova E., Gnitko V., Krutchenko D., Naumemko Y. Free and forced vibrations of liquid storage tanks with baffles J. Modern Technology & Engineering Vol.3, No.1, 2018, pp.15-52. URL: http://jomardpublishing.com/UploadFiles/Files/journals/JTME/V3No1/StrelnikovaE.pdf

ВЄРУШКІН Іван Олексійович – аспірант, Інститут проблем машинобудування ім. А. М. Підгорного НАНУ, вул. Пожарського, 2/10, Харків, 61046, Україна;
e-mail: ivanveruskin86@gmail.com; ORCID: 0000-0002-3837-5567.
Наукові інтереси: Механика сплошной среды, гиперсингулярные уравнения

VIERUSHKIN Ivan Alekseevich – PhD student, A. Podgorny Institute of Mechanical Engineering Problems NASU, Pozharsky, st. 2/10, Kharkiv, 61046, Ukraine;
e-mail: ivanveruskin86@gmail.com; ORCID: 0000-0002-3837-5567.
Research interests: Solid mechanics, hypersingular equations

УДК 004.056 **СЕКЦІЯ 5**

ВІЛІГУРА В.В.

СИСТЕМАТИЗАЦІЯ ЗАГРОЗ І ВРАЗЛИВОСТЕЙ ХАРАКТЕРНИХ ДЛЯ БАЗ ДАНИХ І СУБД

Вступ

При створенні більшості інформаційних систем (ІС), в тому числі їх основного функціонального компонента - бази даних (БД) виникає проблема забезпечення їх захищеності від різного роду погроз. Система захисту інформації - це комплекс програмних, технічних, криптографічних, організаційних та інших методів, засобів і заходів, що забезпечують цілісність, конфіденційність, автентичність і доступність інформації в умовах впливу на неї загроз природного або штучного характеру. І процес створення такої системи дуже складний. З'являється безліч питань, які необхідно вирішити. В тому числі таких як: наскільки адекватні діючі механізми, що забезпечують безпеку, існуючих ризиків і чи потрібні нові; чи є в системі помилки, вразливості, що дозволяють зловмисникам зламати систему захисту; як оцінити рівень захищеності; які контрзаходи дозволять реально підвищити рівень захищеності і багато інших. Відповіді на ці питання далеко не очевидні. Уміння оцінювати і управляти ризиками, знання можливих загроз і вразливостей, володіння методами аналізу і спеціалізованим інструментарієм, це далеко не повний перелік якостей, якими повинні володіти як проектувальники безпечних ІС, БД, так і експерти, які проводять аналіз захищеності, при виробленні обґрунтованих рекомендацій. І в цьому відношенні важливу роль може зіграти відома модель системи безпеки з повним перекриттям [1], що традиційно вважається основою формального опису систем захисту. Її використання дозволяє оцінити захищеність інформаційної системи (в тому числі і БД), розрахувати витрати на побудову системи захисту для неї, визначити оптимальний варіант побудови системи інформаційної безпеки, а також порівняти захищеність різних ІС, БД. Але для того, щоб повною мірою можна було б скористатися всім цим її потенціалом, в першу чергу необхідно визначити, систематизувати перелік актуальних загроз і вразливостей для досліджуваної системи, що захищається, так як в моделі розглядається взаємодія «області загроз», «області, що захищається» та «системи захисту». Тому нижче, визначившись з досліджуваної системою, що захищається, в ролі якої виступає база даних і СУБД, визначимо і систематизуємо відомі і актуальні для них загрози і вразливості, які існують незалежно один від одного. Хоча при реалізації загрози може використовуватися одна або більше вразливостей активу (сутності, що представляє цінність для власника об'єкта оцінки [2]), а один тип вразливості може призвести до безлічі загроз безпеки різної спрямованості.

Основні загрози безпеки баз даних

Узагальнюючи багаторічний досвід різних міжнародних дослідницьких груп і компаній [3-8] – фахівців в області безпеки БД, можна виділити наступні основні, актуальні на сучасному етапі, великі загрози (типи загроз) безпеки баз даних, носіями яких є різні джерела (в даному випадку інтерес більшою мірою представляють антропогенні джерела загроз, які обумовлені діями суб'єктів):

– надмірні і невикористовувані привілеї. Коли хто-небудь отримує привілеї, які перевищують вимоги необхідні для виконання посадових обов'язків, виникає ймовірність зловживання цими привілеями;

– зловживання законними привілеями. Існує ймовірність використання користувачами своїх легітимних прав доступу в протиправних цілях;

– ін'єкції вводу;

– шкідливе програмне забезпечення (ПЗ) – malware. Не знаючи, про інфікування шкідливим ПЗ, законні користувачі стають посередниками, за допомогою яких зловмисники отримують доступ до важливих даних.

– слабкі аудиторські сліди або недостатність заходів щодо аудиту даних;

– незахищеність носіїв (резервних копій) інформації;

© ВІЛІГУРА В.В., 2021

- експлуатація вразливих, невірно сконфігурованих баз даних;
- некеровані конфіденційні дані;
- логічний висновок (inference);
- відмова в обслуговуванні;
- недолік знання і досвіду з питань інформаційної безпеки і деякі інші.

Дані типи загроз є одними з найнебезпечніших для підприємств і організацій будь-яких форм власності, практично будь-якої галузі діяльності людини: банки, фінанси, медицина, торгівля, високі технології, промисловість, транспорт, державні органи, силові структури, освіта, муніципальні установи і т. д.

Основні типи вразливостей характерні для систем баз даних

На основі проведеного аналізу існуючих таксономій вразливостей (vulnerabilities), що мають відношення до конкретного екземпляру продукту або системи, які можуть бути безпосередньо використані зловмисником для реалізації загроз безпеки [9], загальних слабких місць (weakness) програмного і апаратного забезпечення, які можуть привести до виникнення вразливостей [10, 11], а також деяких інших класифікацій вразливостей і недоліків безпеки активів [12-14] був визначений перелік основних загальних слабких місць (недоліків), як деяких типів вразливостей. За основу була взята специфікація з Common Weakness Enumeration (CWE), точніше класифікація абстрактного уявлення Концепції дослідження (Research Concepts) CWE [15], що використовується академічними дослідниками, аналітиками вразливостей, постачальниками засобів оцінки. У число основних типів вразливостей, з урахуванням специфіки розглянутих аспектів, зумовлених характерними особливостями забезпечення безпеки, властивими базам даних і СУБД (не беручи до уваги можливості реалізації загроз за допомогою вразливостей, пов'язаних з недоліками в програмному забезпеченні, архітектурі і конфігурації мереж і операційних систем), увійшли:

1) *неправильне (невідповідне) управління привілеями*: неправильне призначення привілеїв, підвищення (ескалація) привілеїв, виконання операцій з зайвими привілеями;

2) *неправильна авторизація*: неправильне призначення дозволів для критичного ресурсу, відсутня авторизація, некоректна авторизація, розкриття конфіденційної інформації через метадані, розкриття конфіденційної інформації за допомогою запитів даних. Не виконується або неправильно виконується перевірка авторизації, коли суб'єкт намагається отримати доступ до ресурсу або виконати деяку дію;

3) *неправильна автентифікація*: слабкий пароль, застарілий пароль, обхід автентифікації, неправильна реалізація алгоритму автентифікації, невідповідний термін дії сеансу, використання хешу пароля замість пароля для автентифікації і т. д.;

4) *неконтрольоване споживання ресурсів*: належним чином не контролюється розподіл обмеженого ресурсу, тим самим дозволяючи суб'єкту впливати на кількість споживаних ресурсів, що в кінцевому підсумку призводить до їх вичерпання;

5) *зберігання конфіденційної інформації у відкритому вигляді*;

6) *недостатня якість шифрування*;

7) *неправильна очищення конфіденційних даних з виведеного з експлуатації пристрою*: очищення може бути відсутнім, бути недостатнім або некоректним;

8) *використання зламаного або небезпечного криптографічного алгоритму*: використання нестандартного, з не доведеною стійкістю криптографічного примітиву;

9) *використання недостатньо випадкових значень*;

10) *недостатня перевірка достовірності даних*: завантаження коду без перевірки цілісності, неправильна перевірка (відсутність перевірки) значення контрольної суми, неправильна перевірка (відсутність перевірки) криптографічного підпису;

11) *неправильна помилки під час введення*: неправильна перевірка синтаксичної правильності вхідних даних, неправильна перевірка зазначеного типу вхідних даних, неправильна перевірка узгодженості вхідних даних, неправильна перевірка небезпечної еквівалентності вхідних даних (наприклад, простий механізм захисту XSS може спробувати перевірити, що вхід не має тегів <script>, використовуючи зіставлення з урахуванням регістра, але оскільки HTML не чутливий до регістру при обробці веб-браузерами, зловмисник може ввести <ScrIpT> і запустити XSS).

Вхідні дані або не перевіряються, або перевіряються неправильно - без гарантії того, що їх використання не призведе в подальшому до неправильної і небезпечної обробки даних;

12) *неправильна нейтралізація (придушення) спеціальних елементів*: неправильна нейтралізація (придушення) початкових символів або послідовностей, службових символів, пробілів, значення null і т. д.;

13) *використання забороненого коду*: використовуються функції, бібліотеки або сторонні компоненти, які були явно заборонені розробником або замовником;

14) *вбудований шкідливий код*: троянський кінь (Trojan horse), люк (trapdoor), time bomb, logic bomb, шпигунське ПЗ (spyware);

15) *порушення принципів безпечного проектування*: непотрібна складність в механізмі захисту (використовується більш складний механізм, ніж необхідно); опора на єдиний фактор при прийнятті рішення про безпеку; недостатня компартменталізація (compartmentalization) - недостатньо поділяються функціональність або процеси, що вимагають різних рівнів привілеїв, прав чи дозволів; не передбачена перевірка доступу до об’єкту, що захищається, яка виконується кожен раз при зверненні суб'єкта до цього об'єкта; недостатня психологічна прийнятність (складність і незручність використання механізму захисту часто спонукає користувачів не зловмисників відключати або обходити його випадково або навмисно); опора на безпеку через невідомість (використовується механізм захисту, сила якого в значній мірі залежить від його невідомості); недосконаліть механізму підтримки цілісності даних;

16) *некоректне (неправильне) надання зазначеної функціональності*: код не працює відповідно до опублікованих специфікацій, що може привести до неправильного використання;

17) *прихована функціональність*: є функції, що не задокументовані, не є частиною специфікації і недоступні через інтерфейс або послідовність команд (в тому числі, наприклад, функції, що реалізують шкідливий код);

18) *неповна документація*: немає описів всіх відповідних елементів продукту, таких як його використання, структура, інтерфейси, проектування, реалізація, конфігурація, експлуатація і т. д., що, природно, ускладнює обслуговування, побічно впливаючи на безпеку через недостатню поінформованість, ускладнюючи пошук і / або виправлення вразливостей або забираючи багато часу, що також може спростити впровадження вразливостей;

19) *недолік конфігурації*: недотримання вимог безпеки при інсталяції і конфігурації БД (встановлені адміністративні, допоміжні, навчальні облікові записи, що прописуються до БД за замовчуванням без належного їх аналізу і зміни паролів за замовчуванням, не встановлено обмежень на довжину і складність паролів, не заблоковані невикористовувані облікові записи, не встановлені критичні оновлення, неналежним чином налаштована система аудиту подій т. д.).

Надалі для того, щоб можна було б застосувати для оцінки захищеності БД модель системи безпеки з повним перекриттям, кожна з загроз і вразливостей представляється (формалізується) у вигляді елемента деякого кортежу. Для визначеності позначимо кожен тип загроз, як t_i елемент кортежу $T = \langle t_1, t_2, ..., t_i, ... \rangle$, а кожен тип вразливості як γ_k елемент кортежу $\Gamma = \langle \gamma_1, \gamma_2, ..., \gamma_k, ... \rangle$. Загрози і вразливості (як тип, а не як екземпляр для конкретного активу) доцільно розглядати в комплексі. Тільки разом вони можуть стати причиною небажаного інциденту, який може завдати шкоди системі (активам). І в цьому випадку необхідно чітко визначити загрози, вразливості і взаємозв'язок між ними.

Висновки

На підставі проведеного аналізу систем баз даних, як інформаційного продукту з подвійною природою, який має з одного боку компоненту у вигляді програмних засобів СУБД, незалежних від сфери їх застосування, структури, смислового змісту даних, що накопичуються і обробляються, а з іншого – компоненту у вигляді власне збережених даних, існуючих загроз безпеки інформації, таксономій вразливостей були визначені основні значущі для безпеки баз даних загрози, типи вразливостей, як потрібні елементи, необхідні для оцінки захищеності БД з використанням моделі системи безпеки з повним перекриттям. Після ідентифікації загроз і вразливостей, а також оцінки можливості їх об'єднання, визначаються ймовірності небажаного інциденту, для відповідних пар «загроза-вразливість».

ЛІТЕРАТУРА

1. Хоффман Л. Дж. Современные методы защиты информации / Л. Дж. Хоффман . – Москва: Советское радио, 1980. – 264 с.
2. ISO/IEC 15408-1:2009 Information technology. Security techniques. Evaluation criteria for IT security. Part 1: Introduction and general model. URL: https://www.iso.org/standard/50341.html (Last accessed: 22.03.2021).
3. Sandhu R. S., Jajodia S. Data and database security and controls // Handbook of information security management, Auerbach Publishers. – 1993. – P. 481-499.
4. Kulkarni S., Urolagin S. Review of attacks on databases and database security techniques // International Journal of Emerging Technology and Advanced Engineering, Volume 2, Issue 11, – 2012. – P. 2250-2459.
5. Rohilla S., Mittal P. K. Database Security: Threads and Challenges // International Journal of Advanced Research in Computer Science and Software Engineering. – 2013, Vol. 3, Issue 5, P. 810–813.
6. Pfleeger C. P., Pfleeger S. L., Margulies J. Security in Computing. Fifth Edition. – Prentice Hall. – 2015. – 944 p.
7. Imperva Whitepaper. Top ten database security threats. 2015. URL: https://files.meetup.com/5631682/WP_TopTen_Database_Threats.pdf (Last accessed: 22.03.2021)
8. Imperva Whitepaper. Top 5 Database Security Threats. 2016. URL: https://www.imperva.com/docs/gated/WP_Top_5_Database_Security_Threats.pdf (Last accessed: 22.03.2021).
9. MITRE. CVE. Common Vulnerabilities and Exposures. URL: https://cve.mitre.org/data/downloads/allitems.html (Last accessed: 22.03.2021).
10. MITRE. CWE Version 4.2. 2020-08-20. URL: https://cwe.mitre.org/data/published/cwe_v4.2.pdf (Last accessed: 22.03.2021).
11. MITRE. Common Weakness Enumeration. CWE List Version 4.2. URL: https://cwe.mitre.org/data/index.html (Last accessed: 22.03.2021).
12. Защита информации. Уязвимости информационных систем. Классификация уязвимостей информационных систем : ГОСТ Р 56546-2015. – Действующий с 2016-04-01. – Москва : Стандартинформ, 2015. – 13 с.
13. Муханова А., Ревнивых А. В., Федотов А. М. Классификация угроз и уязвимостей информационной безопасности в корпоративных системах // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2013. – Т. 11, № 2. – С. 55-72.
14. Марков А. С., Фадин А. А. Систематика уязвимостей и дефектов безопасности программных ресурсов // Защита информации. Инсайд. – 2013. –№ 3. – с. 2-7.
15. MITRE. CWE VIEW: Research Concepts. URL: https://cwe.mitre.org/data/definitions/1000.html (Last accessed: 22.03.2021).

ВІЛІГУРА Владислав Вікторович – аспірант кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: viligura93@gmail.com; ORCID: 0000-0002-1137-2382.

Наукові інтереси: *бази даних і забезпечення їх безпеки.*

УДК 004.73 **СЕКЦІЯ 2**

ВОЛИНСЬКИЙ В.В, БЕРДНІКОВ А.Г.

МОДЕЛЮВАННЯ SMART-ЛАБОРАТОРІЇ ДЛЯ НАВЧАЛЬНОГО ПРОЦЕСУ

В сучасному світі розвиток технологій набирає колосальну швидкість. Він охоплює абсолютно всі сфери людського життя, тому люди і створюють різні методи і технології для того, щоб покращити, спростити, убезпечити своє життя. Яскравим прикладом такої тенденції розвитку є Internet of Things (IoT). Це поняття з'явилося в 2008-2009 роках, коли число приладів в мережі Інтернет стало більше, ніж населення планети [2]. Тож, можна сміливо сказати, що IoT-це технологія основою якої є машинное спілкування, яка може взаємодіяти між своїми компонентами без будь якого втручання людини, котрі обмінюються знаннями в процесі роботи. З кожним роком популярність IoT зростає, адже цей метод можна використовувати в усіх аспектах життя не тільки людини, а і великих корпораціях, навіть країн. Зростання популярності IoT [8] можна побачити на рис.1.

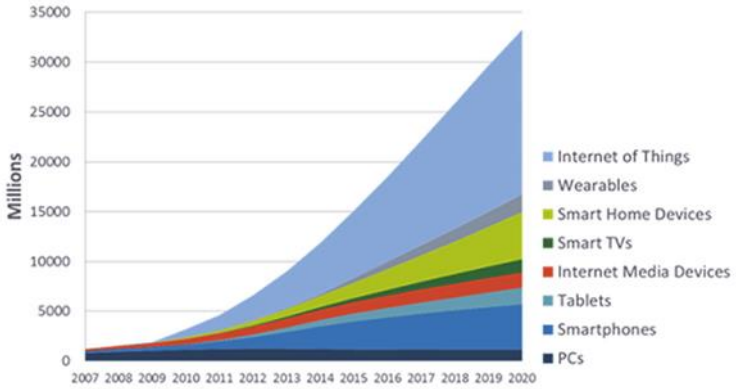


Рис. 1. Зростання популярності IoT в сучасному світі.

В основі IoT лежать такі технології:

- виконавчі пристрої.
- засіб ідентифікації;
- засоби виміру, тобто преобразування зовнішніх даних в формат, який "розуміє" мережа;
- засоби передачі і обробки даних;

Оскільки IoT це частина Всесвітньої мережі, то вона взразлива і тому почали активно створюватися різні методи захисту IoT. Тож в майбутньому виникнуть біль потужні, більш безпечні, більш захищені прилади IoT. Найбільшими гігантами, які займаються розробкою різних "smart девайсів" є Amazon, Google, Auriga. Саме ці корпорації ведуть передові розробки в сфері IoT і в майбутньому планують збільшити фінансування. Коли ми говоримо про IoT - то це не тільки "розумні будинки". Значну частку складають так названі "розумні міста", тобто цілі мегаполіси, які використовують найсучасніші технології для покращення умов життя, для зменшення шкідливих викидів і тому подібне. На рис. 2 зображений поділ IoT на сегменти [6].

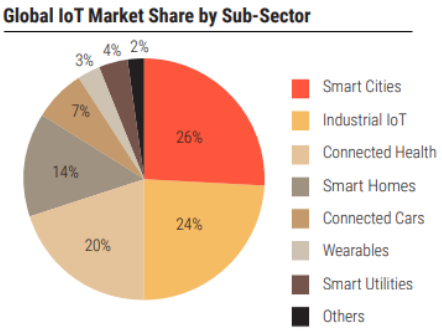


Рис. 2. Поділ частки ринку IoT.

© ВОЛИНСЬКИЙ В.В, БЕРДНІКОВ А.Г., 2021

Кожен з нас стикався з поняттям “розумний дім”. Але не всі до кінця розуміють, який сенс закладено в цей термін. Прийнято вважати, що це система домашньої автоматизації, комплекс приладів, принципів та протоколі, які допомагають приймати рішення і виконувати повсякденні задачі по дому. За допомогою спеціальних датчиків, пристроїв, приладів і різних допоміжних елементів функціонує “розумний” будинок. Ця тема цікава, тому що вона допомагає правильно організувати безпеку, домашній затишок, простоту керування, а також відповідність всім нормам сучасного телекомунікаційного світу будинку, який проектується. Всі пристрої “розумного будинку” пов’язані через Інтернет. Завдяки цьому користувач може керувати своєю системою за допомогою будь-якого зручного пристрою, будь то телефон, планшет чи ноутбук і робити це безпечно з будь-якого місця. Варто зазначити, що багато пристрої “розумного дому ” працюють в бездротовій мережі і для такої роботи їм потрібна енергія, а деякі прилади споживають багато її досить немало, тому цю особливість приладів також слід враховувати. З іншого боку бездротовий зв’язок зменшує витрати на встановлення і налаштування системи. Але кожен тип підключення підходить для різних ситуацій, тому не обов'язково використовувати щось одне. На рис. 3 зображено зростаючу популярність “розумного будинку” [7].

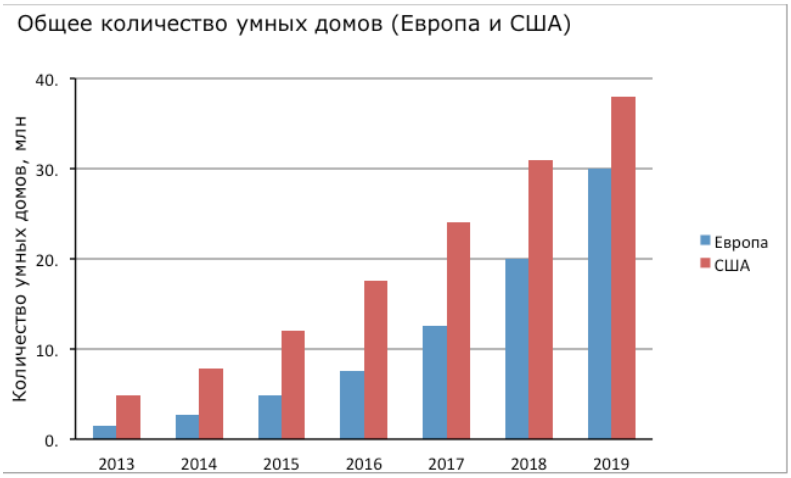


Рис. 3 Популярність “розумного дому”

У роботі [5] дуже обширно, зрозуміло і досить просто описано як поняття IoT, так і технологію “розумного дому”.

Постановка задачі дослідження

Провівши дослідження Internet of Things і такого поняття, як “розумний дім” можна сказати, що всі компоненти легко використати для того, щоб змоделювати та дослідити схему “Smart лабораторії” для покращення навчального процесу. Можна виділити такі переваги: контроль; моніторинг; вартість; комфорт і безпека; гибкість і можливість до замообучення; Ми маємо всі компоненти для того, щоб створити систему, яка зможе задовольнити потреби, які встановлюються для навчального процесу. Тому потрібно дослідити такі аспекти:
1)проаналізувати технологію IoT; 2) проаналізувати і вивчити принципи функціонування мережі “розумного дому”; 3) розглянути, як можна застосовувати “розумні пристрої” в процесі навчання. 4) створити власну мережу “розумної лабораторії” на основі вибраного типу зв’язку; 5) налаштувати управління мережею за допомогою мережевих пристроїв.

Використання додатку Cisco Packet Tracer для створення моделі “розумної лабораторії”

Якщо ми говоримо про створення деякої моделі “smart” - проекту, то краще всього використовувати додаток Cisco Packet Tracer [3]. Це симулятор мережі передачі даних, створений Cisco Systems. Він дозволяє експерименту над мережею, оцінювати сценарії розвитку подій. В цьому емуляторі можна дослідити роботу маршрутизаторів, комутаторів, точок бездротового доступу, а також інструментарій IoT. Важливим плюсом системи є те, що можна власноруч ствоювати і програмувати різні приладі з IoT. Але з'являється питання: чи можна інтегрувати дану технологію не тільки для покращення життя, але і для покращення

процесу навчання? Для того, щоб створити таку систему потрібно вибрати технології бездротових зв'язків, в комбінації з дротовим підключенням. Такий етап не варто ігнорувати, адже від нього залежить, який тип приладів слід встановлювати. По-перше, потрібно визначити, заради яких цілей створюється система. По-друге, встановити які елементі і вузли потрібно автоматизувати. По-третє, визначити, як саме автоматизувати, за допомогою яких протоколів, правил і методів. Для того, щоб розробити таку систему, потрібно використати обладнання, зображене на рис. 4.

Рис. 4 Прилади "розумного дому".

Важливим плюсом Cisco Packet Tracer - є те, що можна розділяти розроблену мережу на логічний та фізичний рівні [4]. Якщо ми створюємо мережу в логічному простору, то мається на увазі, що всі компоненти знаходяться в єдиному місці. Це непогано, коли мережа невелика і проста в функціонування, але для "розумної лабораторії" цього замало. З іншого боку, фізичний простір допомагає розділити мережу на "регіони" і налаштовувати для кожного компоненту змінні середовища, за допомогою яких і буде правильно функціонувати система "розумної лабораторії". Безумовно, що пристрої з IoT можуть покращити процес навчання. Тому що, багато різних аспектів буде обробляться і система автоматично буде реагувати на них. Прикладом є датчики напруги, датчики вуглекислого газу, датчик диму, детектор температури і тому подібне [1]. В залежності від кількості студентів, їх потреб, факторів зовнішнього середовища система "розумної лабораторії" буде реагувати і виконувати ті дії, які зможуть покращити процес навчання так, щоб він проводився більш ефективно і менш утомливо, на відміну від "розумного дому" , який зазвичай реалізується для того, щоб покращити саме умови життя в помешканні, задовольняти повсякденні потребності жильців в плані комфорту, затишку або виконання рутинних повсякденних обов'язків.

ЛІТЕРАТУРА

1. Google Your Home: Setting Up a Network of Nest Devices In Your Home. December 12, 2019. P. 45-46.
2. Klaus Schwab. The Fourth Industrial Revolution. 2017. P. 23-25.
3. Cisco Networking Academy. Introduction to IoT/ Cisco. [Електронний ресурс]. Режим доступу до ресурсу: https://www.netacad.com/portal/learning. – 2020.
4. Andrea Finardi. IoT simulations with Cisco Packet Tracer. [Електронний ресурс]. Режим доступу до ресурсу https://www.theseus.fi/bitstream/handle/10024/150158/Andrea%20Finardi%20%20Master%20of%20Engineering%20%20Information%20technology.pdf?sequence=1&isAllowed=y.– 2017.
5. Cathy Young.Smart Home: Digital Assistants, Home Automation, and the Internet of Things (Our Internet of Things) – 2018. P. 182-415.
6. WHIN - IOT [Електронний ресурс]. – 2020. Режим доступу до ресурсу: https://whin.org/iot/.
7. Умный дом: Развитие и тенденции. [Електронний ресурс]. Режим доступу до ресурсу: https://habr.com/ru/company/gsgroup/blog/387651/.
8. Jānis Kampars. Internet of Things: Structure, Features and Management.– 2016. P. 81-82.

ВОЛИНСЬКИЙ Валентин Володимирович – кафедра прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: xa11867787@student.karazin.ua; ORCID 0000-0002-2692-2279

Наукові інтереси: *технологія "smart" будинку;*

БЕРДНІКОВ Анатолій Георгійович – к. т. н., доцент; доцент кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені. В.Н. Каразіна; м. Харків, майдан. Свободи, 6, 61022; e-mail:a.berdnikov@karazin.ua , ORCID 0000-0001-7923-0754

Наукові інтереси: *застосування методів теорії ухвалення рішень при управлінні складними телекомунікаційними системами, управління проектами.*

УДК 004.75, 004.8 **СЕКЦІЯ 3,7**

ГАВРИЛЮК Є.А., СПОРОВ О.Є., ГУЩИН І.В.

ПОРІВНЯННЯ ЗАСТОСОВНОСТІ АЛГОРИТМІВ НАВЧАННЯ З ПІДКРІПЛЕННЯМ У МУЛЬТИАГЕНТНИХ РОЗПОДІЛЕНИХ СИСТЕМАХ

Загальні відомості про алгоритми навчання з підкріпленням

Алгоритми навчання з підкріпленням на сьогодні є одним з найперспективніших напрямків у машинному навчанні взагалі. Використовуючи їх, можна моделювати проблеми тварин, людей та роботів, пов'язані з винагородами. Так, у алгоритмах навчання з підкріпленням виділяють такі поняття, як «агент» та «середовище», де агент - це система, що навчається у даному середовищі.

Так само, як багатопоточність може дати приріст у продуктивності системи, багатоагентність, або ж, мультиагентність, може дати приріст у продуктивності системи навчання з підкріпленням. У даному дослідженні розглядатимуться такі типи мультиагентних алгоритмів: повністю кооперативні, повністю конкурентні та комбіновані. Буде запропоновано можливі критерії оцінки та проведено порівняльний аналіз їх застосовності за методом аналізу ієрархій.

На сьогодні можна виділити такі два основних методи навчання з підкріпленням: методи стратегій та методи значень. Далі у статті використовуються приклади саме з методами стратегій, адже на сьогодні вони набувають більшої популярності [1].

Розглянемо принцип дії алгоритмів навчання з підкріпленням на прикладі методів стратегій. Методи градієнтної стратегії працюють шляхом обчислення оцінки градієнта стратегії та підключення його в стохастичний алгоритм сходження градієнта. Найбільш часто використовуваний оцінювач градієнта має форму (1):

$$\hat{g} = \hat{E}_t[\nabla_\theta log\pi_\theta(a_t|s_t)\hat{A}_t], \quad (1)$$

де π_θ – стохастична стратегія;

$\hat{A}_t$ – оцінювач функції переваг на t кроці за часом;

$\hat{E}_t$ – очікування, що вказує на емпіричне середнє значення за кінцеву вибірку.

Сучасними методами, наприклад, можна вважати такі, як TRPO (Trust Region Policy Optimization) та більш молодий метод Proximal Policy Optimization (PPO).

У методах TRPO цільова функція максимізується за умови дотримання обмеження на розмір оновлення стратегії (2-3):

$$\underset{\theta}{maximize}\hat{E}_t[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}\hat{A}_t], \quad (2)$$

$$\text{subject to } \hat{E}_t\left[KL\left[\pi_{\theta_{old}}(\cdot|s_t), \pi_\theta(\cdot|s_t)\right]\right] \le \delta, \quad (3)$$

де θ_{old} – вектор параметрів стратегії перед оновленням;

KL – відстань Кульбака-Лейблера (відносна ентропія).

На перевагу TRPO, алгоритми типу PPO, наприклад, що були представлені у 2017 році командою OpenAI як альтернатива методам класу TRPO, замість того, щоб додавати функції для обмеження, вбудовує ці властивості у саму цільову функцію (4). Він отримує ті ж переваги у продуктивності та уникає складностей, оптимізуючи сурогатну ціль [2]:

$$L^{CLIP} = \hat{E}_t\left[\min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right], \quad (4)$$

де ϵ – деякий гіперпараметр, скажімо, 0.2.

Детальніше про ці алгоритми викладено у роботі [2].

© ГАВРИЛЮК Є.А., СПОРОВ О.Є., ГУЩИН І.В., 2021

Алгоритми навчання з підкріпленням у мультиагентних системах

Мультиагентні системи навчання з підкріпленням вирішують ту ж саму проблему, що й одноагентні – послідовного прийняття рішень, але за участі більш ніж одного агенту. Таким чином, еволюція системи та винагорода кожного агенту залежать від сумісних дій усіх агентів, що має підвищити загальну продуктивність системи.

У роботі [1] виділяють три види мультиагентних алгоритмів навчання з підкріпленням в залежності від типу налаштувань, до яких вони звертаються: повністю кооперативні, повністю конкурентні та комбінацію з двох. В умовах кооперації агенти співпрацюють для оптимізації загальної довгострокової віддачі; тоді як у конкурентних умовах віддача агентів зазвичай дорівнює нулю. Змішаний тип залучає як кооперативних, так і конкурентних агентів із загальною прибутковістю. Розглянемо кожний тип окремо.

Повністю кооперативні алгоритми навчання

Як зазначено у роботі [1], найважливішим є вивчення оптимальної спільної стратегії, так, щоб кожний агент мав доступ лише до локальної та сусідньої інформації у мережі. Одним з запропонованих варіантів є алгоритм «Суб'єкт-Критик» (від англ. «Actor-Critic»), алгоритм навчання з підкріпленням, що об'єднує у собі підходи оптимізації значень та оптимізації стратегії. Так, у рамках цього алгоритму існує дві моделі – Суб'єкт та Критик.

Суб'єкт виконує роль навчання. Середовищу передаються ймовірності кожної дії, що спрогнозував Суб'єкт, та спостерігається, що змінилося. Якщо щось позитивне трапляється як результат дій агенту, тоді середовище посилає позитивний відгук у формі нагороди, та навпаки. Ці нагороди беруться шляхом тренування Критика.

Головною роллю моделі Критика є навчання оцінюванню, чи дія конкретного Суб'єкта призвела до кращого стану або до гіршого, і давати свій відгук Суб'єкту. Критик видає число з плавучою точкою, що відображає рейтинг (Q-value) дії у попередньому стані. Шляхом зрівняння цього рейтингу, Суб'єкт може зрівняти свою поточну стратегію з новою стратегією та вирішити, як йому діяти, щоб покращити результат. Детальніше алгоритм «Суб'єкт-Критик» розглянуто у роботі [1].

Повністю конкурентні алгоритми навчання

Такі алгоритми зазвичай моделюються як ігри з нульовою сумою. Для методів стратегій, наприклад, намагаються максимізувати результат агенту, що очікується, знехтуючи стратегіями інших агентів. Можна виділити два великих сімейства методів: методи фіктивної гри та методи мінімізації контрфіктивного жалю.

Щодо методів фіктивної гри, все доволі просто. Гравці (агенти) повторювано грають, а кожен гравець «підганяє» свою стратегію під загальносередню стратегію інших агентів. Говорячи про методи мінімізації контрфіктивного жалю, це був справжній прорив у спробах вирішення великомасштабних ігор з розширюваною формою. Так, цей метод широко використовується для ігор з неповною інформацію. Обидва методи детальніше розглянуті у роботі [1].

Комбіновані алгоритми навчання

Як слідує з назви, методи цієї групи об'єднують у собі обидва підходи, а саме, скажімо, модель «Суб'єкт-Критик» та метод фіктивної гри. Сьогодні, наприклад, використовується підхід «self-play»: агенти поділяються на дві конкуруючих команди, що грають одна проти одної. Нагорода гравця у такому випадку розділяється кожним гравцем у команді.

З одного боку, може здатися, що використання комбінованих алгоритмів має дати більший поштовх з точки зору ефективності мультиагентних систем, але через підвищену складність їх імплементації та неповністю досліджену продуктивність у зрівнянні з повністю конкурентними та повністю кооперативними алгоритмами, вони на сьогодні залишаються маловивченими, а існуючі реалізації малочисельними [1].

Порівняння застосовності алгоритмів навчання у мультиагентних системах

Якщо взяти у приклад використання вже зазначеної вище моделі «Суб'єкт-Критик», то тут можуть виникати заперечення. Так, наприклад, існують два основних варіанти цієї моделі,

що широко використовуються сьогодні – A3C (Asynchronous Advantage Actor Critic) та A2C (Advantage Actor Critic). Ці два підходи відрізняються тим, що модель A3C імплементує паралельне навчання, коли декілька гравців у паралельних середовищах незалежно корегують глобальну функцію значень (асинхронно), коли модель A2C – це та ж модель A3C, але без асинхронної частини, тобто гравець один [3].

Заперечення складається у тому, що, на перший погляд, модель A3C здається більш продуктивною у порівнянні з A2C, але, виходячи з інформації, що надали науковці з команди OpenAI у статті [4], їх синхронна імплементація A2C показує себе краще, ніж асинхронна. Таким чином, це питання залишається відкритим для багатьох науковців сьогодні.

Щодо повністю конкурентних алгоритмів, метод фіктивної гри є досить поширеним у використанні та не має такої невизначеності, як, наприклад, використання різних імплементацій методу «Суб'єкт-Критик», тож їх використання у алгоритмах навчання з підкріпленням не викликає труднощів.

Говорячи про комбіновані методи, можна зазначити, що їх перспективи у порівнянні із повністю кооперативними та повністю конкурентними методами є кращими, адже, взявши з обох типів методів найкраще у поєднанні, можна, теоретично, отримати більш продуктивну систему. Тим не менше, ці методи потребують більш детального вивчення та практичного підтвердження їх теоретичної значущості [1].

На основі викладеної інформації, можна, наприклад, виділити такі критерії для порівняння різних типів алгоритмів навчання з підкріпленням у мультиагентних системах:

1) Складність імплементації (для скорочення далі – Складність);
2) Вивченість;
3) Продуктивність.

Скористаємось методом аналізу ієрархій для оцінки методів за цими критеріями [5]. Для цього складемо матрицю попарних зрівнянь критеріїв, при цьому, використовується наступна шкала оцінок:

1) рівно, без різниці = 1;
2) трохи краще = 3 (1/3=0,333);
3) краще = 5 (1/5=0,2);
4) багато краще = 7 (1/7=0,143);
5) принципово краще = 9 (1/9=0,111).

При проміжній думці використовуються проміжні бали (2,4,6,8).

Так, критерій продуктивності можна визначити головним, що можна побачити з Таблиці 1, адже саме від нього залежить час виконання алгоритму навчання. Знаходимо суми за стовбцями та ділимо кожне значення на цю суму (нормуємо матрицю), після чого знаходимо середнє кожного рядка за нормованими значеннями (значення ваги кожного критерія).

Табл.1 Попарне зрівняння критеріїв

	Складність	Вивченість	Продуктивність	Середнє
Складність	**1**/11=0,09	**0.333**/7,333=0,045	**0,143**/1,31=0,109	**0,081**
Вивченість	**3**/11=0,273	**1**/7,333=0,136	**0,167**/1,31=0,127	**0,178**
Продуктивність	**7**/11=0,636	**6**/7,333=0,818	**1**/1,31=0,763	**0,739**
СУМА	11	7,333	1,31	

Далі складаються таблиці (Таблиці 2-4) зрівняння альтернатив за критеріями (операції над матрицями ті ж самі, для більш наглядної інтерпретації операцію нормування у таблиці не вказано, але середнє рахується за нормованими значеннями так само, як зазначено у Таблиці 1).

Табл.2 Попарне зрівняння за критерієм «Складність»

Складність	Повністю кооп.	Повністю конк.	Комбіновані	Середнє
Повністю кооп.	1	0,333	6	**0,429**
Повністю конк.	3	1	7	**0,5**
Комбіновані	0,167	0,143	1	**0,071**

Табл.3 Попарне зрівняння за критерієм «Вивченість»

Вивченість	Повністю кооп.	Повністю конк.	Комбіновані	Середнє
Повністю кооп.	1	0,5	5	**0,343**
Повністю конк.	2	1	6	**0,575**
Комбіновані	0,2	0,167	1	**0,082**

Табл.4 Попарне зрівняння за критерієм «Продуктивність»

Продуктивність	Повністю кооп.	Повністю конк.	Комбіновані	Середнє
Повністю кооп.	1	1	0,2	**0,143**
Повністю конк.	1	1	0,2	**0,143**
Комбіновані	5	5	1	**0,714**

Отримаємо наступну матрицю ваги альтернатив за кожним критерієм (Таблиця 5).

Табл.5 Ваги альтернатив за кожним критерієм

	Складність	Вивченість	Продуктивність
Повністю кооп.	0,429	0,343	0,143
Повністю конк.	0,5	0,575	0,143
Комбіновані	0,071	0,082	0,714

Перемноживши цю матрицю на вектор-стовпець з вагою критерієв з Таблиці 1, отримаємо вагу альтернатив з точки зору досягнення цілі (Таблиця 6):

Табл.6 Загальні ваги альтернатив

Альтернатива	Вага	Вага у відсотках
Повністю кооп.	0,201	20,1%
Повністю конк.	0,249	24,9%
Комбіновані	0,548	54,8%

Таким чином, можна побачити, що найбільшу вагу мають саме комбіновані алгоритми (54,8%).

Висновки

У результаті проведеного дослідження було виконано аналіз основних методів навчання з підкріпленням: метод політик та метод аналізу ієрархій, та виконано порівняння застосовності алгоритмів навчання з підкріпленням у мультиагентних розподілених системах. Крім того, було запропоновано підхід до порівняння трьох типів методів навчання за методом аналізу ієрархій. У результаті отриманих оцінок визначено, що найкращим вибором, незважаючи на малу вивченість та складність імплементації, будуть комбіновані методи. При цьому слід пам’ятати, що на даний час критерій продуктивності комбінованих методів є об’єктом інтенсивних досліджень та має великі перспективи при використанні [1].

ЛІТЕРАТУРА

1. K. Zhang, Z. Yang, T. Başar Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms. *ArXiv, abs/1911.10635*, 2019.
URL: https://arxiv.org/pdf/1911.10635.pdf (Last accessed: 09.04.2021).

2. J. Schulman, F. Wolki, P. Dhariwal, A. Radford, O. Klimov Proximal Policy Optimization Algorithms. *ArXiv, abs/1707.06347*, 2017.
URL: https://arxiv.org/pdf/1707.06347.pdf (Last accessed: 09.04.2021).
3. V. Mnih, A. P. Badia, M. Mirza, Asynchronous Methods for Deep Reinforcement Learning. *ArXiv, abs/1602.01783*, 2016.
URL: https://arxiv.org/pdf/1602.01783 (Last accessed: 09.04.2021).
4. Y. Wu, E. Mansimov, Sh. Liao OpenAI Baselines: ACKTR & A2C, 2017. URL: https://openai.com/blog/baselines-acktr-a2c/ (Last accessed: 09.04.2021).
5. П. А. Гудков Методы сравнительного анализа. Учеб. пособие. – Пенза: Изд-во Пенз. гос. ун-та, 2008 – 81с.; 5 ил., 15 табл., библиогр. 12 назв. URL: http://window.edu.ru/resource/707/59707/files/stup418.pdf (Last accessed: 09.04.2021).

ГАВРИЛЮК Єгор Андрійович – бакалавр; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: havryliuk2020ks51@student.karazin.ua; gavrilyk2903@gmail.com; ORCID: 0000-0002-4392-2000.
Наукові інтереси:
- *алгоритми навчання з підкріпленням, штучний інтелект, розробка програмного забезпечення в області аналізу даних.*

СПОРОВ Олександр Євгенович – кандидат фізико-математичних наук, доцент кафедри штучного інтелекту та програмного забезпечення; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, ауд. 5-20, Тел. 707-50-76, м. Харків, Україна, 61022; e-mail: sporov@karazin.ua; ORCID: 0000-0002-4610-9656.
Наукові інтереси:
– *штучний інтелект, розподілені обчислення, операційні системи, комп'ютерне моделювання фізичних процесів.*

ГУЩІН Іван Валерійович – старший викладач кафедри штучного інтелекту і програмного забезпечення; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: gushchin.iv@gmail.com; ORCID: 0000-0002-1917-716X.
Наукові інтереси:
– *розробка програмного забезпечення в області аналізу даних, комп'ютерне моделювання фізичних процесів.*

УДК 004.056.5: 342.922/951 **СЕКЦІЯ 5**

ГАЙКОВА В.В., МАЛАХОВ С.В.

УЗАГАЛЬНЕННЯ УМОВ РЕАЛІЗАЦІЇ АКЦІЙ КІБЕРБУЛІНГУ З УРАХУВАННЯМ ОСОБЛИВОСТЕЙ СУЧАСНИХ ІНФОРМАЦІЙНИХ СИСТЕМ

Введення

Розвиток інформаційних технологій (ІТ) привносить в сучасне суспільство багато позитивних та корисних тенденцій, однак, нажаль, він має і свої «темні» сторони, що приховані від непосвячених в реалії існування сучасного цифрового світу. Однією з таких сторін, є явище булінгу, яке в наслідок стрімкого науково-технічного розвитку останніх десятиліть, непомітно «перейшло» в кіберпростір, та відкрило нову, непривабливу сторінку технологічної історії людства – сторінку кібербулінгу [1-3]. Процес протистояння булінгу в кіберпросторі [4, 5], обумовлює необхідність постійної розробки нових та вдосконалення вже існуючих технологій і засобів протидії [6], включаючи, і навчальні технології (наприклад, всілякі ігри, інтерактивні уроки, відео та ін.), які спрямовані на завчасне коригування "мережевої поведінки" користувачів різних інформаційних систем, та онлайн сервісів.

Основні узагальнення та їх аргументація

Результати аналізу інцидентів, що пов'язані з різними формами проявів булінгу в кіберпросторі, дозволяє зробити кілька принципових узагальнень:

1 – всі випадки цькування в кіберпросторі мають два рівня впливу (в сенсі обраної кінцевої цілі): - груповий або глобальний та індивідуальний (персональний);

2 – об'єкти кібертравлі значною мірою збігаються з основними об'єктами впливу в ході ведення інформаційних операцій [7];

3 – практично всі випадки кібербулінгу, мають структуру і основні складові, які притаманні для сучасних[1] умов ведення інформаційних операцій [7, 8];

4 – доступність використовуваних онлайн сервісів і послуг, що надаються різними інформаційними системами, є визначальним (критично важливим) фактором для досягнення кінцевих цілей акцій кібербулінгу;

5 – характер поведінки і ролі [9] основних учасників процесу кібертравлі, багато в чому ідентичні умовам і цілям проведення експерименту Стенлі Мілгрема (відомого, як «Експеримент про покору») [10-14]. У випадках кібербулінгу, ми є свідками процесу масової реінкарнації цього експерименту в умовах існування інформаційного суспільства [15, 16], з відповідною «еволюцією»[2] уявлень учасників процесу, про соціальну поведінку людини[3] у кіберпросторі.

Примітки:

[1] – мається на увазі, використання можливостей існуючих інформаційних систем (платформ) і ІТ - технологій;

[2] – може і не бути такою, а бути, всього лише, похідною поточних комерціалізованих тенденцій, сформульованих домінуючими постачальниками телекомунікаційних послуг і сервісів (наприклад, гігантами ІТ-індустрії Google, Twitter, Facebook, YouTube та ін.);

[3] – однієї зі сторін процесу може бути комп'ютеризована система (бот), що реалізує, як наперед задані поведінкові алгоритми, так і підтримує функції самонавчання (штучного інтелекту - ШІ). У другому випадку складно говорити про «соціалізацію» поведінкових алгоритмів боту, зважаючи на значну складність формалізації його поведінкових реакцій.

Розглянемо представлені узагальнення і зробимо деякі пояснення по кожному з них.

1. В контексті зазначених рівнів впливу, кінцевою ціллю (жертвою) проведених акцій кібертравлі можуть бути:

- конкретні персони (у т.ч. користувачі будь-яких онлайн сервісів і послуг);
- об'єднання або групи людей (учасники різних мережевих суспільств з різними критеріями групоутворення (соціальні, культурні, майнові та ін.));

© ГАЙКОВА В.В., МАЛАХОВ С.В., 2021

- корпоративний сегмент (представники окремих бізнес структур, рівнів державної влади, культових організацій та ін.).

2. Рівень ієрархічної важливості (значущості) об'єктів обраних для кібертравлі, обумовлює ступінь необхідної інтеграції (об'єднання) функцій управління контентом і основних складових ресурсного забезпечення інформаційних систем (Рис. 1). Ця взаємозалежність у повній мірі справедлива і для умов ведення інформаційних операцій [7];

3. В якості підтвердження запропонованого узагальнення, стосовно тотожності структури кібербулінгу (особливо для випадків глобального рівня впливу) і основних складових інформаційних операцій, варто розглянути схему на Рис. 1. Аналіз цієї схеми дозволяє констатувати, що основні складові процесу кибербуллингу, співпадають з відповідними елементами, які притаманні для умов ведення інформаційних операцій (виділені подвійним контуром). Можливі відмінності, в частині представництва та ролей учасників процесу (операції) в даному разі не принципові, так як у будь-якому випадку, присутні і виконувачі акції, і цільові жертви атаки (група «**КІНЦЕВІ СПОЖИВАЧІ**»).

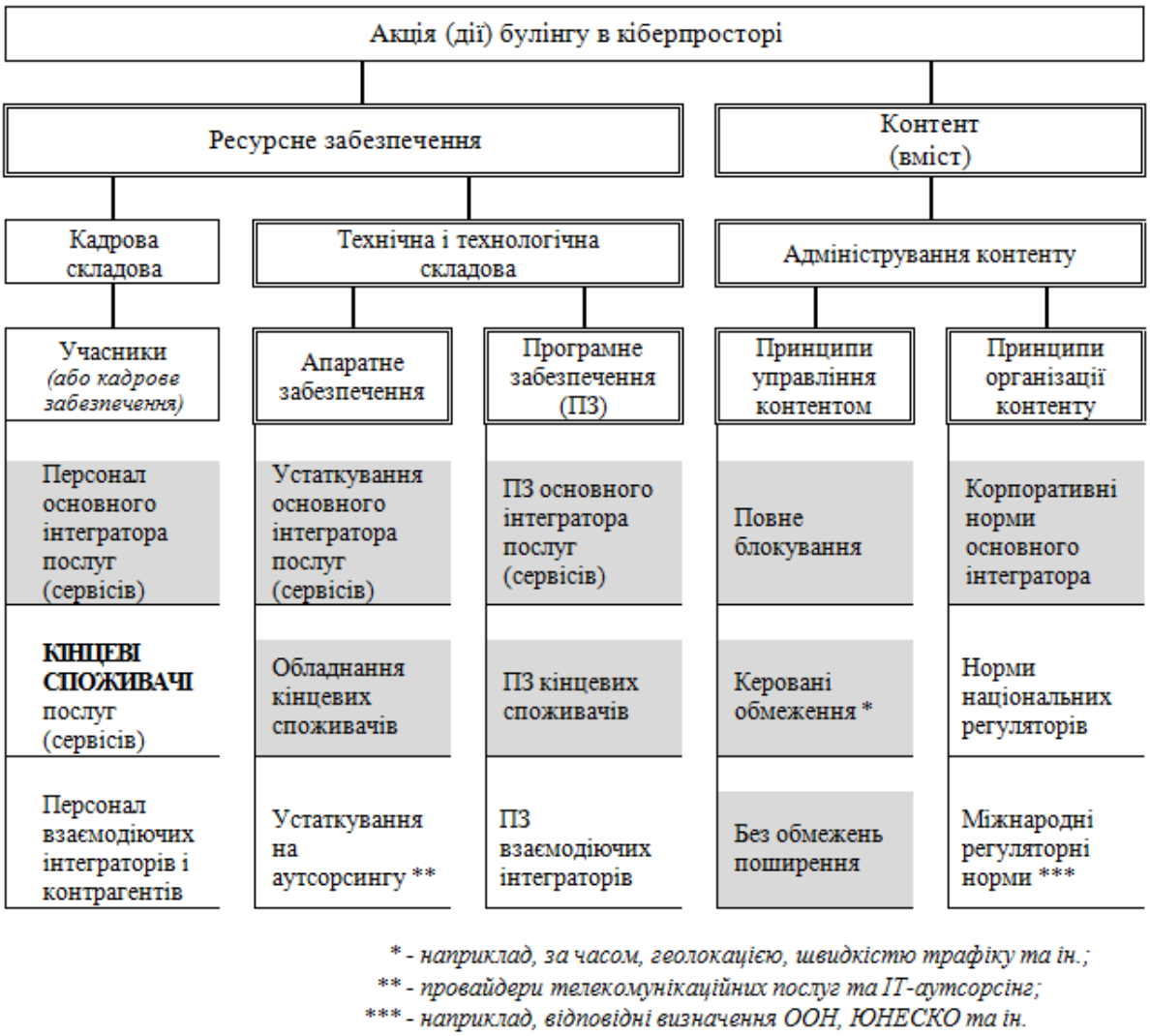


Рис. 1 Структура складових кібербулінгу, яка притаманна умовам ведення інформаційних операцій (для високих рівнів ієрархічної важливості об'єктів кібертравлі)

Принциповим є те, що саме категорія «**КІНЦЕВІ СПОЖИВАЧІ**», в залежності від конкретної ситуації, може одночасно поєднувати в собі декілька ролей кібертравлі [9], перш за все: - «жертв», «хамелеонів», «захисників» і «спостерігачів». Причому, в разі здійснення булінгу на корпоративному рівні впливу або проти об'єктів з високим рівнем ієрархічної значущості, категорія «Персонал взаємодіючих інтеграторів…» (Рис. 1) може поєднувати в собі

представників ролевих груп кібербулінгу «помічники агресорів» і «хамелеони» (розглянуто нижче в п. №4 і №5), які будуть діяти згідно загального плану акції мережевого цькування.

4. Фактор доступності (неперервності) використовуваних онлайн сервісів і послуг визначає не тільки спосіб, масштаб і тривалість реалізації процесу кібертравлі, а й обумовлює саму можливість його здійснення. Характерним прикладом успішно реалізованих заходів кібертравлі, що експлуатують фактор доступності комунікаційних сервісів, є акція по одночасної «ізоляції» (виключення і ігнорування) колишнього президента США Д. Трампа, на основних національних інформаційних (TV компанії) і глобальних комунікаційних ресурсах (соціальні мережі і месенджери). Такий спосіб атаки забезпечує підтримку головної умови успішності проведеної акції – нерівність сил агресора[4] і жертви.

Примітка:

[4] *– уточнення рольової категорії «агресор», у разі монополізації основних складових процесу кібербулінгу, запропоновано нижче, при розгляді питань аналогій з умовами експерименту С. Мілгрема [12];*

Показовим прикладом критичності фактора доступності, є можливість потенційного об'єднання, в рамках однієї компанії, холдингу або держави, основних складових «Ресурсного забезпечення» і функцій «Адміністрування контенту» (виділено сірим тоном на Рис. 1). Подібний варіант інтеграції складових можливий, наприклад, у разі повного розгортання низькоорбітальної глобальної супутникової системи Starlink (від компанії SpaceX).

Монополізація основних програмно-апаратних складових та функцій адміністрування контенту (Рис. 1), стають серйозною проблемою навіть для рівня окремих держав, не кажучи вже про персональний рівень впливу. В такому випадку потенційний «агресор», отримує у «свої руки» всі ключові функції процесу: - планування; - супровід; - контроль результату. Крім того, домінування корпоративних норм основного інтегратора послуг в питаннях адміністрування контенту, фактично зумовлює кінцевий результат. Таким чином, можлива протидія обмежується, лише, заходами ліцензування і контролю поставок абонентського обладнання, з усіма витікаючими наслідками...

5. Відповідно до умов експерименту Мілгрема, до його проведення залучались: – «експериментатор» (персона з повноваженнями коригувати хід експерименту); – «вчитель» (випробуваний, який лише частково уявляв сенс експерименту); – «учень» (актор, що грав свою роль і мав повне уявлення про сенс експерименту). Які ж аналогії можна виділити, порівнюючи учасників «експерименту про покору» та ролі основних фігурантів акцій цькування в кіберпросторі? Тут доречно зауважити, що рольова структура учасників звичайного булінгу набагато простіше, ніж рольова структура учасників кібербулінгу.

Так, у ситуації «традиційного» булінгу присутні три основні ролі: - жертва, переслідувач і свідок. При цьому рольова структура кібербулінгу передбачає присутність наступних акторів [9]: 1 - «агресори» (переслідувачі); 2 - «помічники агресорів», забезпечують підтримку процесу цькування; 3 - «жертви», це учасники, які піддаються публічному цькуванню; 4 - «захисники і хамелеони», представники цих груп не стабільні і можуть мігрувати між групами «спостерігачів» і «агресорів»; 5 - «спостерігачі» це найчисленніша і ситуативно не ангажована група учасників (принаймні, на початковому етапі подій).

Враховуючи фактор анонімності присутності представників 5-ї групи в ході здійснення мережевого цькування, можна стверджувати, що на відміну від ситуації «традиційного» булінгу, в даному разі відсутній прямий контакт між «агресором» і «спостерігачами». Віддалена присутність учасників цієї групи і можливість швидкої зміни їх чисельності визначає основну рольову функцію даної групи, заради якої, власне, організовується та підтримується все дійство. З огляду на швидкість, масштабованість і повторюваність процесу цькування, ця група учасників, де-факто, є кінцевою ціллю проведення всієї акції, за результатами якої повинні бути скориговані їх соціальний (мережевий) статус або соціальна (мережева) поведінка, або ж і те й інше одночасно. Тобто, в даному випадку, можна говорити про приховану насильницьку корекцію самоідентифікації учасників даної групи, шляхом послідовної і навмисної зміни їх уявлень, про норми соціальної (мережевої) поведінки в кіберпросторі. Вочевидь, що саме таке трактування прихованих цілепокладань процесу кібербулінгу, є тією сполучною, яка забезпечує його концептуальну аналогію з основною парадигмою експерименту С. Мілгрема.

У заявленому контексті неявній ролі «спостерігачів», роль «жертви» трансформується в інструмент для досягнення іншої, прихованої глобальної мети – забезпечення абсолютного домінування в найбільш представницької мережевій рольової групі! При такій постановці питання, роль «жертви», як це не дивно виглядає, це всього лише «ширма», основним призначенням якої, є нав'язування мережевого дискурсу із заздалегідь визначеним наративом і прихованим кінцевим цілепокладанням. Іншими словами, в даному випадку, можна говорити про одну з можливих реалізацій стратегії непрямих дій, що адаптована до умов існування інформаційного суспільства і можливостям ведення інформаційних операцій в кіберпросторі.

А тепер, власне, про аналогії з «експериментом про покору» [10-13]. Вочевидь, що найпростіше виділити учасників групи «вчитель» (головний випробовуваний цього експерименту). Відносно процесу булінгу в кіберсфері такі функції виконує «агресор». При цьому визначальною якістю представників цієї групи (мережевого співтовариства) повинна бути готовність «йти до кінця», в рамках реалізації всього комплексу заходів цькування, об'єднаних загальною стратегією дій. Крім того, група «вчитель» може поглинати і рольову категорію «помічники агресорів», яка в більшості випадків знайома з порядком та умовами проведення акції. Однак, у випадках проведення заходів з «глобальним рівнем впливу», представники цієї рольової групи, швидше за все, не будуть ознайомлені з кінцевою метою проведеної акції (залучаються для завдань підготовки контенту або підтримки процесу цькування). Іншими словами завдання акторів групи «вчитель» можуть бути фрагментовані, а їх використання обмежуватися рамками передбачених етапів / стадій мережевої атаки.

Виникає логічне запитання: - хто ж визначає «жертву» та формує стратегію всієї акції (тобто, правила адміністрування контенту, стадії, час і масштаб проведення заходів)? Очевидно, що в пошуках відповіді ми «виходимо» на повноваження «експериментатора» (у термінології експерименту про покору). Враховуючи тенденцію останніх років, до зрощення джерел контенту і основних складових ресурсного забезпечення інформаційних систем, очевидно, що учасниками рольової групи «експериментатор», при високих рівнях ієрархічної важливості об'єктів кібертравлі, можуть бути тільки монополісти результатів подібного об'єднання. При цьому, в заходах мережевого цькування на нижчих рангах ієрархії (тобто в «традиційному» розумінні випадків кібербулінгу), в ролі «експериментатора», нажаль, можуть виступати представники тимчасових мережевих суспільств, які визначають локальну інформаційну повістку (онлайн голосування, рейтингові оцінювання, опитування та ін.) та формують потрібну для них форму уявлень, про норми мережевої поведінки в «їх» кіберпросторі.

Висновки

Характер поведінки і ролі основних учасників процесу кібертравлі, багато в чому ідентичні умовам і цілям проведення експерименту Стенлі Мілгрема.

В залежності від фактичного рівня ієрархічної важливості об'єктів кібертравлі, можливі істотні відмінності в представництві групи «експериментатор».

При реалізації заходів кібербулінгу роль «жертви» зводиться до функцій демонстратора можливостей «агресора», стосовно змін мережевого статусу та коригування уявлень про норми мережевої поведінки представників найбільш численної рольової групи - «спостерігачів».

Головна мета представників рольової групи «агресор» при реалізації акцій кібербулінгу, це забезпечення абсолютного домінування в ролевій групі «спостерігачів».

Рівень ієрархічної значущості представників групи "жертв", обумовлює необхідний ступінь інтеграції (монополізації) функцій адміністрування контенту і основних складових ресурсного забезпечення сучасних інформаційних систем.

При високих рівнях ієрархічної важливості об'єктів кібертравлі, представниками рольової групи «експериментатор», є власники інтегрованих активів (перш за все інформаційних і технічних).

Об'єкти кібертравлі значною мірою корелюють з основними об'єктами впливу в ході ведення інформаційних операцій.

ЛІТЕРАТУРА

1. Цькування. // Wikipedia. URL: http://surl.li/omyd (Last accessed: 25.03.2021).

2. What is cyberbullying? // nuedusec. URL: https://nuedusec.com/blog/cyberbullying/ (Last accessed: 25.03.2021).
3. Интернет-травля. // Wikipedia. URL: http://surl.li/omyr (Last accessed: 25.03.2021).
4. Закон України «Про основні засади забезпечення кібербезпеки України» № 2163-VIII від 5 жовтня 2017 року. URL: https://zakon.rada.gov.ua/laws/show/2163-19 (Last accessed: 25.03.2021).
5. Vdovenko, S., Danik, Y., & Faraon, S. (2019). Дефініційні проблеми термінології у сфері кібербезпеки і кібероборони та шляхи їх вирішення. *Комп'ютерні науки та кібербезпека*, (1), 18-30. https://doi.org/10.26565/2519-2310-2019-1-02 (Last accessed: 25.03.2021).
6. Гайкова, В., & Малахов, С. (2020). Дослідження явища кібербулінгу і шляхів протидії його проявам. *Комп'ютерні науки та кібербезпека*, *1*(1), 14-32. https://doi.org/10.26565/2519-2310-2020-1-02 (Last accessed: 08.03.2021).
7. Макаренко С. И. Информационные операции: Фрагмент монографии. URL: https://psyfactor.org/psyops/makarenko-304.htm (Last accessed: 24.03.2021).
8. Інформаційна операція. // Wikipedia. URL: http://surl.li/onen (Last accessed: 25.03.2021).
9. Ролевая структура кибербуллинга. Wikipedia URL: http://surl.li/ocjv (Last accessed: 20.03.2021).
10. Эксперимент Милгрэма полвека спустя. URL: http://surl.li/onca (Last accessed: 20.03.2021).
11. Emilie A. Caspar, Julia F. Christensen, Axel Cleeremans, and Patrick Haggard Coercion Changes the Sense of Agency in the Human Brain. URL: https://www.cell.com/fulltext/S0960-9822(16)00052-X (Last accessed: 20.03.2021).
12. Експеримент Мілґрема. Wikipedia URL: http://surl.li/oubj (Last accessed: 21.03.2021).
13. Conducting the Milgram experiment in Poland, psychologists show people still obey. URL: https://www.eurekalert.org/pub_releases/2017-03/sfpa-ctm030917.php (Last accessed: 25.03.2021).
14. Почему люди готовы убивать по приказу начальства. URL: http://surl.li/onbe (Last accessed: 28.02.2021).
15. Информационное общество. // Wikipedia. URL: http://surl.li/onlr (Last accessed: 25.03.2021).
16. Информационное общество. Новая философская энциклопедия: в 4 т. / Электронная библиотека Института философии РАН. URL: http://surl.li/onnv (Last accessed: 25.03.2021).

ГАЙКОВА Валерія Валеріївна – студентка 5-го курсу, кафедра безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: valeriagaikova98@gmail.com; ORCID: 0000-0001-8065-3842.

Наукові інтереси:

– *безпека інформаційних систем, соціальний інжиніринг, протидія кібербулінгу*.

МАЛАХОВ Сергій Віталійович – к.т.н., с.н.с.; доцент кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: malakhov@karazin.ua; ORCID: 0000-0001-8826-1616.

Наукові інтереси:

– *інформаційна безпека, захист від несанкціонованих дій, обробка даних*.

УДК 004.051 **СЕКЦІЯ 6**

ГАЛЮК Я.О, ТОВСТОКОРЕНКО О.Ю

ПРО МЕТОДИ РОЗРОБКИ ТА КРИТЕРІЇ ОЦІНКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМ «РОЗУМНИЙ БУДИНОК»

Вступ

Сьогодні галузь розробки програмного забезпечення активно займається розвитком методів, які б охоплювали всі аспекти проекту, щоб задовольнити замовника. Для того, щоб дізнатися конкурентні переваги і вибрати метод, який відповідає конкретним потребам проекту, необхідно провести аналіз.

«Розумний будинок» – є окремою галуззю IoT, яка фокусується на домашніх пристроях і приладах. Для реалізації IoT-рішень, зокрема систем «Розумний будинок», використовують найбільш популярні підходи до розробки програмного забезпечення, це Scrum, Kanban, Scaled Agile Framework та відносно нові Ignite | IoT Methodology та IoT Methodology.

Основна проблема розробки полягає в тому, що люди мають різні потреби, і система повинна бути адаптована до більшості користувачів. Вирішити ці задачі можливо, збалансувавши критерії оцінки з урахуванням передбачуваного використання та проектування відповідного архітектурного рішення.

Методи розробки програмного забезпечення

Для реалізації рішень IoT існують підходи до розробки програмного забезпечення, такі як Scrum та Kanban. Проте, для вирішення проблем, пов'язаних з розподіленими, мобільними й незалежними (standalone), які можуть зростати в міру розвитку нового продукту і появи нових проблем, необхідна адаптивна методологія, таких як Scaled Agile Framework (SAFe). Незважаючи на це, проектні команди IoT можуть пропонувати свої власні методології відповідно до їх дослідженнями та досвідом, таким як Ignite | IoT Methodology і IoT Methodology.

У «14th Annual State of Agile Report»[1], статистичними даними показаний вибір цих методів, заснований на багатьох попередніх дослідженнях, який показує, що Scrum (75%) і Kanban (7%) є найбільш поширеними гнучкими методами, використовуваними організаціями, крім того, SAFe (35%) домінує над методологіями масштабування. З іншого боку, варто звернути увагу на Ignite | IoT Methodology і IoT Methodology [2], тому що це перші методи, створені в організаціях спеціально для IoT проектів.

Ignite, і IoT Methodology – це монолітні методи, які повторно використовують багато чинних загальних практик, комбінуючи їх з новими інноваційними практиками, що ускладнює повторне використання і поширення нових практик[3]. Однак цю проблему можна легко вирішити, піднявши їх на наступний рівень, зробивши їх обьеднуючими й звільнивши їх практики. (Практика визначається як повторюваний підхід до виконання чого-небудь з певною метою).

Вибір конкретного методу залежить від його розуміння, тобто від розуміння практики, процесу, ролей і та інших аспектів[4]. Визначимо деякі критерії, які дають загальне уявлення про кожен з обраних методі:

1) підхід: стратегія управління, яка означає, що команда керує та контролює процеси, компонентами їх організацію;

2) розмір команди: кількість членів команди розробників, необхідних для підвищення продуктивності при використанні даного методу;

3) ролі та обов'язки: необхідною умовою для виконання процесу – створення та інформування про те, хто які має зобов'язання;

4) циклічність: дозволяє вчасно привести контрольний показник регулярності;

5) філософія змін: вказує коли метод приймає зміни вимог в ході розробки проекту;

6) пріоритет: вимога, яка повинна виконуватися першою;

7) апаратне забезпечення: вказує, чи використовується даний метод для апаратного забезпечення;

© ГАЛЮК Я.О, ТОВСТОКОРЕНКО О.Ю, 2021

8) програмне забезпечення: вказує, чи використовується даний метод для програмного забезпечення.

Табл 1 Порівняльння методів розробки системи «Розумний будинок»

	Scrum	**Kanban**	**SAFe**	**Ignite**	**IoT methodology**
Підхід	Ітераційний	Ітераційний	Ітераційний	Не ітераційний	Ітераційний
Розмір команди	Малий	Залежить від проекту	50-124 осіб в кожному релізі (більше 1)	Великий	Великий
Циклічність	2-4 тижні	1 тиждень	6-12 тижні	-	-
Зміни	Після кожного спринту	В будь-який час	Після ітерації	-	Після ітерації
Ролі	Product owner Scrum master Крос-функціональна команда		Epic owner Entreprise		Користувач Дизайнер Виконавець Менеджер проекту
Пріоритет	Елементи заплановані в поточному спринту	Елементи на Kanban Board	Елементи заплановані в поточній ітерації	Безпека в IoT Проектування рішення	Створення прототипу рішення
Апаратне забезпечення	-	-	-	+	+
Програмне забезпечення	+	+	+	+	+

Критерії оцінки програмного забезпечення

Модель якості не може бути виражена простим включенням всіх функціональних вимог. Деякі атрибути якості підтримують один одного, а інші суперечать один одному. Канонічний приклад останнього – безпека і зручність використання [5].

Таким чином, було побудовано матрицю компромісів для деяких загальних атрибутів якості, актуальних для систем «Розумний будинок», які були отримані в результаті проведеного аналізу кореляції результатів опитування[6], де приведено експертні оцінки критичних атрибутів якості. На основі зазначених результатів опитування та методу аналізу ієрархій було побудовано порівняльну таблицю 2.

Двостороння стрілка (<=>) позначає – підтримує пару, а хрестик (X) зазначає конфліктну комбінацію (порожні клітинки означають байдужість). Адаптивність і зручність використання очевидні для користувача. Ефективність відноситься до обох категорій.

Табл 2 Компроміси між різними якісними характеристиками

	Адаптивність				
Портативність	<=>	Переносимість			
Зручність використання	<=>		Зручність використання		
Безпека	X	X	X	Безпека	
Безпечність			<=>		Безпечність
Ефективність	X	X	X	X	

Замість того, щоб просто декларувати важливі атрибути системи «Розумний будинок», важливо виділити причини вибору. Відправною точкою послугує продуктивність як функція пропускної здатності й затримки. Пропускна здатність – це загальний показник, що відноситься до кількості даних, які обробляються за одиницю часу. Усунення надлишкових переміщень даних, безумовно, збільшує пропускну здатність; те ж саме і зі швидкістю обробки даних. Затримка стає критичною в системах IoT в реальному часі, де також спрацьовує закон фізики (швидкість світла) [5]. Цей параметр визначає швидкість реакції керованої даними архітектури та може бути пов'язаний з довжиною петлі зворотного зв'язку (мається на увазі час прийому-передачі між пристроєм і додатком).

У наступному списку показані ті атрибути якості [6][7], які підтримують розгортання системи «Розумний будинок» в різних місцях:

Портативність. Система повинна бути портативною, якщо вона призначена для гетерогенних вузлів. Це може бути досягнуто шляхом використання технологій віртуалізації або упаковки результатів в приховану форму операційної системи хоста (наприклад, образ Docker). *Адаптивність.* Для підтримки великого списку пристроїв і надання додаткових сервісних API для цілей інтеграції обов'язкова наявність адаптації системи «Розумний будинок». Це означає, що рішення IoT має бути здатне взаємодіяти з усіма відомими технічними системами без надмірної настройки. Оскільки немає гарантії, що платформа буде обслуговуватися тими ж постачальниками, які її встановили. Але вона також повинна бути розширюваною: дозволяти безперервний потік даних і операцій, в той час, як кращі функції розширюються. *Зручність використання.* Щоб зменшити проблеми з розгортанням і швидко налаштувати користувачів на платформу IoT, вона повинна бути в зручній для користувача формі в багатьох аспектах. Сюди входять засоби керування, нагляду та звітності. *Ефективність*. Система «Розумний будинок» в ідеалі повинна використовувати передові технології зберігання даних і вимагати адекватних апаратних ресурсів для використання як в реальному часі, так і в звичайному контексті. *Безпечність*. Система «Розумний будинок» ніколи не повинна робити те, чого не повинна робити. Важливим чинником, що змінює правила гри стосовно програмного забезпечення в області Інтернету речей, є безпека в поєднанні з підзвітністю і відповідальністю. Будь-яка прикладна автоматизація за допомогою рішення IoT означає, що ми віримо в систему і впевнені, що вона ніколи не завдасть шкоди навколишньому середовищу. *Безпека.* Система «Розумний будинок» повинна забезпечувати належне управління пристроями (через механізми аутентифікації та авторизації), конфіденційність, цілісність і конфіденційність даних за допомогою безпечного зв'язку та шифрування даних. Безпека особливо важлива для платформи Інтернету речей, оскільки вона буде більше покладатися на автоматизовану безпеку. Завжди повинні бути передбачені заходи безпеки, що забезпечують передачу потокової інформації тим, хто її потребує, і якомога далі від тих, хто цього не робить. *Ремонтопридатність.* У сьогоднішній мінливій робочій середі важливо, щоб системні протоколи були зрозумілі новій команді співробітників. Систему необхідно оцінювати на предмет її мобільності. Платформи повинні бути добре організованими, інтуїтивно зрозумілими й інтерактивними, щоб забезпечити плавний перехід у міру того, як естафету перехоплюють нові адміністратори та розробники. Всі ці атрибути повинні бути збалансовані з урахуванням передбачуваного використання і включені в архітектуру.

Висновок

В цьому дослідженні проведено аналіз основних методів розробки та критерії оцінки програмного забезпечення «Розумний будинок». Не зважаючи на те що на сьогодні найпопулярніший та найбільш універсальний метод вважають Scrum, для розробки систем «Розумний будинок», у певних випадках він не найкращім. Це обумовлено тим, що даний метод не має критеріїв розробки з урахуванням інтеграції апаратного забезпечення, так як і Kanban та SAFe –критично для даної системи. Методології Ignite | IoT Methodology і IoT Methodology, що створено спеціалістами сфери IoT виключно для систем інтернет речей – найперспективніше у виборі. На основі проведеного аналізу було виділено основні критерії їх матрицю компромісів та значення кожного в оцінці програмного рішення для системи розумного будинку.

ЛІТЕРАТУРА

1. Digital.ai. 14th Annual State of Agile Report [Електронний ресурс] / digital.ai. – 2019. – Режим доступу до ресурсу: https://stateofagile.com/#ufh-i-615706098-14th-annual-state-of-agile-report/7027494.
2. Giray G. IoT System Development Methods [Електронний ресурс] / G. Giray, B. Tekinerdogan, E. Tüzün – Режим доступу до ресурсу: http://repository.bilkent.edu.tr/bitstream/handle/11693/51157/IoT_system_development_methods.pdf.
3. Jacobson I. Is There a Single Method for the Internet?, [Електронний ресурс] / I. Jacobson, I. Spence. – 2017. – Режим доступу до ресурсу: https://www.ivarjacobson.com/sites/default/files/field_iji_file/article/internet_of_things_cacm_published.pdf .
4. Merzouk S. IoT methodologies: comparative study [Електронний ресурс] / S. Merzouk, A. Cherkaoui. – 2020. – Режим доступу до ресурсу: https://www.sciencedirect.com/science/article/pii/S1877050920317841.
5. Mijic D. Scalable Architecture for the Internet of Things / D. Mijic, E. Varga. – 2018: O'Reilly Media, Inc.
6. Luor T. Exploring the Critical Quality Attributes and Models of Smart Homes [Електронний ресурс] / T. Luor, L. Hsi- Peng. – 2015. – Режим доступу до ресурсу: https://www.researchgate.net/publication/280447301_Exploring_the_Critical_Quality_Attributes_and_Models_of_Smart_Homes.
7. Matthew J. Evaluating and choosing an IoT platform [Електронний ресурс] / J. Perry Matthew. – 2016. – Режим доступу до ресурсу: https://www.oreilly.com/content/evaluating-choosing-iot-platform/.

ГАЛЮК Ярослав Олегович – студент кафедри моделювання систем та технологій, факультет комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків, Україна, 61022; e-mail: xa11867961@student.karazin.ua; ya.galyuk@gmail.com; ORCID: 0000-0001-9854-7597.

Наукові інтереси:

– методи розробки та критерії оцінки програмного забезпечення IoT.

ТОВСТОКОРЕНКО Олег Юрійович – аспірант кафедри програмної інженерії та інформаційних технологій управління, Національний технічний університет «Харківський політехнічний інститут», вул. Кирпичова, 2, Харків, Україна, 61022; старший викладач кафедри моделювання систем і технологій, Харківський національний університет імені В. Н. Каразіна., Проспект Незалежності, 6, Харків, , Україна, 61000. e-mail: oleg.tovstokorenko@karazin.ua; ORCID: 0000-0003-2664-1650.

Наукові інтереси:

– моделі та інструментальні засоби забезпечення варіабельності вимог в процесах розробки програмних систем; процеси управління варіабельністю програмних систем «Розумний будинок».

УДК 004.41.02 **СЕКЦІЯ 6**

ГАМЗАЄВ Р.О., ТКАЧУК М.В. , ШЕВКОПЛЯС Д.О.

ЗАСТОСУВАННЯ ЗНАННЯ-ОРІЄНТОВАНИХ МЕТОДІВ І ТЕХНОЛОГІЙ ДЛЯ МОДЕЛЮВАННЯ ВАРІАБЕЛЬНОСТІ В РОЗРОБЦІ ЛІНІЙОК ПРОГРАМНИХ ПРОДУКТІВ

Вступ

Сучасний стан досліджень та практичних розробок у галузі програмної інженерії характеризується появою нових методів та технологій створення програмного забезпечення (ПЗ), застосування яких має забезпечити появу на ІТ-ринку нових програмних систем (ПС) з меншими витратами, за коротший час і з відповідним рівнем їх якості. Одним з таких підходів є побудова лінійок програмних продуктів (ЛПП) [1], основною перевагою якого є: (1) можливість гнучкого врахування постійних змін потреб користувачів, (2) ефективне використання великої кількості різноманітних за своєю функціональністю програмних та апаратних компонентів для реалізації проблемно-орієнтованих ПС, наприклад в таких високотехнологічних галузях, як створення систем інтернету речей та систем типу «Розумний будинок» [2]. Для таких ПС необхідно забезпечити певний рівень варіабельності проектних рішень на всіх етапах їх повного життєвого циклу (ПЖЦ), і це передбачає розробку нових інтелектуальних підходів до моделювання різних варіантів їх структури та функціональності. Саме тому дослідження можливостей застосування знання-орієнтованих методів та технологій в розробці ЛПП є актуальною науково-технічною проблемою.

Аналіз особливостей процесу розробки лінійок програмних продуктів з урахуванням необхідності забезпечення варіабельності проектних рішень

На рис. 1 наведена загальновизнана структурна схема процесів розробки ЛПП з роботи К. Поля (K. Pohl et al.) [1], яка визначає 2 логічні рівні: (I) рівень доменної інженерії (Domain Engineering), на якому створюються такі варіабельні проектні рішення багаторазового використання як моделі вимог (requirements model), архітектурні моделі (architecture models), доменні компоненти (domain components) та моделі тестування (tests models); (II) рівень інженерії застосунків (Application Engineering), на якому на основі проектних рішень, отриманих на рівні (I) відповідно розробляються та програмно реалізуються конкретні специфікації вимог, системні архітектури, програмні компоненти та тести для побудови цільової ЛПП.

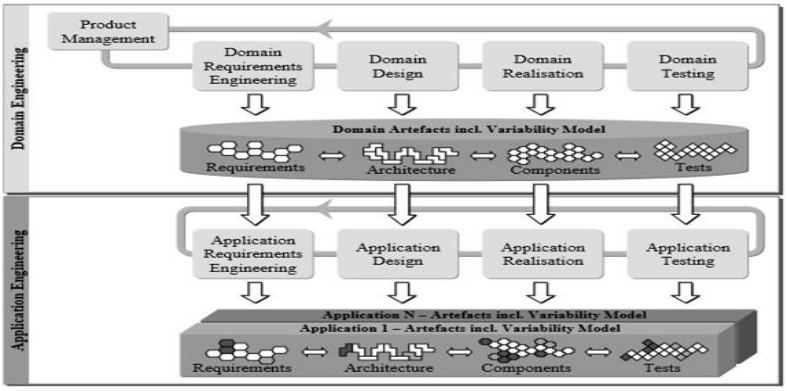


Рис. 1 Структурна схема процесів розробки ЛПП за К. Полем

Кожен із цих 2-х рівнів (I) і (II) має впорядковану послідовність фаз ПЖЦ (див. рис.1) і для їх автоматизації існує достатньо велика кількість методів розробки ПЗ та CASE-інструментів, стислий огляд яких наведено в [3]. Відповідно, постає задача, яким чином можливо забезпечити інтеграцію їх функціональності та результатів застосування на різних етапах ПЖЦ, і для цього пропонується розглядати ще один логічний рівень в структурі процесів розробки ЛПП, а саме це

© ГАМЗАЄВ Р.О., ТКАЧУК М.В. , ШЕВКОПЛЯС Д.О., 2021

проміжний шар опису таких *функціонально-семантичних категорій* (ФСК) як: (a) *«Потреби замовника»*, (b) *«Знання експертів»* та (c) *«Дії розробників».* Всі ці модельні категорії об'єднує те, що вони по своїй суті є слабоформалізованими та структурно-складними процесами і артефактами, для ефективного визначення яких потрібно застосовувати *когнітивні моделі* (cognitive models), *евристичні правила* (heuristic rules) та *алгоритми обробки експертних знань* (expert knowledge algorithms), які можливо узагальнити під терміном *знання-орієнтовані* (knowledge-oriented) методи і технології. При цьому термін «знання» у прикладному його значенні може бути визначений згідно рекомендацій Міжнародного наукового консорціуму *System Thinking World* [4]: це структурована сукупність інформаційних об'єктів та їх взаємозв'язків у поєднанні з семантичними правилами їх обробки та інтерпретації, які уможливлюють отримання нових фактів у визначеній предметній області (ПрО).

В [5] була запропонована технологічна схема розробки ЛПП із використанням конкретних знання-орієнтованих методів доменного моделювання: FODA (Feature-oriented Domain Analysis) і ODM (Object Domain Modeling) та технологій: FeatureIDE та Actifsource, а в [6] розроблена структурна модель процесів розробки ЛПП з урахуванням запропонованих ФКС (a)-(c), на прикладі методології Scrum для гнучкої розробки ПЗ. На основі поєднання цих двох підходів пропонується інтегрована модельно-технологічна схема розробки ЛПП, яка представлена на рис.2.

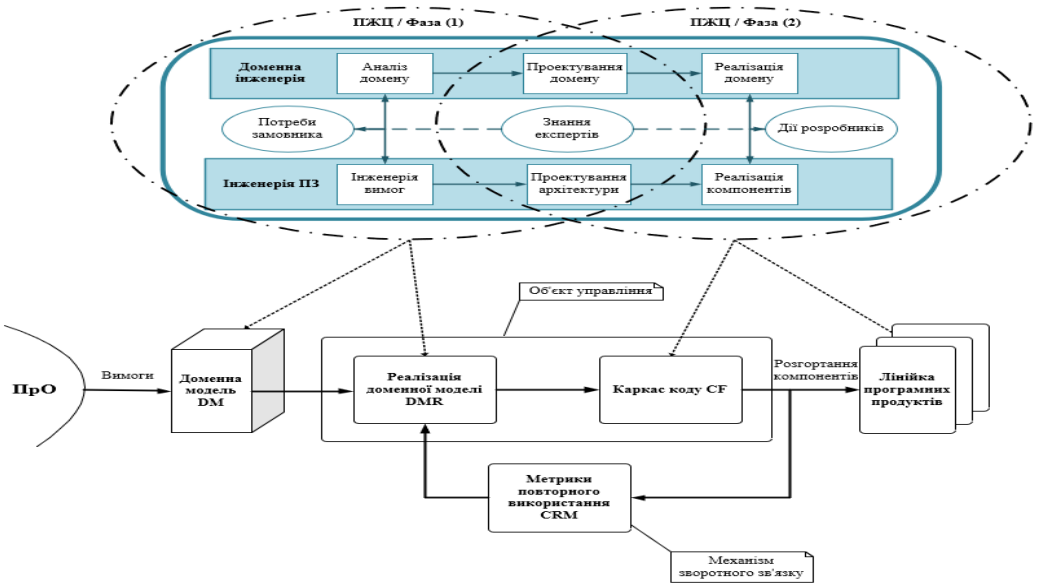


Рис. 2 Структурна модельно-технологічна схема знання-орієнтованої розробки ЛПП

На цій схемі (див. рис. 2), у верхній її частині, основні етапи ПЖЦ упорядковані у 2 логічно-взаємопов'язані фази, а саме:

- ПЖЦ / Фаза (I) – це етапи «Аналіз домену», «Інженерія вимог» та «Проектування домену», які відповідають функціонально-семантичній категорії *«Потреби замовника»*;
- ПЖЦ / Фаза (II) – це етапи «Проектування архітектури», «Реалізації домену» та «Реалізації компонентів», які відповідають функціонально-семантичній категорії *«Дії розробників».*

Ці дві фази ПЖЦ графічно виділені на рис. 2 двома великими штрихованим овалами, що перетинаються, оскільки з проектної точки зору вони технологічно є взаємопов'язаними за рахунок наявності таких проектних етапів як «Проектування домену» та «Проектування архітектури», які відповідають ФКС «*Знання експертів*». Таким чином, саме вона є тим семантичним «містком», який поєднує виконання етапів розробки ЛПП, що належать до ПЖЦ / Фази (I) та ПЖЦ / Фази (II), і таким чином пояснює доцільність застосування відповідних знання-орієнтовані методи і технології розробки ПЗ.

У нижній частині рис. 2, наведена загальна (на відміну від методу Scrum в [6]) технологічна схема розробки ЛПП, яка розглядається як система управління із зворотним зв'язком. Її графічне представлення пояснює те, що для технологічної реалізації всіх її основних

етапів можуть бути застосовані відповідні методи обробки експертних знань. Наприклад, в [7] для отримання доменної моделі (DM) використовується метод побудови репертуарних сіток (repertory grids), який дозволяє опрацьовувати знання декількох експертів щодо певної ПрО і сформувати вимоги у вигляді історій користувачів (User story) до функціональності майбутньої ПС. В подальшому, на основі застосування методів експертної оцінки складності побудованої DM можливо отримати каркас вихідного коду (code framework - CF) з необхідним коефіцієнтом його повторного використання CRM, що в свою чергу забезпечує підвищення ефективності всього процесу розробки та супроводу цільової ЛПП [8].

Функціональна схема знання-орієнтованої інформаційної технології для побудови доменних моделей в процесі розробки лінійок програмних продуктів

Для автоматизації комплексного використання вищенаведених знання-орієнтованих моделей та інструментальних засобів потрібно розробити відповідну інформаційну технологію (IT), функціональна схема якої в вигляді діаграми в нотації IDEF0 [9] представлена на рис. 3.

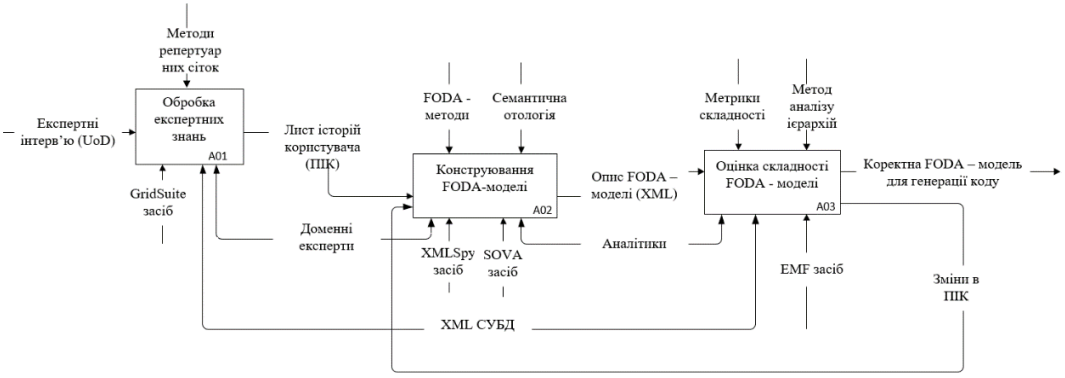


Рис. 3 Функціональна схема знання-орієнтованої IT для побудови ДМ

Основними функціональними блоками (ФБ) на цій діаграмі є наступні:

- A01 «Обробка експертних знань» - в ньому на основі обробки інформації з інтерв'ю експертів у певній ПрО за допомогою методів побудови репертуарних сіток [7] та з використанням інструментального засобу GridSuite, формується «Перелік історій користувачів (ПІК)», який передається на вхід наступного ФБ - A02;
- A02 «Конструювання FODA - моделі» - у ньому на основі інформації, яка міститься в ПІК з використанням методів доменного моделювання FODA та семантичної онтології визначеної ПрО, будується відповідна модель варіабельності доменних компонентів і для цього застосовуються такі засоби як редактор XMLSpy та засіб для побудови онтологій SOVA (Semantic and Ontological Variability Analysis), в результаті чого генерується опис FODA – моделі в форматі XML;
- A03 «Оцінка складності FODA – моделі» - у ньому, з використанням кількісних метрик оцінки складності ДМ [6] та методу аналізу ієрархій [10] для обробки експертних оцінок, обчислюється кількісний показник складності отриманої FODA – моделі; якщо він є прийнятним, то на основі цієї FODA – моделі виконується генерація каркасу вихідного коду (на Java, C# та ін.), якщо ж складність моделі є зависокою, то формуються рекомендації щодо змін в ПІК, які мають бути враховані в ФБ A02 (див. рис.3).

Слід зазначити, що для реалізації ФБ A01-A03 необхідно також розробити відповідні інтерфейси для таких груп користувачів як «Доменні експерти» та «Аналітики».

Висновки

Обґрунтована актуальність застосування знання-орієнтованих методів і технологій для побудови варіабельних ДМ і запропонована розширена 3-х рівнева модельно-технологічна схема процесу розробки ЛПП, яка дозволяє розглядати його як систему управління з зворотним зв'язком. Це дає можливість застосовувати кількісні метрики для оцінки складності побудованих ДМ і на підставі цього отримувати каркас вихідного коду для компонентів ЛПП з певними

показниками його якості. В подальшому планується розробити відповідні модельно-технологічні рішення для ефективного керування процесами виконання програмних компонентів ЛПП на етапі їх розгортання (deployment) та супроводу (maintenance).

ЛІТЕРАТУРА

1. Pohl K., et al: Software Product Line Engineering: Foundations, Principles, and Techniques., Springer, 2005. P. 467.
2. Saeed F, et al. IoT-based Intelligent Modeling of Smart - Home Environment // Journal of Sensor and Actuator Networks. 2019; 7(1). P. 23 – 39.
3. Ткачук М.В., Гамзаєв Р.О., Товстокоренко О.Т. та ін. Аналіз та експериментальне дослідження CASE - засобів для моделювання варіабельності в процесах розробки лінійок програмних продуктів // Системи управління, навігації та зв'язку: зб. наук. пр. ЦНДІ НУ. – К., 2020. – Вип. 1 (45). – С. 105-110.
4. Official Web-site of System Thinking World community, 2004. URL: http://www.systems-thinking.org/kmgmt/kmgmt.htm (Дата звернення: 05.04.2021).
5. Мартинкус І.О., Ткачук М.В., Гамзаєв Р.О. Конструювання лінійок програмних продуктів із застосуванням доменного моделювання та метрик повторного використання коду // Системи управління, навігації та зв'язку: зб. наук. пр. ЦНДІ НУ. – К., 2017. – Вип. 3(43). – С. 93-97.
6. Gamzayev R.O., Tkachuk M.V., Shevkoplias D.O. Knowledge-oriented Information Technology to Variability Management on Domain Analysis Stage in Software Development // Сучасні інформаційні системи, наук.-техн. ж-л, НТУ «ХПІ», том 2, № 4, 2020. – С. 39-47.
7. Gamzayev R.O., Tkachuk M.V., Shevkoplias D.O. Handling of Expert Knowledge in Software Product Lines Development with Usage of Repertory Grids Method. *Вісник Харківського національного університету імені В.Н. Каразіна, Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління».* - № 47, 2020. – С. 13-24.
8. Товстокоренко О. Ю., Гамзаєв Р. О., Ткачук М. В. Експериментальне дослідження ефективності застосування варіабельних проектних рішень на етапі супроводу програмного забезпечення систем «Розумний будинок». *Вісник Національного технічного університету "ХПІ" - Харків: НТУ "ХПІ".* – 2020. - № 2 (4) – С. 39-44.
9. Лаврищева К.М. Електронний підручник «Програмна інженерія» КНУ ім. Т. Г. Шевченка [Электронный ресурс] – Режим доступу: http://csc.knu.ua/uk/library/books/lavrishcheva-6.pdf (Дата звернення: 05.04.2021)
10. Саати, Т.Л. Принятие решений. Метод анализа иерархий. – М.: Радио и связь, 1993. С. 278.

ГАМЗАЄВ Рустам Олександрович - доцент, доцент кафедри моделювання систем і технологій; Харківський національний університет імені В.Н. Каразіна, м. Харків, Майдан Свободи 6, 61022; e-mail rustam.gamzayev@karazin.ua; ORCID: 0000-0002-2713-5664.

Наукові інтереси: *аналіз та моделювання розподілених програмних систем, моделі та технології розробки лінійок програмних продуктів, знання-орієнтовані методи розробки та інженерії якості програмного забезпечення.*

ТКАЧУК Микола Вячеславович - професор, завідувач кафедри моделювання систем і технологій; Харківський національний університет імені В.Н. Каразіна, м. Харків, Майдан Свободи 6, 61022; e-mail: mykola.tkachuk@karazin.com, ORCID: 0000-0003-0852-108.

Наукові інтереси: *концептуальне моделювання інформаційних систем і технологій, проектування проблемно-орієнованих програмних систем, знання-орієнтовані методи розробки та інженерії якості програмного забезпечення.*

ШЕВКОПЛЯС Дар'я Олександрівна - студентка кафедри моделювання систем і технологій; Харківський національний університет імені В.Н. Каразіна, м. Харків, Майдан Свободи 6, 61022; e-mail: shevkoplyasdarya23@gmail.com, ORCID: 0000-0001-5870-771X.

Наукові інтереси: *об'єктно-орієнтоване програмування, експертні методи обробки знань, розробка і реалізація баз даних та знань.*

УДК 004.056.55 **СЕКЦІЯ 5**

ГАРМАШ Д.В, МАЛЄЄВА А.А.

ГЕНЕРАЦІЯ КЛЮЧІВ І ПІДПИСУ ЗА ДОПОМОГОЮ БАГАТОВИМІРНОГО КРИПТОГРАФІЧНОГО АЛГОРИТМУ RAINBOW

Важливішою задачею у вивченні постквантової криптографії є вивчення механізму електронного підпису (ЕП) Rainbow, що є схемою ЕП, яка ґрунтується на криптографічному багатовимірному перетворенні (в квадратичних полях Галуа). Необхідність проведення досліджень ЕП Rainbow обґрунтовується тим, що назріла необхідність розроблення та прийняття постквантового національного стандарту ЕП, а також, що в процесі проведення конкурсу NIST США стосовно математичних основ застосування методу криптографічного перетворення Rainbow, отримані перспективні результати. Тому вважається важливим їх врахування та використання в Україні.

Вступ

Стандартні криптосистеми RSA та EC розроблені на основі складних математичних задач, таких як проста факторизація та дискретні проблеми логарифму.

Однак, як відомо, ці математичні завдання вирішуються за поліноміальний час великомасштабним квантовим комп'ютером. Тому потрібно побудувати криптографію, яка базується на нових математичних задачах і є стійкою до квантових комп'ютерів. Така криптографія називається постквантовою криптографією. У 2015 році Агентство національної безпеки (АНБ) оголосило план перехідного періоду до постквантової криптографії, і Національний інститут стандартів і технологій (NIST) розпочав публічний набір таких кандидатів з криптографії у 2016 році [22]. 22 липня 2020, Rainbow було обрано одним із трьох фіналістів на роль квантового підпису NIST.

Rainbow – це багатовимірна схема підпису, запропонована Дж. Діном та Д. Шмідтом у 2005 році [9]. Ця схема підпису може бути реалізована просто та ефективно за допомогою лінійних методів алгебри над невеликим кінцевим полем і, зокрема, створює коротші підписи, ніж ті, що використовуються в RSA або в інших постквантових схемах підпису.

Теоретична безпека Rainbow базується на тому, що вирішення набору випадкових багатовимірних квадратичних систем є складною проблемою. Математичною теорією є теорія багатовимірних багаточленів алгебраїчної геометрії.

Rainbow пропонує дуже малі підписи лише в кілька сотень бітів (лише 528 біт = 66 байт для I рівня безпеки NIST), які набагато коротші, ніж у інших (постквантових) схем підпису. Крім того, оскільки Rainbow використовує лише прості операції над невеликими скінченими полями, генерація та перевірка підпису надзвичайно ефективні.

I Рівні безпеки та вимоги до майбутніх постквантових ЕП

NIST США для оцінювання існуючих та перспективних проектів стандарту щодо ЕП запропоновані та уже використовуються такі рівні безпеки:

- I рівень – принаймні так важко зламати, як AES 128 (вичерпний перебір ключів);
- II рівень – принаймні так важко зламати, як SHA256 (пошук колізії);
- III рівень – принаймні так важко зламати, як AES192 (вичерпний перебір ключів);
- IV рівень – принаймні так важко зламати, як SHA384 (пошук колізії);
- V рівень – принаймні так важко зламати, як AES256 (вичерпний перебір ключів).

Основні вимоги до кандидатів на стандарти постквантових криптоперетворень конкретизуються у трьох таких напрямках:

- вимоги з безпеки (вимоги до стійкості до криптографічного аналізу);
- техніко-економічні вимоги (в основному щодо часової та просторової складностей);
- технічні характеристики реалізації алгоритмів асиметричних криптоперетворень.

© ГАРМАШ Д.В, МАЛЄЄВА А.А., 2021

II Сильні та слабкі сторони Rainbow

1) Короткі підписи. Підписи, отримані схемою підписів Rainbow, мають розмір, що приблизно вдвічі перевищує відповідний рівень безпеки. Тому Rainbow виробляє один з найкоротших підписів з усіх існуючих схем електронного підпису (як класичного, так і постквантового).

2) Скромні обчислювальні вимоги. Оскільки Rainbow вимагає простої операції лінійної алгебри над невеликим кінцевим полем, її можна ефективно реалізувати на пристроях з низькою вартістю, без необхідності криптографічного співпроцесору.

3) Простота. Конструкція схем Rainbow надзвичайно проста. Тому схема вимагає лише мінімальних знань з алгебри, щоб зрозуміти і реалізувати її. Ця простота також означає, що існує не так багато структурних схем, які можуть бути використані для атаки на Rainbow. Тому малоймовірно, що існують додаткові структури, які можуть бути використані для атаки схеми, які не були виявлені протягом більше, ніж 12 років ретельного криптоаналізу.

4) Суттєве випробовування часом. Як вже було зазначено, Rainbow базується на добре відомій UOV схемі, яка була винайдена ще у 1999 році. Саму схему Rainbow було створено у 2005 році, а остання атака, яка потребувала зміни параметрів сталася у 2008 році. За цей час було створено безліч спроб зламати цю схему, але досі Rainbow залишається однією з найбільш захищених схем підпису.

5) З іншого боку головним недоліком Rainbow є великий розмір відкритих та особистих ключів, виправлення якого є важливим в наших дослідженнях.

III Проблема розміру згенерованих ключів

Розробниками алгоритму було запроваджено новий підхід до алгоритму. Зараз Rainbow можна розділити на стандартний варіант (Standart Rainbow) і циклічний (Cyclic Rainbow). Проаналізувавши програмну реалізацію алгоритму, використовуючи різні параметри, які передаються в методи шифрування, можна побачити суттєвий прорив у скороченні розміру пари ключів.

Табл.1 Розміри ключів і підписів для стандартного Rainbow (Standart Rainbow)

Рівень	Параметри	Розмір відкритого ключа (kB)	Розмір закритого ключа (kB)	Розмір підпису (bit)
I	(GF(16),36,32,32)	157.8	101.2	528
III	(GF(256),68,32,48)	861.4	611.3	1.312
V	(GF(256),96,36,64)	1,885.4	1,375.7	1,632

Табл.2 Розміри ключів і підписів для циклічного Rainbow (Cyclic Rainbow)

Рівень	Параметри	Розмір відкритого ключа (kB)	Розмір закритого ключа (kB)	Розмір підпису (bit)
I	(GF(16),36,32,32)	58	101.2(99.0)	528
III	(GF(256),68,32,48)	258.4	611.3(603.0)	1.312
V	(GF(256),96,36,64)	523.5	1,375.7(1,361.8)	1,696

Висновки

Багато криптографів в даний час ведуть розробку алгоритмів, незалежних від квантових обчислень, тобто стійких до квантових атак. Ці задачі розглянуті на 2-му етапі конкурсу NIST США.

У зв’язку з можливістю появи потужного квантового комп’ютера актуальними є завдання створення постквантових алгоритмів ЕП. В цьому напрямі вже розпочаті дослідження, в певній мірі визначено математичні основи, на яких можуть бути побудовані постквантові алгоритми ЕП. Для цього можна застосувати схему Rainbow.

В процесі першого етапу конкурсу NIST, автори ЕП Rainbow вдосконалили алгоритм генерації ключів, який було подано у пропозиції першого раунду. Щоб прискорити процес генерації ключів Rainbow, автори перейшли від обчислення відкритого ключа шляхом інтерполяції до обчислення відкритого ключа за допомогою матриць. У процесі вдосконалення,

алгоритм було обмежено однорідними матрицями *S*, *F* і *T*. Вказане призводить до однорідного відкритого ключа *P*.

Rainbow виробляє один з найкоротших підписів з усіх існуючих схем електронного підпису, а надзвичайна простота конструкції схеми означає, що існує не так багато структурних схем, які можуть бути використані для атаки на Rainbow. Також не менш важливою є можливість ефективної реалізації на пристроях з низькою вартістю, що сприяє дослідженням для виправлення головного недоліку схеми Rainbow, а саме, великого розміру відкритих та особистих ключів.

В той же час загальновизнано, що складність прямих атак визначається лише однорідною частиною найвищого ступеня. Авторами запропоновано варіанти Rainbow, що є компромісними щодо ключів в частині їх розмірів та продуктивності.

Перший з компромісних варіантів з них позначається як "циклічна схема Rainbow", яка дозволяє зменшити розмір відкритого ключа схеми до 70% більш високої складності перевірки підпису. Другий з компромісних варіантів дозволяє легко зберігати особистий ключ на малому пристрої, але ціною ефективності процесу генерації підписів.

ЛІТЕРАТУРА

1. MQDSS [Електронний ресурс] / [S. Samardjiska, M. Chen, A. Hulsing та ін.]. – 2019. – Режим доступу до ресурсу: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-2/submissions/MQDSS-Round2.zip;

2. GeMSS: A Great Multivariate Short Signature [Електронний ресурс] / [A. Casanova, J. Faugere, G. Macario-Rat та ін.]. – 2019. – Режим доступу до ресурсу: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-2/submissions/GeMSS-Round2.zip;

3. І. Д. Горбенко, І.С. Кудряшов, В.В. Онопрієнко. Порівняльний аналіз пост квантових стандартів електронного підпису на основі мультиваріативних квадратичних перетворень // Радіотехника: всеукр. межвід. наук.-техн. зб. –Харків: ХНУРЕ. – 2018. – Вип. 195. – С. 46 60;

ГАРМАШ Дмитро Васильович – аспірант кафедри безпеки інформаційних систем і технологій, Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: donni.dima@gmail.com

Наукові інтереси:

– *криптологія, криптографічні системи та протоколи.*

МАЛЄЄВА Ганна Андріївна – аспірант кафедри БІТ, Харківський національний університет радіоелектроніки, проспект Науки, 14, Харків, Харківська область, Україна, 61166; e-mail: hanna.malieieva@nure.ua

Наукові інтереси:

– *криптографія, криптоаналіз та їх застосування з метою захисту інформації.*

УДК 621.548:621.311.24 **СЕКЦІЯ 1**

ГЕТМАНЕЦЬ О.М.

МОДЕЛЮВАННЯ ВИРОБІТКУ ВІТРОВОЇ ЕНЕРГЕТИЧНОЇ УСТАНОВКИ

Вступ

Проектування і будівництво нових потужних вітрових електричних станцій (ВЕС) розвивається неухильними темпами в усьому світі і в тому числі на Україні. Зростання вітроенергетичного сектора енергетики України в цілому за 2020 р. становило 8,3% і за станом на 31 грудня 2020 р. сумарна потужність усіх ВЕС України досягла позначки в 1110 МВт, що складає близько 2% від загальної потужності об'єднаної енергосистеми України [1]. Найважливішими завданнями при проектуванні нової ВЕС є: 1) вибір місця можливого будівництва ВЕС та оптимальної висоти розташування осі вітроколеса (ВК) вітряної енергетичної установки (ВЕУ); 2) вибір найбільш оптимальної ВЕУ з існуючих пропозицій на світовому ринку. Перше з цих завдань визначається даними вітрового кадастру обраної місцевості для потрібної висоти розташування осі ВК або, за їх відсутністю, іншими відомими способами. Друге завдання визначається, з одного боку, фінансовими можливостями інвестора при придбанні обраної ВЕУ, а з іншого – досягненням максимально можливого значення коефіцієнта використання встановленої потужності (КВВП) для обраної ВЕУ в конкретних вітрових умовах її експлуатації. Таким чином, комплексне розв'язання проблеми оптимального вибору і розміщення конкретної ВЕУ на заданій площі і оптимальній висоті осі ВК вимагає одночасного знання і врахування вітрових умов місцевості, характеристики потужності ВЕУ, а також їх взаємного впливу.

Ціль роботи

Моделювання аналітичної залежності КВВП ВЕУ від параметрів її характеристики потужності і вітрових умов на обраній місцевості розміщення ВЕС та на заданій висоті розташування осі ВК.

Матеріали і методи досліджень

Розв'язання поставленого завдання потребує дослідження і моделювання двох факторів – вітрових умов місцевості і характеристики потужності ВЕУ. У більшості досліджень (див., наприклад, огляд [2]), що були проведені в різні пори року на різних континентах, було встановлено, що диференціальна функція розподілу швидкості вітру $f(V)$ на певній висоті добре описується законом Вейбулла-Гнеденко [3, 4]:

$$f(V)=\frac{c}{b}\left(\frac{V-a}{b}\right)^{c-1}\exp\left[-\left(\frac{V-a}{b}\right)^{c}\right], \qquad (1)$$

де V – швидкість вітру; a – параметр зсуву; b – параметр масштабу; c – параметр форми розподілу. У цих дослідженнях було також встановлено, що нульова швидкість вітру (тобто штиль) практично завжди має місце, тому параметр зсуву сміливо можна покласти $a=0$. Таким чином, розподіл (1) для швидкості вітру залежить лише від двох параметрів b і c.

Разом з диференціальним розподілом Вейбулла-Гнеденко (ДРВГ) $f(V)$ (1) часто розглядається інтегральний розподіл (ІРВГ) $F(V)$:

$$F(V)=\int_0^V f(V)dV=1-\exp\left[-\left(\frac{V}{b}\right)^{c}\right]. \qquad (2)$$

Асимптотична оцінка виробітку ВЕУ P [Вт] зводиться до обчислення інтеграла:

$$P=\int_0^{\infty}\Phi(V)\cdot f(V)dV, \qquad (3)$$

© ГЕТМАНЕЦЬ О.М., 2021

де $\Phi(V)$ [Вт] – характеристика потужності ВЕУ, яка пов'язує виробіток електроенергії зі швидкістю вітру. Характеристика потужності більшості ВЕУ складається з трьох зон в залежності від швидкості вітру V: зона A ($0 \le V \le V_0$), де V_0 – стартова швидкість вітру, при якій ВЕУ починає працювати (для більшості ВЕУ $V_0 \approx 3 \div 4$ м/с); зона B ($V_0 < V \le V_N$), де V_N – номінальна швидкість вітру, при якій ВЕУ набуває номінальної потужності P_N; зона C ($V_N < V \le V_S$), в якій ВЕУ працює на номінальній потужності P_N, V_S – швидкість вітру, при досягненні якої відбувається аварійне відключення ВЕУ.

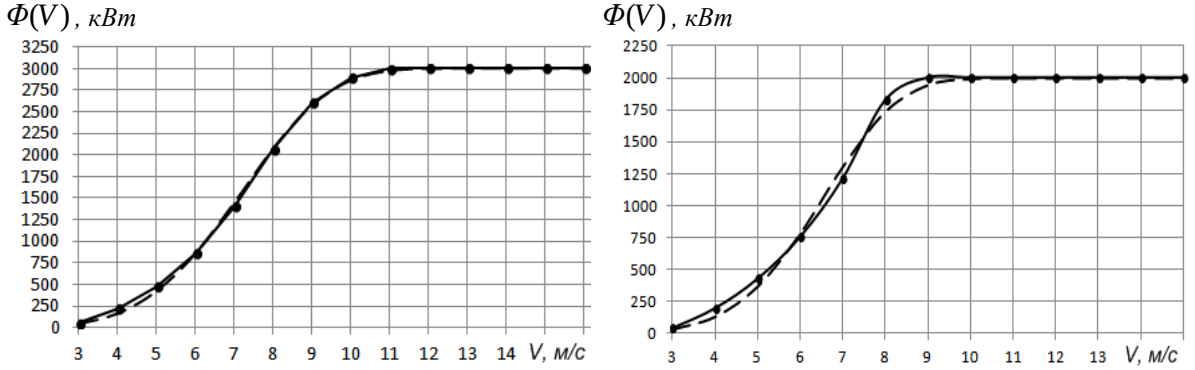
Зауважимо, що формула (3) має асимптотичний характер, оскільки інтегрування поширюється на півнескінченний інтервал зміни швидкості вітру $V \in [0, \infty)$.

Криві потужності ВЕУ добре відомі (якщо не становлять комерційної таємниці фірми-виробника) і, як правило, моделюються поліномами за V від третього ступеня і вище, лінеаризованими сегментованими моделями, моделями нейронних мереж і іншими статистичними моделями [5, 6]. Але отримані результати моделювання нічого не говорять про аналітичну залежності між КВВП ВЕУ і параметрами її характеристики потужності в певних вітрових умовах роботи.

Нами було зроблено припущення, що характеристики потужності більшості існуючих ВЕУ добре моделює ІРВГ (2). Це було перевірено для 86-ти типів ВЕУ різних виробників, які працюють в діапазоні потужностей від 2,0 МВт до 3,6 МВт (ми використовували базу даних thewindpower.net. [7]). При визначенні параметрів ІРВГ (2) для моделювання характеристики потужності ВЕУ був застосовний пакет «Пошук рішення» з Пакетів прикладних програм (ППП) MS Excel. Важливо, що для характеристик потужності всіх розглянутих типів ВЕУ було отримано значення параметра зсуву $a = 0$. Але, як показали проведені числові обчислення, гіпотетичним виробітком ВЕУ в зоні A (при $V \le V_0$) можна нехтувати.

Результати та їх обговорення

На рисунку 1 в якості прикладу наведені характеристики потужності двох ВЕУ: W2E 138 ($P_N = 3000$ кВт) фірми Fuhrlander (Німеччина) та SANY SE 12120 ($P_N = 2000$ кВт) фірми SANY (Китай), які порівнюються з результатами моделювання з використанням ІРВГ.



а) W2E 138: b = 7,70 м/с; c = 4,03; R^2 = 0,993 *б)* SANY SE 12120: b = 6,92 м/с; c = 4,89; R^2 = 0,955

Рис.1. Порівняння характеристик потужності двох типів ВЕУ (точки та суцільні криві) з результатами моделювання за допомогою ІРВГ (штрих-пунктир) при значенні параметра a = 0. Значення параметрів b і c та відповідні коефіцієнти детермінації R^2 наведені на рисунку

Для усіх розглянутих типів ВЕУ було встановлено, що параметри ІРВГ знаходилися в діапазоні: $a = 0$; $b \in [6{,}99 \text{ м/с}; 11{,}02 \text{ м/с}]$; $c \in [3{,}68; 4{,}89]$. При цьому середнє значення коефіцієнта детермінації виявилося дуже високим: $\overline{R}^2 = 0{,}976$. Тобто, ІРВГ надійно описує близько 97,6 % характеристики потужності ВЕУ, що добре ілюструє рис. 1.

Якщо поділити характеристику потужності ВЕУ $\Phi(V)$ на її номінальну потужність P_N, то отримаємо нормовану на одиницю характеристику потужності ВЕУ $P(V) = \Phi(V)/P_N$.

Таким чином, завдання моделювання КВВП ВЕУ зводиться до обчислення згортки двох РВГ з різними ядрами – ІРВГ $P(V)$ для турбіни і ДРВГ $f(V)$ для швидкості вітру:

$$\text{КВВП} = \int_0^\infty P(V)\cdot f(V)dV = 1-\int_0^\infty e^{-ky^p}e^{-y}dy = 1-I, \qquad (4)$$

де змінною інтегрування є $y=(V/b_w)^{c_w}$; показники $k=(b_w/b_t)^{c_t}$, $p=c_t/c_w$, індекс w (*wind*) відноситься до характеристик швидкості вітру, а індекс t (*turbine*) – до характеристики ВЕУ;

$$\text{інтеграл } I=\int_0^\infty e^{-ky^p}e^{-y}dy. \qquad (5)$$

Якщо розкласти експоненту e^{-y} в підінтегральному виразі (5) в ряд, то можна отримати аналітичний вираз для I (5) у вигляді нескінченного ряду гамма-функцій $\Gamma\left(\frac{n}{p}\right)$:

$$I=\int_0^\infty e^{-ky^p}e^{-y}dy=\frac{1}{p}\sum_{n=1}^{\infty}\frac{(-1)^{n-1}}{(n-1)!}\cdot\frac{1}{k^{\frac{n}{p}}}\Gamma\left(\frac{n}{p}\right). \qquad (6)$$

Даний ряд збігається досить швидко (вже перші 10 членів забезпечують відносну похибку не вище за 1 %), тому вираз (6) було застосовано для отримання чисельних оцінок КВВП за сумою перших 100 членів ряду в межах програми Maple-12 фірми Maplesoft.

В якості прикладу конкретних вітрових умов ми скористалися результатами вимірювань розподілу швидкості вітру, які були проведені на метеопосту «Іванівка» в Очаківському районі Миколаївської області в період з 15.08.2011 р. по 16.08.2012 р. (всього 8605 годин вимірювань) і оброблені з нашою участю. Були отримані наступні середньорічні значення параметрів розподілу швидкості вітру на висоті 100 м: $b_w = 8{,}294$ м/с; $c_w = 2{,}648$ [8]. При цьому показники $p=c_t/c_w$ і $k=(b_w/b_t)^{c_t}$ для усіх типів ВЕУ, що розглядалися, знаходилися в діапазоні: $p\in[1{,}39;\ 1{,}82]$, $k\in[0{,}35;\ 2{,}42]$.

На рисунку 2 наведені результати чисельних розрахунків КВВП 86-ти розглянутих ВЕУ для середньорічних вітрових умов метеопоста «Іванівка» (висота 100 м). Вони свідчать про те, що КВВП залежить головним чином від показника $k=(b_w/b_t)^{c_t}$ і плавно зростає одночасно з ним. Застосування команди «Додати лінію тренда» з ППП MS Excel призводить до наступного рівняння моделі кубічної регресії для КВВП за показником k:

$$\text{КВВП} = 0{,}1148+0{,}5648k-0{,}2572k^2+0{,}0445k^3. \qquad (7)$$

Це рівняння має дуже високе значення коефіцієнта детермінації $R^2 = 0{,}9974$.

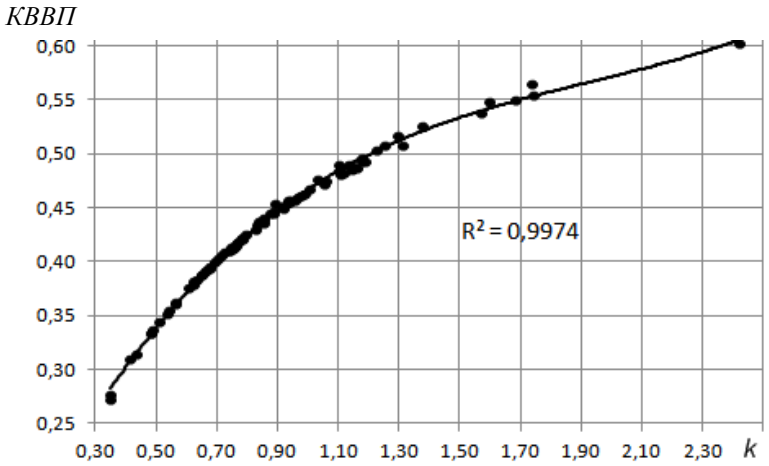


Рис.2. Порівняння розрахованих КВВП для 86-ти ВЕУ (точки) з передбаченнями моделі (7)

Графік моделі кубічної регресії (7) показано на рисунку 2 суцільною лінією. Максимальна розбіжність розрахованих і передбачених моделлю (7) значень КВВП складає 2 %. Найбільше значення КВВП (близько 0,6) для даних вітрових умов має ВЕУ SANY SE 12120 фірми SANY

(Китай) номінальної потужності 2000 кВт (крайня права точка на рисунку 2). Крива потужності цієї ВЕУ наведена на рисунку 1 б. Дана ВЕУ має найменше значення номінальної швидкості вітру (9 м/с), найменше значення параметру масштабу (b_t = 6,92 м/с) і тому найвище значенням показника k = 2,42 у порівняння з усіма іншими типами ВЕУ. Ліва границя зони *C* для цієї ВЕУ є найнижчою – тому вона більший час працює в зоні *C* на номінальній потужності 2000 кВт. Тобто, для заданих вітрових умов ця ВЕУ є оптимальною.

Висновки

1. На основі аналізу характеристик потужності 86-ти типів ВЕУ провідних світових виробників, що працюють в діапазоні 2,0 МВт – 3,6 МВт, вперше доведена можливість застосування інтегрального розподілу Вейбулла-Гнеденко для моделювання даних характеристик. При цьому середнє значення коефіцієнтів детермінації становило $\overline{R}^2 = 0{,}976$.

2. Доведено, що асимптотичне значення коефіцієнта використання встановленої потужності ВЕУ визначається в основному показником $k = \left(\frac{b_w}{b_t}\right)^{c_t}$, де b_w – параметр масштабу ДРВГ швидкості вітру; b_t – параметр масштабу ІРВГ характеристики потужності ВЕУ; c_t – параметр форми ІРВГ характеристики потужності ВЕУ.

3. Отримана проста кубічна за показником *k* модель для КВВП, передбачення якої з точністю до двох відсотків збігаються з результатами числових розрахунків.

4. Модель може бути застосована для обґрунтованого вибору оптимальної за своїми параметрами (b_t і c_t) ВЕУ для розміщення на обраній місцевості на заданій висоті розташування осі ВК за даними вітрового кадастру (b_w і c_w) та оцінювання виробітку цієї ВЕУ.

ЛІТЕРАТУРА

1. НЕК «Укренерго». Встановлена потужність енергосистеми України на 31/12/2020. URL: https://ua.energy/vstanovlena-potuzhnist-energosystemy-ukrayiny/ (дата звернення 01.04.2021).
2. Manwell J. F., McGowan J, Rogers A. Wind energy explained: theory, design, and application – 2nd ed. Chichester: John Wiley &Sons Ltd., 2008. P. 59 – 61.
3. Weibull W. A statistical distribution function of wide applicability. *Journal of Applied Mechanics*. 1958. vol. 18(3). P. 293–297.
4. Гнеденко Б. В. Предельные законы для сумм независимых случайных величин. *Успехи математических наук*. 1944. № 10. С. 115–165.
5. Lydia M., Kumar S. S., Selvakumar A. I. et al. Comprehensive review on wind turbine power curve modeling techniques. *Renewable and Sustainable Energy Reviews*. 2014. Vol. 30. P. 452–460.
6. Sohoni V., Gupta S. C., Nema R. K. A Critical Review on Wind Turbine Power Curve Modelling Techniques and Their Applications in Wind Based Energy Systems. *Journal of Energy*. 2016. No 1. P. 1–18.
7. URL.: https://www.thewindpower.net/ (дата звернення 01.04.2021).
8. Спосіб оцінювання виробітку вітряної електричної турбіни: пат. 135302 Україна. МПК6 F03D 1/00, G01P 5/00. № 2019 00597; заявл. 21.01.2019; опубл. 25.06.2019, Бюл. № 12. 6 с.

ГЕТМАНЕЦЬ Олег Михайлович – к. ф.-м. н., доцент; доцент кафедри штучного інтелекту та програмного забезпечення; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: oleg.m.getmanets@karazin.ua; ORCID: 0000-0002-0543-0961.

Наукові інтереси:– *математичні моделі в енергетиці, нафтовидобуванні та екології*

УДК 004.85:616.07 **СЕКЦІЯ 2**

ГОЛОВЧЕНКО В.Б., СТРІЛЕЦЬ В.Є.

ВИПАДКОВИЙ ЛІС – УНІВЕРСАЛЬНИЙ МЕТОД МАШИННОГО НАВЧАННЯ

Випадковий ліс є одним із універсальних алгоритмів машинного навчання, оскільки може бути застосований для розв'язання різноманітних задач, таких як, класифікація, регресія, кластеризація, відбір ознак тощо.

Перші роботи Лео Бреймана та Адель Катлер, пов'язані з побудовою ансамблів дерев ухвалення рішень, були опубліковані на початку 90-х років минулого століття. Їх роботи засновані на використанні евристичних процедур. Застосовувався також підхід, заснований на бустінгу (boosting) дерев ухвалення рішень. Більш суворий підхід, названий методом випадкових підпросторів, був розвинений в роботах Тін Кам Хо [1]. Суть підходу полягає в тому, що для побудови кожного дерева використовується тільки фіксована частка випадково відібраних ознак. Ансамблі побудованих дерев Хо назвав випадковими лісами рішень (Random Decision Forests).

Пізніше, Лео Брейман запропонував метод, відомий як випадковий ліс [2]. Визначення, дане ним, є досить загальним: випадковим лісом називається класифікатор, що складається з набору дерев $\{h(x, \Theta_k), k = 1,...\}$, де Θ_k – незалежні однаково розподілені випадкові вектори й кожне дерево вносить один голос при визначенні класу x.

Метод заснований на побудові великої кількості (ансамблю) дерев ухвалення рішень (один із параметрів методу), кожне з яких будується за вибіркою, що отримується з початкової навчальної вибірки за допомогою бутстрепу (тобто вибірки з поверненням).

Алгоритм індуктивної побудови й використання випадкового лісу вперше був розроблений Л. Брейманом і А. Катлер і реалізований в ряді комерційних пакетів. Алгоритм індукції випадкового лісу може бути представлений таким чином:

1. Для $i = 1,2,...,B$ (де B — кількість дерев в ансамблі) виконати:
- Сформувати бутстреп вибірку S розміру l по початковій навчальній вибірці $D = \{\mathbf{x}_i, y_i\}_{i=1}^{l}$;
- По бутстреп вибірці S викликати не усічене дерево рішень T_i з мінімальною кількістю спостережень в термінальних вершинах рівним n_{min}, рекурсивно виконуючи наступний алгоритм:
  a. з початкового набору n ознак випадково вибрати p ознак;
  b. з p ознак вибрати ознаку, яка забезпечує найкраще розщеплення;
  c. розщепити вибірку, відповідну оброблюваній вершині, на дві підвибірки;
2. В результаті виконання кроку 1 отримуємо ансамбль дерев рішень $\{T_i\}_{i=1}^{B}$;
3. Передбачення нових спостережень здійснювати так:
   a. для регресії:
$$\hat{f}_{rf}^{B}(x) = \frac{1}{B}\sum_{i=0}^{B} T_i\,(x);$$
   b. для класифікації: нехай $\omega\hat{}_i(\mathbf{x}) \in \{\omega_1,\omega_2,...,\omega_c\}$ – клас, передбачений деревом рішень T_i, тобто $T_i(\mathbf{x}) = \omega\hat{}_i(\mathbf{x})$; тоді $\omega\hat{}_{rf}^{B}(\mathbf{x})$ — клас, який найчастіше зустрічається у множені $\{\hat{\omega}_b(\mathbf{x})\}_{i=1}^{B}$.

Метод випадкових лісів має такі *переваги*:

– гарантія захисту від перенавчання навіть в тому випадку, коли кількість ознак перевищує кількість спостережень. Ця властивість є унікальною серед інших методів класифікації та є надзвичайно цінною для розв'язання прикладних задач;

– для побудови випадкового лісу по навчальній вибірці потрібно задати лише два параметри: розмір ансамблю та кількість випадково підібраних ознак, за якими виконується розщеплення в вершинах;

© ГОЛОВЧЕНКО В.Б., СТРІЛЕЦЬ В.Є., 2021

– оцінка ймовірності помилкової класифікації випадкових лісів на основі спостережень, що не входять до навчального бутстреп вибірки, але використані для побудови дерев, забезпечується методом Out-Of-Bag (OOB), запропонований Брейманом [3]. Ці спостереження називаються OOB вибірками;

– випадкові ліси можуть використовуватися не тільки для задач класифікації і регресії, а й для задач виявлення найбільш інформативних ознак, кластеризації, виділення аномальних спостережень і визначення прототипів класів;

– навчальна вибірка для побудови випадкового лісу може містити ознаки, виміряні за різними шкалами: числовими, порядковими і номінальними, що неприпустимо для багатьох інших класифікаторів;

– метод допускає легку паралелізацію (тобто програмну реалізацію, придатну для паралельних обчислень), що має велике значення при великому розмірі навчальної вибірки.

Окрім зазначених вище переваг методу випадкових лісів, також можна окремо виділити особливості, які сприяли поширенню застосування методу:

1. Для випадкових лісів застосовується подвійна ін'єкція випадковості: баггінг і використання методу випадкових підпросторів при розщепленні кожної вершини. Внаслідок цього значно підвищується точність методу.
2. Досить легка організація паралельних обчислень.
3. Невелика кількість параметрів методу забезпечує простоту застосування.
4. Відсутність процесу обрізання дерева, що призводить до збільшення обчислювальної ефективності.
5. Відсутня проблема перенавчання.

Існує кілька різновидів випадкових лісів.

Випадкові ліси виживання. Узагальненням випадкових лісів на випадок даних, цензурованих справа, важливим прикладом яких є дані про виживання, є випадкові ліси виживання (Random Survival Forests). З їх допомогою можлива побудова непараметричних регресійних кривих виживання, що не припускають пропорційності ризику.

Квантильні ліси регресії. У разі *безперервного* відгуку випадкові ліси є гарним методом побудови непараметричної регресії, тобто дають добре наближення до умовного середнього відгуку. В роботі [73] запропоновано важливе узагальнення – квантильні ліси регресії (Quantile Regression Forests), які дають повнішу картину умовного розподілу відгуку, а саме, можливість непараметричної оцінки квантилів умовного розподілу. Це дозволяє, зокрема, використовувати їх для побудови довірчих інтервалів і виявлення аномальних значень відгуку.

Логічні ліси. В роботі [4] запропонований метод побудови ансамблів класифікаторів, заснованих на логічних деревах ухвалення рішень. *Теоретичною* основою є логічна регресія [5], що припускає, що всі ознаки, включаючи класову, є бінарними. Передбачення відгуку в логічній регресії здійснюється з використанням логічних комбінацій бінарних ознак, дозволяючи будувати гнучкі моделі складних взаємодій ознак. Однак при наявності шуму можливості логічної регресії йдуть на спадок. Крім класифікації логічні ліси можуть використовуватися для виявлення найбільш інформативних ознак і їх взаємодій.

Випадковий наївний байєсівський класифікатор і випадкова поліноміальна логіт-модель. Експериментальні дослідження і практика застосування випадкових лісів показали, що метод має низку хороших якостей – високу точність, робастність щодо зашумленості навчальної вибірки й відсутність перенавчання. У зв'язку з цим природно припустити, що основні елементи алгоритму Бреймана – формування підвибірок методом баггінг і випадковий відбір ознак на стадії розщеплення вершин дерев рішень можуть бути успішно використані для деяких інших базових класифікаторів. Ця ідея була реалізована в роботі [6]. У цій роботі як базові класифікатори розглядалися наївний байєсівський класифікатор і поліноміальна логіт-модель. Авторами проведено ряд експериментів в порівнянні точності класифікації відповідних ансамблів з точністю випадкових лісів і машин опорних векторів, які показали, що випадковий наївний

байєсівський класифікатор і випадкова поліноміальна логіт-модель в певних ситуаціях перевершують ці методи.

На основі проведеного аналізу методу випадкових лісів було прийняте рішення використати даний метод для розв'язання задачі класифікації і прогнозування стану пацієнтів у системах медичного моніторингу.

ЛІТЕРАТУРА

1. Ho, Tin Kam. Random decision forests // Proceedings of 3rd International Conference on Document Analysis and Recognition. – 1995. – vol.1. – P. 278–282.
2. Breiman, Leo. Random Forests // Machine Learning: journal. – 2001. – Vol. 45, no. 1. – P. 5 – 32.
3. Breiman L. Out-of-bag estimation // Technical report, Statistics Department University of California, Berkeley. 1996. P. 1– 13.
4. Wolf B. J., Slate E. H., Hill E. G. Logic Forest: An ensemble classifier for discovering logical combinations of binary markers
5. Ruczinski I., Kooperberg C., LeBlanc M. Logic Regression methods and software / Eds: D. Denison et al. // Proceedings of the MSRI workshop on Nonlinear Estimation and Classification. New York: Springer, 2002. P. 333– 344.
6. Prinzie A., Poel D. Random Multiclass Classification: Generalizing Random Forests to Random MNL and Random NB // Working paper, Department of Marketing, Ghent University, 2007

ГОЛОВЧЕНКО Віталій Борисович – студент кафедри прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: xa11867788@student.karazin.ua, ORCID 0000-0002-0008-3999

Наукові інтереси:

- *методи машинного навчання;*
- *методи штучного інтелекту.*

СТРІЛЕЦЬ Вікторія Євгенівна – к.т.н.; доцент кафедри прикладної та теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: viktoria.strilets@karazin.ua; ORCID: 0000-0002-2475-1496.

Наукові інтереси:

- *методи машинного навчання;*
- *методи штучного інтелекту.*

УДК 621.327:621.391 **СЕКЦІЯ 5**

ГОНЧАРОВ М.О, МАЛАХОВ С.В.

МОДЕЛЮВАННЯ ПРОЦЕДУР ПІДГОТОВКИ ДАНИХ СТЕГАНОАЛГОРІТМА З БАГАТОРІВНЕВИМ МУЛЬТИПЛЕКСУВАННЯМ КОНТЕНТУ

Введення

Добре відомо, що одним із ефективних напрямів забезпечення приховування фактів передачі і зберігання інформації, є застосування різних стеганографічних методів [1]. Цифрова стеганографія, як окремий науковий напрямок, вивчає можливості використання властивостей цифрового контенту різного типу для забезпечення більш ефективного вирішення завдань, які пов'язані с синтезом нових методів і способів прихованої передачі, зберігання та маркування цільової інформації (надалі контенту). Принциповим є те, що в незалежності від використаного напрямку стеганографії, необхідно забезпечувати мінімізацію демаскуючих аномалій використовуваних контейнерів та підтримувати заданий рівень стійкості контенту стовно спроб його неавторизованої екстракції, а в деяких випадках, і стійкості до спроб навмисного спотворення контейнерів.

Основна частина

При приховувані в цифрових об'єктах будь-якої іншої – корисної інформації, нажаль, виникають певні спотворення цих об'єктів – переносників даних [1]. При збалансованих налаштуваннях відповідного стеганоалгоритму, можливо забезпечувати рівень спотворень контейнерів, які знаходяться нижче порога чутливості зорової системи людини [2] або чутливості його слухової системи, що забезпечує фактичну відсутність візуально або аудіо помітних змін (аномалій) переносників інформації (контейнерів). Тим самим забезпечується необхідний баланс між збереженням характерних властивостей для використовуваного типу контейнерів (у нашому випадку, напівтонових зображень [2]), та величиною допустимих спотворень, прийнятних для заданого типу прихованого контенту (в даному випадку зображень з різною вірогідністю перепаду яскравості між сусідніми елементами). В цьому сенсі зрозуміло, що кількість і інтенсивність проявів різних артефактів зображень контейнеру і контенту, знаходяться в прямий залежності від коректності обраних для них режимів обробки. При цьому, слід мати на увазі, що при обробці даних контейнерів і контенту можуть бути використані, як однотипні (симетричні) режими обробки, так і режими, які реалізують різні параметри обробки даних (асиметричні). Такими відмінностями можуть бути: – відмінності в розмірах блоків; – відмінності в параметрах попередньої обробки масивів даних контейнерів і контенту (що є головним змістом даного матеріалу); – відмінності в критеріях оцінки значущої інформації контейнерів і контенту; – відмінності реалізацій прискорення обчислювальних процедур та інше. Таким чином, на одних і тих же типах даних (в даному випадку зображень) можна отримати різний ефект, з точки зору помітності артефактів, що зменшує зайвий привід задуматися про причини появи цих аномалій даних [1-2].

В основу дослідного алгоритму положено принцип внутрішньо-кадрового стиснення зображень, який передбачає послідовне використання двох різних методів: 1 - методу кодування довжин серій [3, 4]; 2 - методу кодування з перетворенням, а саме ДКП (дискретного косинус перетворення) [2, 4]. Використання властивостей ДКП дозволяє «вбудовувати» контент в матриці коефіцієнтів перетворення контейнерів, визначених алгоритмом попередньої обробки, як опорний блок (тобто перший в серії подібних). Процедура інкапсуляції стеганоконтенту має два рівня реалізації – внутрішньоблоковий та міжблоковий. Перший рівень реалізується за результатами проведення селекції коефіцієнтів перетворення [4-5] для контейнера і контенту (можливі відмінності в параметрах обробки). На кожному з цих рівнів здійснюється змішування коефіцієнтів (на 1-му рівні) або блоків (на 2-му рівні) у відповідності з маскою мультиплексування, яка притаманна для кожного з рівнів обробки. Використання методів кодування з перетворенням забезпечує отримання матриць спектральних коефіцієнтів, в яких

© ГОНЧАРОВ М.О, МАЛАХОВ С.В., 2021

значна частина коефіцієнтів може бути виключена з подальшої обробки, а врахування властивостей людського зору, дозволяє апроксимувати коефіцієнти перетворення з малими втратами якості відновлюваних [2, 4, 5].

Для авторизованої екстракції контенту потрібна інформація о параметрах внутрішньо та міжблокового мультиплексу коефіцієнтів і опорних блоків. У разі її отримання здійснюється відновлення вихідної матриці блоків контенту (довжин серій опорних блоків), та проводиться зворотне ДКП. Відомості стосовно розмірів блоків, значення параметра маскування коефіцієнтів з координатами (0;0), та параметрів маски внутрішнього і міжблокового мультиплексу, складають ключову інформацію для легітимного відновлення контенту.

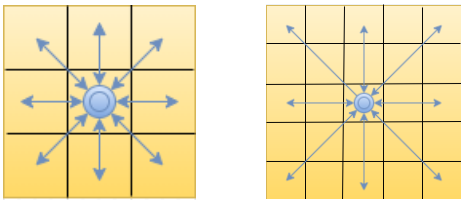
Для протидії процедурам неавторизованого вилучення контенту в дослідному алгоритмі використано спрощений механізм міжблокового мультиплексування (тільки для опорних блоків контенту). Згідно з ним, опорні блоки контенту вбудовуються в контейнери не послідовно, а відповідно до варіанту маски мультиплексу, яка зависить від поточного співвідношення кількості опорних блоків контейнера і контенту, та є варіативним елементом складеного ключа вилучення прихованої інформації. Окреме мультиплексування елементів з координатами (0;0) не проводилося.

Всі процедури дослідного алгоритму поділені на окремі етапи, кожен з яких, має своє функціональне значення. Основним змістом даного етапу є дослідження варіантів реалізації процедур попередньої обробки вихідних масивів даних для контейнерів і контенту. Головним завданням етапу, є зменшення кількості візуально малопомітних перепадів яскравості елементів вихідних зображень за рахунок згладжування їх малоінформативних областей. Використання процедур перетворення структури вхідних даних дозволяє забезпечити необхідні умови для одночасного вирішення 3-х завдань: - зменшення обчислювальної складності алгоритму; - ускладнення процедур аналізу і неавторизованої екстракції прихованого контенту (багаторівневий мультиплекс даних); - створення сприятливих умов (в сенсі різниці кількості опорних блоків) для наступного формування серій подібних блоків.

Варіанти обробки та отримані результати

Досліджено три варіанти реалізації процедур згладжування, з різним порядком взаємного порівняння складових елементів блоків, та різним способом оновлення їх змісту при перевищенні заданих критеріїв подібності (для контейнера і контенту). Особливу увагу звернено на важливість локалізації присутності контурів (їх фрагментів) в блоках які обробляються. Для всіх досліджуваних варіантів проводилося розбиття вхідних масивів даних на квадратні блоки (маска згладжування) $m \times m$, де m = 3, 5 та 7 елементів. Після цього в кожному блоці, кожного з варіантів, проводилась відповідна оцінка різниці значень яскравості для всіх складових елементів цього блоку. Для всіх варіантів обробки, значення порога загрублення яскравості складових елементів (P_Z), дискретно змінювалося в діапазоні 3, 5, 7, 16, 32 та 64 градацій яскравості. Також всі варіанти згладжування і зміни параметрів *«m»* та *«P_Z»* перевірялись для 3-х типів зображень, з різною ймовірність перепаду яскравості (p) між сусідніми елементами: – мнемосхема (0.03÷0.01), портрет (0.05÷0.03) та пейзаж (0.1÷0.05).

В 1-му варіанті згладжування, здійснювалась оцінка різниці значень яскравості центрального елемента з його периферійним оточенням (Рис. 1(а)). У разі, якщо різниця менше заданого значення порога загрублення (P_Z), то елемент в цій позиції заміщався значенням яскравості центрального елементу.



а) *б)*

Рис. 1 Варіанти обробки елементів (для маски 3×3 (а) та 5×5 елементів (б))

На рис. 3 наведено результати використання першого варіанту згладжування для маски згладжування 5×5 елементів (Рис. 1) для тестового зображення типу портрет (Рис. 2). З аналізу Рис. 3 видно, що дана реалізація процедур згладжування вимагає певного доопрацювання, оскільки на зображенні візуально помітні певні дефекти (уривчастість ліній і розмитість).

Згідно другого варіанту обробки, якщо периферійні значення перевищують яскравість центрального елемента більш ніж на величину P_Z, , то виконуємо заміну всіх значень даного блоку на середнє значення яскравості елементів для цього блоку. В результаті, отримуємо поліпшену версію 1-го варіанту (див. Рис. 4).

Рис. 2 Вихідне зображення типу «Портрет»

Рис. 3 Перший варіант обробки (маска 5×5; P_Z =3)

Рис. 4 Другий варіант обробки (маска 5×5; P_Z =3)

Рис. 5 Третій варіант обробки (маска 5×5; P_Z =3)

Третій варіант обробки, забезпечує перебір «всіх з усіма», де головним завданням є перевірка на наявність елементів, різниця між якими перевищує значення P_Z. При наявності такої різниці даний блок залишаємо без змін (можлива присутність контуру), в іншому випадку робимо заміну всіх значень на середнє значення яскравості всіх його елементів. В результаті, отримуємо згладжене зображення зі збереженими контурами (Рис.5). При використанні 3-го варіанту, встановлено, що основні контури зображення утримуються до P_Z = 14.

Для порівняння дослідних варіантів згладжування використано значення середньоквадратичної помилки (СКП), яке надає можливість оцінити ступінь спотворень вихідних зображень після їх обробки (чим менше СКП тим менш помітні аномалії і тим краще отриманий результат). На рис. 6-7 представлено вихідне тестове зображення типу «Пейзаж» та результат його обробки з використанням 1-го варіанту згладжування (найгіршого, як показали наступні результати моделювання). Як видно з рис. 7, навіть при використанні великої матриці згладжування і явно завищеного значення P_Z, візуальна помітність спотворень знаходиться на прийнятному рівні (за дрібними винятками), а саме зображення не зазнало критичних змін структури фону і контурів основних деталей. На рис. 8-9 представлені результати обробки зображень для різних варіантів згладжування. Час виконання для кожного з трьох варіантів обробки зображення представлено на рис. 10.

Рис. 6 Вихідне зображення типу «Пейзаж»

Рис. 7 Перший варіант обробки (маска 5×5; P_Z =32)

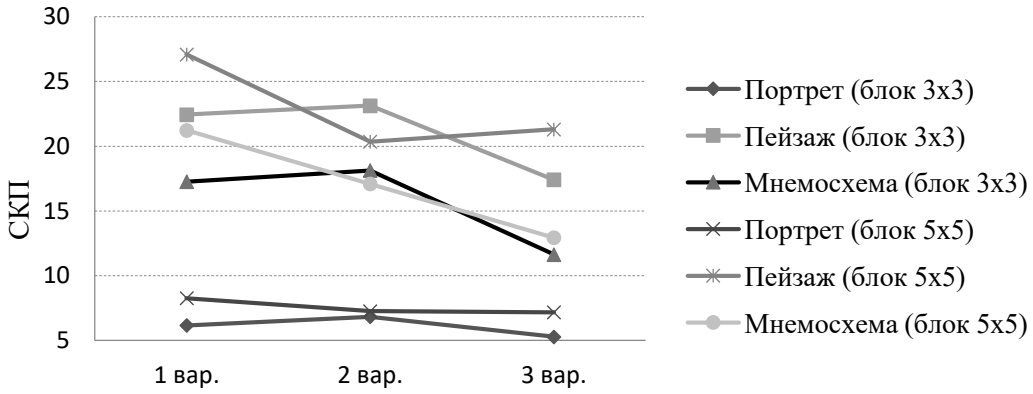


Рис. 8 Значення СКП для різних типів зображень і матриць згладжування (при P_z = 3)

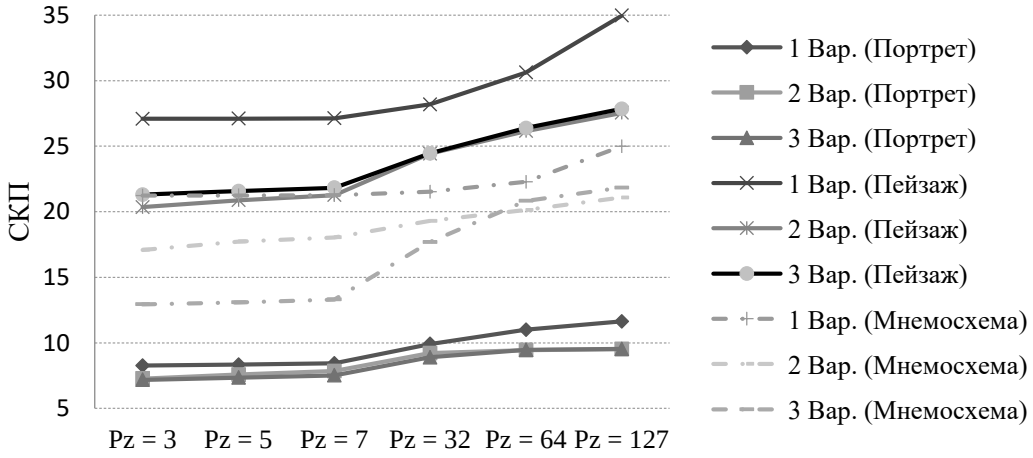


Рис. 9 Залежність СКП від P_Z для різних типів зображенньв (для блоків 5×5 ел.)

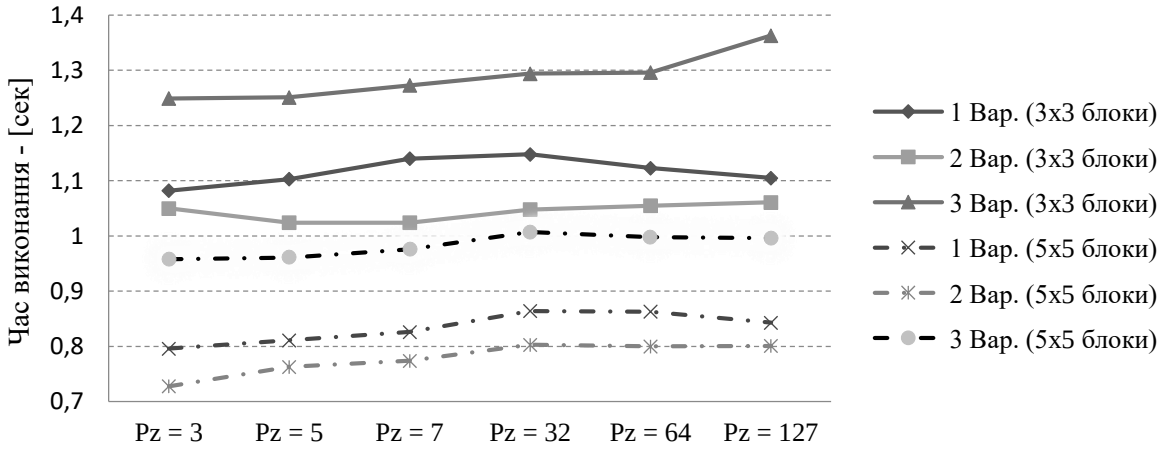


Рис. 10 Час виконання різних варіантів обробки для блоків 3×3 та 5×5 ел.

Висновки

1. Для всіх типів зображень 3-й варіант згладжування надає кращі результати, причому зі зменшенням розмірності матриць згладжування інтенсивність аномалій зменшується.

2. З врахуванням часу обробки гідною альтернативою для 1-го варіанту є 2-й варіант, котрий дає відповідну перевагу при любих розмірностях матриць згладжування (Рис. 8, 10).

3. З зростанням порогу загрубления яскравості елементів (P_Z) кількість і помітність артефактів зростає для всіх типів зображень (Рис. 10). Стрімке зростання викривлень відбувається при виборі значень P_Z більш 7. Припустимою слід вважати величину $P_Z = 14$.

4. В незалежності від встановлених значень P_Z та обраного варіанту згладжування, зростання величини СКП, та як наслідок візуальна помітність викривлень, відбувається в наступній послідовності: портрет – мнемосхема – пейзаж (див. Рис. 9).

5. Для зображень типу «мнемосхема» різниця значень СКП, в залежності від використовуваного типу згладжування, є найбільш помітною (Рис. 8-9), що можна пояснити чутливістю контурів до їх найменших змін і особливостями структури таких зображень.

6. Використання 2 і 3-го варіантів згладжування задовольняє умовам і вимогам (за параметрами часу обробки та ступеня допустимих спотворень) для їх використання на етапі попередньої обробки масивів даних контенту і контейнера.

7. Фактор малої обчислювальної складності процедур етапу попередньої обробки, є принциповим з урахуванням концепту створення мобільних додатків.

8. Збільшення кількості областей зображень з підготовленою, відповідним чином текстурою, створює необхідні стартові умови для ефективної реалізації етапу формування серій подібних блоків. Кількість таких блоків залежить, як від структури зображень, так і заданого параметра їх розмірності.

9. Використання для контенту і контейнеру різних варіантів попередньої обробки та/або різних параметрів їх настроювань (різна розмірність матриць згладжування та різні значення P_Z), вносить потрібну асиметрію в структуру даних, які обробляються, з усіма очевидними корисними наслідками...

10. Виходячи з відомостей про поточні характеристики контенту, можливо адаптивно і оперативно змінювати властивості наявних (в пам'яті мобільного пристрою) контейнерів.

ЛІТЕРАТУРА

1. Грибунин В.Г. Цифровая стеганография / Грибунин В. Г., Оков И. Н., Туринцев И. В. – М.: Солон-Пресс, 2002. – 272 с.
2. Зубарев Ю.Б., Дворкович В.П. Цифровая обработка телевизионных и компьютерных изображений. – М.: МЦНТИ, 1997. – 212 с.
3. Обработка изображений на ЭВМ / Е.А Бутаков, В.И. Островский, И.Л. Фадеев. – М.: Радио и связь, 1987. – 240 с.
4. Прэтт У. Цифровая обработка изображений. т. 1,2. - М.: Мир, 1985. - 736 с.
5. Быков С.Ф. Алгоритм сжатия JPEG с позиции компьютерной стеганографии / Защита информации. Конфидент. – СПб.: 2000, № 3. – С. 26.

ГОНЧАРОВ Микита Олександрович – студент кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: worldxdark@gmail.com; ORCID: 0000-0002-9790-7260.

Наукові інтереси:

– *цифрова обробка зображень, стеганографія.*

МАЛАХОВ Сергій Віталійович – к.т.н., с.н.с.; доцент кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: malakhov@karazin.ua; ORCID: 0000-0001-8826-1616.

Наукові інтереси:

– *інформаційна безпека, захист від несанкціонованих дій, обробка відеоданих.*

УДК 621.391 **СЕКЦІЯ 5**

ГОРБЕНКО І.Д., ЗАМУЛА О.А., ХО ЧІ ЛИК

ОПТИМІЗАЦІЯ ПОШУКУ СКЛАДНИХ СИГНАЛІВ НА ОСНОВІ ОЦІНКИ ЇХ КОРЕЛЯЦІЙНИХ І СТАТИСТИЧНИХ ХАРАКТЕРИСТИК

Вступ

Зусилля дослідників спрямовані на пошук ансамблів складних сигналів, характеристики яких з ростом тривалості наближуються до границі «щільної упаковки» [1-2], тобто ансамблю, усі представники якого мають нульову постійну складову, ідеальні періодичну функцію автокореляції (ПФАК) і періодичну функцію взаємної кореляції (ПФВК), та володіють якомога великим об'ємом. Широко розповсюдженим критерієм подібного наближення є мінімаксний критерій, який орієнтується на синтез ансамблю за мінімізацією максимальних значень бокових піків на множині всіх небажаних кореляцій.

Основні результати досліджень

Оскільки кодове розділення каналів ІКС грунтується на відмінності сигналів, які належать абонентам системи, побудова таких систем визначаються вибором сигналів (дискретних послідовностей) і їх властивостями (насамперед: ансамблевими, структурними, технологічними і кореляційними). Саме тому увага вчених зосереджена на створенні і дослідженні методів та засобів аналізу, синтезу і обробки дискретних сигналів або ДП.

Використовувані в ІКС системи широкосмугових сигналів (ШСС), які застосовуються як фізичний переносник даних, не дозволяють забезпечити необхідні (для тих чи інших додатків ІКС) показники інформаційної безпеки (імітостійкість, інформаційна скритність) і завадозахищеності (енергетична, структурна скритність і завадостійкість прийому).

В процесі функціонування багатокористувачевих систем з кодовим поділом мають місце взаємні завади, які є наслідком одночасної роботи абонентів в загальній смузі частот. Для забезпечення максимально можливої сумісності абонентів повинні бути виконані вимоги до властивостей сигнатур. Іншими словами, при кодовому поділі необхідно так вибрати параметри сигналів, щоб рівень взаємних завад був як завгодно малим, саме таким чином буде забезпечена задана завадостійкість прийому сигналів.

Застосування широкосмугових сигналів (ШСС) дозволяє підвищити завадостійкість прийому сигналів в ІКС при впливі структурних (взаємних), ретрансльованих і організованих завад. При цьому реальна завадостійкість буде нижчою за потенційну. Причинами зниження завадостійкості при входженні в синхронізм і при розрізненні сигналів є наявність бічних піків кореляційних функцій (КФ).

Сформулюємо вимоги до характеристик кореляційних функцій ШСС, виконання яких дозволить досягти необхідних показників завадостійкості (ймовірності помилки) при прийомі сигналів. Для цього необхідно з'ясувати, яким чином характеристики кореляційних функцій сигналів (бокові піки взаїмнокореляційних функцій, їх статистичні характеристики) впливають на завадостійкість прийому даних у багатокористувачевих ІКС. Вирішення цього питання дозволить сформулювати правила вибору систем сигналів для досягнення необхідних значень показників завадостійкості прийому.

Ймовірність помилки при прийомі протилежних сигналів, при умові, що здійснюється прийом даних n-м абонентом може бути визначена як:

$$P_{пом.\,ng} = P(W_{ng} > 1) = \int_{1}^{\infty} \omega_{W_{ng}}(x)\,dx, \qquad (1)$$

де $\omega_{W_{ng}}(x)$ - щільність ймовірності випадкової величини W_{ng}.

При рівноймовірних поєднаннях, ймовірність помилки для n-го абонента при усередненні по всім абонентам дорівнює

© ГОРБЕНКО І.Д., ЗАМУЛА О.А., ХО ЧІ ЛИК, 2021

$$P_{пом.n} = m_1(P_{пом.\,ng}) = 1/C_{L_n}^{v} \cdot \sum_{g=1}^{C_{L_n}^{v}} P_{пом.\,ng} \,. \quad (2)$$

Середня ймовірність помилки є середньоарифметичне усіх $P_{пом.n}$:

$$P_{пом.} = 1/L_n \cdot \sum_{n=1}^{L_n} P_{пом.n} \,. \quad (3)$$

Вираз (3) можна вважати за правило вибору системи сигналів, яке формулюється таким чином: кращою є така система сигналів, яка забезпечує меншу ймовірність помилки.
В [2] показано, що для багатокористувачевих систем, коли для передачі даних використовуються протилежні сигнали ймовірність помилки визначається як

$$P_{пом.ng} = (1 - F(q_{ng})) \cdot \{1 + (\gamma_{ng}/24n) \cdot (q_{ng}^4 \cdot 3 \cdot q_{ng}^2)\}, \quad (4)$$

де: $F(q_{ng})$ – інтеграл ймовірності; $q_{ng} = \frac{1}{\sigma_{ng}} \cdot \sqrt{n}$ є відношенням сигнал/завада, ($n = 2 \cdot g$); γ_{ng} – коефіцієнт ексцесу.

Визначимо правило вибору сигналів для випадку, коли розподіл взаємної завади у багатокористувачевій системі відрізняється від нормального. Будемо вважати, ВКФ сигналів володіють однаковими дисперсіями: $\sigma_{nj}^2 = \sigma^2$. Тоді, виходячи з (4), маємо

$$P_{пом.ng} = (1 - F(q)) \cdot \{1 + (\gamma_{ng}/24n) \cdot (q_{ng}^4 \cdot 3 \cdot q^2)\}, \quad (5)$$

де $\gamma_{ng} = \sum_{j=1}^{g} \gamma_{nj} / g$.

Здійснюючи усереднення за усіма поєднаннями і абонентам, отримаємо

$$P_{пом.} = (1 - F(q)) \cdot \{1 + (\gamma/24n) \cdot (q^4 \cdot 3 \cdot q^2)\}. \quad (6)$$

Коефіцієнт ексцесу системи дорівнює:

$$\gamma = (L_n \cdot (L_n - 1))^{-1} \cdot \sum_{n=1}^{L_n} \sum_{j=1}^{L_n} \gamma_{nj} \,. \quad (7)$$

Згідно з виразів (5) - (6), середня ймовірність помилки тим менша, чим менше коефіцієнт ексцесу. Таким чином як правило вибору систем сигналів слід вважати таке: при рівності дисперсій систем необхідно обирати сигнали, для яких коефіцієнт ексцесу є найменшим.

В [2] показано, що при нормалізації взаємної завади для забезпечення меншого значення ймовірності помилки, необхідно обирати систему сигналів з найменшою дисперсією. Якщо дисперсії сигналів однакові, то краще та система, для якої менше розкид дисперсій ВКФ сигналів.

Саме тому, із урахуванням результатів, які наведені вище, і які стосуються впливу властивостей сигналів (насамперед, статистичних характеристик ВКФ), зупинимося докладніше на деяких системах сигналів, які, саме завдяки притаманним їм статистичним властивостям ВКФ, можуть знайти широке розповсюдження у багатокористувачевих ІКС, до яких висуваються підвищені вимоги щодо забезпечення необхідних показників завадостійкості прийому сигналів, інформаційної, енергетичної і структурної скритності функціонування.

В [3] сформульована і вирішена задача синтезу нелінійних криптографічних дискретних сигналів (КС). Синтез таких сигналів засновано на використанні випадкових або псевдовипадкові процесів, в тому числі, алгоритмів криптографічного перетворення даних. Статистичні властивості нелінійних КС (як показали результати тестування [4]) задовольняють вимогам, що визначені у міжнародних і регіональних стандартах і, таким чином, володіють властивостями непередбачуваності проходження символів, незворотності, випадковості, рівноймовірності, незалежності і ін. Даний клас сигналів, на відміну від відомих лінійних класів сигналів, можуть бути синтезовані для будь-яких значень періоду. в періодичному і аперіодичному режимах передачі інформації (границями «щільної упаковки»).

В [4-5] показана можливість застосування систем КС, як таких, що продукують, для утворення похідних систем сигналів (ПСС). В якості вихідних сигналів такого класу використані ортогональні системи сигналів, а саме, рядки (стовпці) матриці Адамара.

В [6] наведено теоретичні основи синтезу, отримані методи формування і результати досліджень властивостей нелінійних дискретних складних характеристичних сигналів (ХС). Побудова таких сигналів базується на використанні характеру мультиплікативної групи поля $GF(p^n)$. Правила синтезу ХС призводять до сигналів з числом елементів $N = 4x + 2 = p^n - 1$ та $N = 4x = p^n - 1,\ (x = 1, 2, 3, \ldots)$, які володіють дворівневою періодичною функцією автокореляції (ПФАК). При цьому максимальні бокові піки ПФАК таких сигналів не перевищують значень $R_{\mu}(m) = \{-2,\ 2\}$ та $R_{\mu}(m) = \{0,\ -4\}$. Такі значення бокових піків ПФАК свідчать, що у відповідності до границь «щільної упаковки», ХС відносяться до так званих мінімаксних (оптимальних) дискретних послідовностей. Об'єм системи ХС становить $M = \varphi(N)/n$, де $\varphi(N)$ - функція Ейлера. Як КС, так і ХДС, відповідають вимогам, які висуваються до сигналів, що продукують, для утворення похідних систем сигналів, а саме, володіють покращеними автокореляційними властивостями. Крім того, структурні властивості зазначених класів сигналів відповідають вимогам, що висуваються до випадкових послідовностей символів: незворотність, непередбачуваність, випадковість і ін.

На основі багаточисельних обчислень для довжин сигналів 64, 256,1024,2048, були розраховані статистичні і кореляційні характеристики криптографічних та характеристичних дискретних послідовностей, а також похідних сигналів, які утворені на основі зазначених сигналів, а в якості вихідних - застосовані рядки матриці Адамара відповідного порядку. Аналіз отриманих результатів показує, що статистичні характеристики похідних сигналів, що утворені на основі КС, близькі до відповідних характеристик інших, широко застосовуваних сигналів. При цьому значення максимальних бокових викидів функції взаємної кореляції ПСС, утворених на основі криптографічних сигналів, є меншими за значення максимальних бокових викидів лінійних М - послідовностей. Крім того, отримані результати розрахунків показують, що похідні сигнальні системи мають менше значення γ в порівнянні з сигналами, що продукують. У роботі [2] показано, що для ОДС (рядків матриці Адамара порядку N=64) коефіцієнт ексцесу дорівнює 20, тоді як для похідних сигналів, сформованих з використанням ортогональних сигналів і КС $\gamma = 6,3 \times 10^{-2}$, що значно (на порядок) знижує ймовірність помилки при прийомі сигналів.

Висновки

У даній роботі показано, що завадозахищеність ІКС, у суттєвій мірі, залежить від кореляційних і статистичних властивостей ВКФ систем сигналів, що застосовуються. На основі комп'ютерного моделювання і проведених розрахунків показано, що найменше значення ймовірності помилки при вирішенні задачі розрізнення сигналів буде мати місце для так званих похідних сигналів, які утворені на основі рядків матриці Адамара (як вихідних сигналів) і нелінійних дискретних складних характеристичних та криптографічних сигналів (як таких, що продукують). Це обумовлено тими обставинами, що такі системи сигналів володіють покращеними, у порівнянні з ортогональними сигналами, кореляційними і статистичними властивостями ВКФ (насамперед, значення максимальних бокових піків, дисперсія піків та коефіцієнт ексцесу). При цьому похідні сигнали на основі криптографічних послідовностей і рядків матриці Адамара володіють покращеними, у порівнянні з ортогональними і лінійними

класами сигналів, ансамблевими і структурними властивостями. Зазначене дозволить не тільки поліпшити показники завадостійкості прийому сигналів, а й показники інформаційної безпеки і скритності функціонування ІКС.

ЛІТЕРАТУРА

1. Sarvate, D.V. Crosleration Properties of Pseudorandom and Related Sequences. *IEEE Trans. Commun*, 1980. Vol. Com 68. P. 59–90.
2. Варакин Л. Е. Системы связи с шумоподобными сигналами, 1985. 384 с.
3. I.D. Gorbenko, A.A. Zamula Cryptographic signals: requirements, methods of synthesis, properties, application in telecommunication systems // Telecommunications and Radio Engineering Volume 76, 2017. Issue 12, pages 1079-1100. DOI: 10.1615/TelecomRadEng.v76.i12.50.
4. I. D. Gorbenko, A. A. Zamula, Ho Tri Luk Synthesis of derivatives of complex signals based on nonlinear discrete sequences with improved correlation properties// *Радиотехника: Всеукраинский межведомственный научно – технический сборник* – 2019 г. - Вып. 199. – С. 110-120..
5. I. D. Gorbenko, A. A. Zamula, A. E. Semenko, V. L. Morozov Method for complex improvement of characteristics of orthogonal ensembles based on multiplicative combining of signals of different classes. *Telecommunications and Radio Engineering* Volume 76, 2017 Issue 18, pages 1581-1594 . DOI: 10.1615/TelecomRadEng.v76.i18.10.
6. Gorbenko, I., Zamula, A., Morozov, V. Information and communication systems based on signal systems with improved properties building concept systems with improved properties building concept. *CEUR Workshop Proceedings*. 2019, 2353, pages 974–991.

ГОРБЕНКО Іван Дмитрович – д. т. н., професор, професор кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: GorbenkoI@iit.kharkov.ua.

Наукові інтереси: *математичні моделі, методи побудови систем і комплексів захисту інформації.*

ЗАМУЛА Олександр Андрійович – д. т. н., доцент, професор кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: zamylaaa@gmail.com; https://orcid.org/0000-0002-8973-6190.

Наукові інтереси: *математичні моделі, методи побудови систем і комплексів захисту інформації.*

ХО ЧІ ЛИК – магістрант кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: hotriluc97@gmail.com

Наукові інтереси: *математичні моделі, методи побудови систем складних сигналів.*

УДК 004.056.55 **СЕКЦІЯ 5**

ГОРБЕНКО Ю.І., ПОНОМАР В.А., ЄСІНА М.В., ГОРБЕНКО І.Д.

СТАН РОЗРОБКИ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ ДОКАЗОВО СТІЙКИХ ЕЛЕКТРОННИХ ПІДПИСІВ ДЛЯ ПОСТКВАНТОВОГО ПЕРІОДУ

Вступ

Наразі, та очевидно, в деякій перспективі, для криптографічного захисту інформації застосовуються та будуть застосовуватись методи та алгоритми асиметричних криптоперетворень типу електронний підпис (ЕП), що орієнтовані на застосуванні існуючих, стандартизованих криптоперетворень типу ЕП. Вони є суттєвою складовою забезпечення інформаційної безпеки та кібербезпеки в частині забезпечення цілісності, доступності, неспростовності інформації тощо. Але є обґрунтовані підозри, що у постквантовий період існуючі стандарти асиметричних криптоперетворень типу ЕП будуть зламуватись за допомогою квантових криптоаналітичних систем криптоаналітиком 3-го рівня з поліноміальною чи суб'експоненційною складністю [1, 2].

Метою цієї доповіді є аналіз та порівняння проектів національних стандартів ЕП з аналогічними проектами стандартів ЕП NIST США – Dilithium, Falcon та Rainbow [2].

Сутність проблеми розробки стандарту ЕП в Україні

Аналіз показує, що в Україні є розуміння існування загроз стандартизованим існуючим асиметричним криптоперетворенням типу ЕП. Так, на відміну від пропозицій та можливостей, що затверджені та прийняті NIST 8309, є можливість використовувати їх з рівнями безпеки 384 та 512 бітів. Найвищий рівень безпеки проектів Dilithium, Falcon та Rainbow 256 бітів проти класичного та 128 бітів проти квантового криптоаналізу. Перевагою національних рішень є те, що в них закладено вимоги забезпечення 384 і 512 біт безпеки ЕП проти класичного криптоаналізу та 192 і 256 біт безпеки ЕП проти квантового криптоаналізу, які підтверджені та реалізовані в проектах «Вершина» [4]. Крім того, національні проекти в режимі 128 біт класичної та 64 біт квантової стійкості можуть бути на нижніх рівнях безпеки сумісними з міжнародними.

Проект стандарту ЕП «Вершина 1» ґрунтується на підході, що отримав назву "Fiat-Shamir з перериваннями". Важливим результатом цього проекту є обґрунтування та вирішення задачі обчислення (генерування) системних параметрів з рівнями безпеки до 384 та 512 бітів.

Проект стандарту «Вершина 2» у цілому є механізмом ЕП на алгебраїчних решітках з випадковою вибіркою. Вважається, що його безпека ґрунтується на складності проблеми SIS (коротке ціле рішення) над решітками NTRU, а докази безпеки наведені, як у випадковій класичній моделі оракула (ROM), так і у квантовій моделі QROM. Як показують дослідження, ЕП Falcon є складнішими для впровадження, наприклад, ніж Dilithium. Він вимагає використання деревних структур даних, операцій з плаваючою крапкою та вибірки з гаусових розподілів.

Проект стандарту ЕП «Вершина 3» ґрунтується на багатовимірних перетвореннях. Теоретична безпека Rainbow базується на тому, що вирішення набору випадкових багатовимірних квадратичних систем є NP-складною проблемою [2, 4]. Rainbow заявляє стійкість з випадковою «сіллю» та пропонує дуже малі ЕП лише в кілька сотень бітів, які є набагато коротшими, ніж у інших кандидатів конкурсу NIST щодо ЕП.

Методика порівняння національних та міжнародних ЕП

Порівняння ЕП здійснювалось в 3 етапи [3, 4]. На першому етапі спочатку робляться оцінки та перевірки криптопримітиву на відповідність системі часткових безумовних критеріїв і на їх основі обчислюється безумовний інтегральний критерій з використанням якого приймається рішення.

© ГОРБЕНКО Ю.І., ПОНОМАР В.А., ЄСІНА М.В., ГОРБЕНКО І.Д., 2021

На другому етапі отримуються оцінки з використанням часткових умовних критеріїв і на їх основі обчислюється інтегральний умовний критерій. Застосування інтегрального умовного критерію дозволяє прийняти рішення щодо якості ЕП в середньому.

На 3-му етапі, у залежності від вимог, що висуваються до особливо важливих характеристик, здійснюється порівняння за техніко-економічними та техніко-експлуатаційними критеріями. При чому в якості основних, скоріше всього, потрібно використовувати такі критерії як довжини особистих та відкритих ключів, довжини електронних підписів та довжини блоків, складність (швидкодію) основних прямих та зворотних криптоперетворень ЕП, складність генерування ключів та параметрів. Також важливо враховувати їх взаємну залежність, у тому числі і у залежності від показників щодо криптостійкості та розмірів параметрів і ключів, та від математичних методів, що використовуються для реалізації криптопримітивів типу ЕП тощо [3]. Таким чином, важливістю 3-го етапу є те, що на ньому здійснюється перевірка щодо відповідності часткових безумовних та умовних критеріїв вимогам, що висунуті щодо них нормативними документами.

Результати порівняння національних та міжнародних перспективних ЕП

При порівнянні розглядались проекти стандартів «Вершина 1» та «Вершина 2», а також алгоритми ЕП Dilithium, Falcon та Rainbow [1]. Проект ЕП Dilithium, який за попередніми дослідженнями мав кращі результати серед постквантових алгоритмів підпису конкурсу NIST США [1], що засновані на перетвореннях на алгебраїчних решітках. На рис. 1 відображено гістограму загальної відносної переваги алгоритмів ЕП з урахуванням умовного інтегрального критерію при застосуванні вагових коефіцієнтів.

Як видно із рис. 1 найбільшу перевагу має алгоритм «Вершина 1» і ця перевага зберігається для кожного рівня стійкості. Алгоритм «Вершина 2» показує кращі результати, ніж Dilithium (на своїх рівнях стійкості). Що цікаво «Вершина 1» 512 біт стійкості за сукупністю критеріїв програє лише параметрам інших алгоритмів, що гарантують найменший, серед представлених, рівень стійкості.

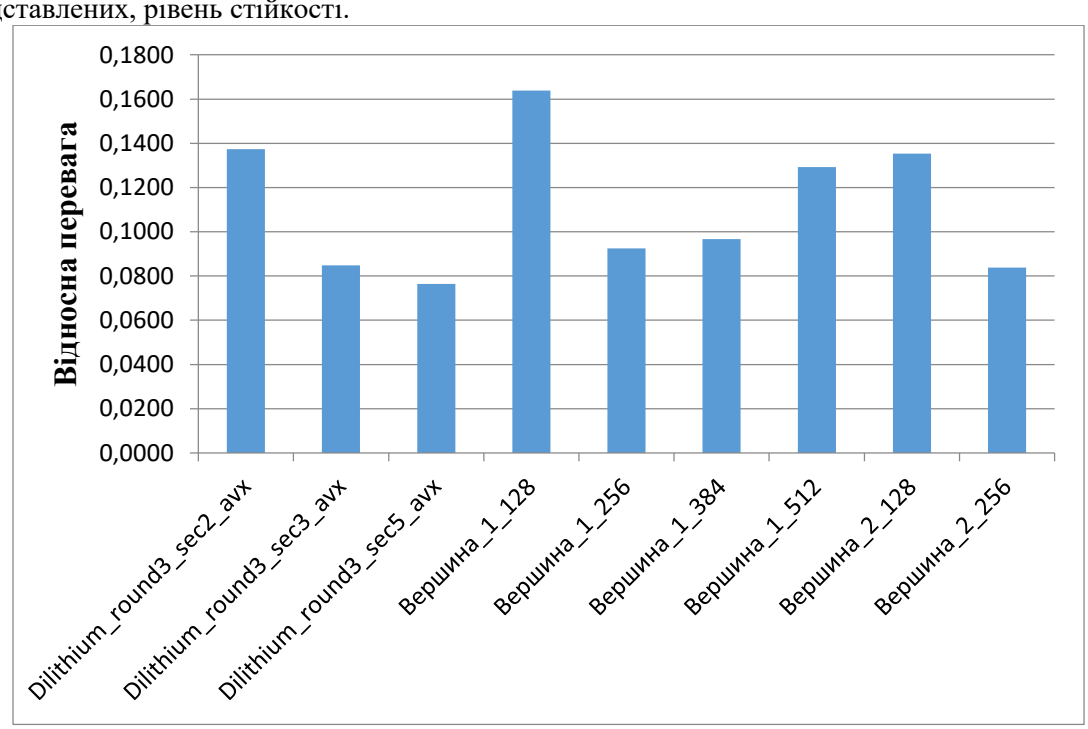


Рис. 1 Переваги алгоритмів ЕП на алгебраїчних решітках

Результати повторного порівняння перспективних механізмів ЕП з використанням методу ранжування

Для порівняння потенційних ЕП методом ранжування варіанти реалізації алгоритмів за їхніми параметрами були розбиті на 2 групи: група параметрів середнього (3-4 рівні) та високого рівня.

На рис. 2 відображено гістограму загальної відносної переваги потенційних алгоритмів ЕП (з параметрами середнього рівня захисту) з урахуванням вагових коефіцієнтів характеристик з використанням методу ранжування. На першому місці алгоритм «Вершина 1», тобто він має кращі показники, а на другому – «Вершина 2».

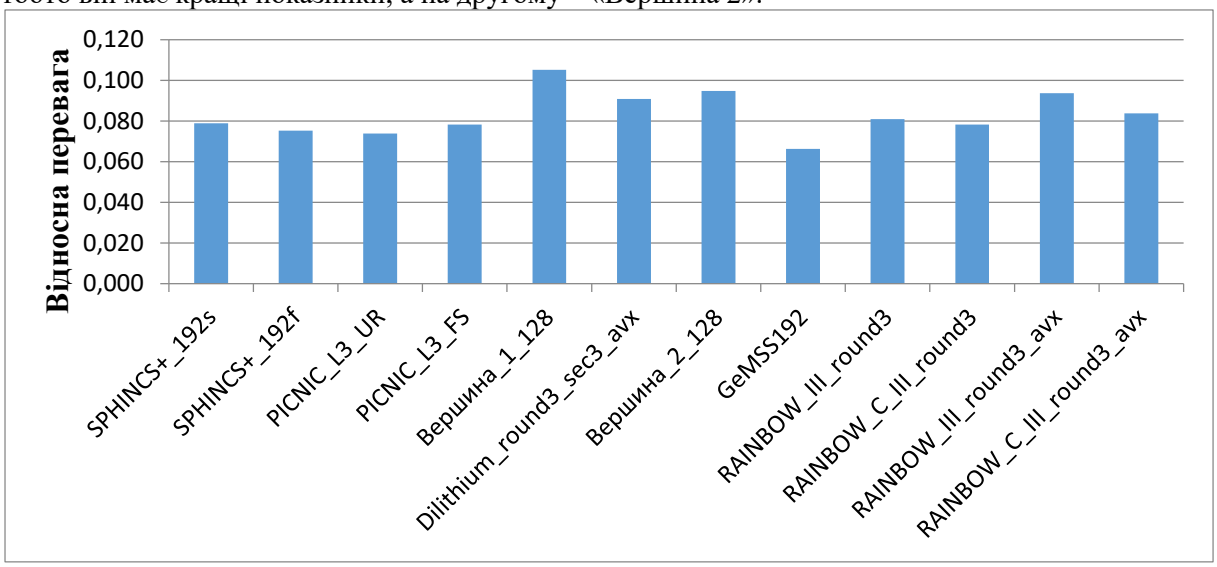


Рис. 2 Переваги алгоритмів ЕП, середнього рівня захисту з використанням методу ранжування

На рис. 3 наведено результати порівняння постквантових алгоритмів ЕП високого рівня захисту стійкості.

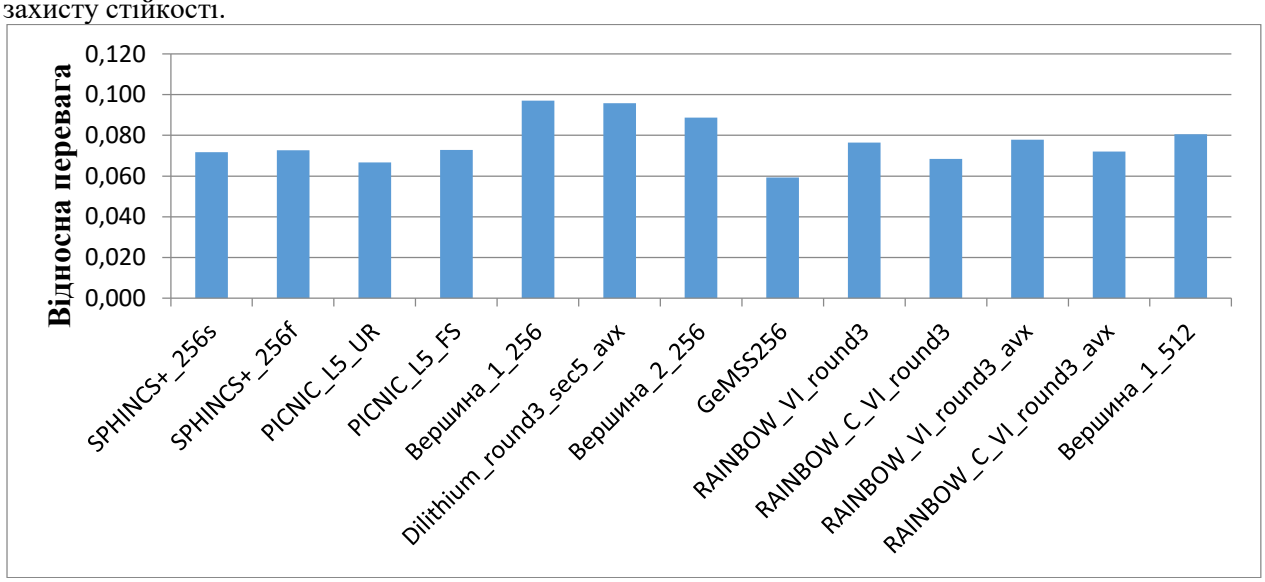


Рис. 3 Переваги алгоритмів ЕП, високого рівня захисту

Кращий результат у алгоритму «Вершина 1», друге місце у алгоритму Dilithium, а третє у «Вершина 2». Це співпадає з результатами попереднього порівняння. При порівнянні методом ранжування з'ясувалося, що кращий результат мають алгоритми, які побудовані на основі перетворень в решеті числового поля.

Висновки

1. Розроблена методика дозволяє порівняти як існуючі, так і перспективні національні та міжнародні проекти стандартів ЕП. Згідно методики порівняння ЕП здійснювалось в 3 етапи. На першому етапі спочатку робляться оцінки та перевірки криптопримітиву ЕП на відповідність системі безумовних критеріїв. На другому етапі отримуються оцінки з використанням часткових умовних критеріїв і на їх основі обчислюється інтегральний умовний критерій. На третьому етапі, у залежності від вимог, що висуваються до особливо важливих характеристик, здійснюється порівняння за техніко-економічними та техніко-експлуатаційними критеріями.

2. Порівнювалися алгоритми, що пройшли до 3-го етапу конкурсу NIST, а також алгоритми проекти національних стандартів «Вершина». При порівнянні для отримання більш точної оцінки, в залежності від вимог до алгоритмів ЕП, використовувалися два методи порівняння. Отримані результати є достатньо близькими. Тому практично кожен з них може застосовуватись для оцінки та порівняння.

3. Кращі показники, тобто переваги, має проект стандарту «Вершина 1», в якості альтернативи можна розглядати алгоритми «Вершина 2» та Rainbow. Крім того «compressed» варіант параметрів Rainbow дозволяє розглядати даний алгоритм в якості повноцінної альтернативи, навіть для систем з обмеженнями до розмірів ключів. Найменші розміри ключів у алгоритмі SPHINCS+_s, але він досить сильно програє за швидкодією.

ЛІТЕРАТУРА

1. PQC Standardization Process: Third Round Candidate Announcement. July 22, 2020. URL: https://csrc.nist.gov/News/2020/pqc-third-round-candidate-announcement Last accessed: 20.11.2020).
2. Gorjan Alagic NISTIR 8309 Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process / Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, John Kelsey, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, Ray Perlner, Angela Robinson, Daniel Smith-Tone //. – URL: https://doi.org/10.6028/NIST.IR.8309 (Last accessed: 20.01.2021).
3. Горбенко І. Д. Порівняння, оцінювання, дослідження можливості використання та переваг постквантових алгоритмів / І. Д. Горбенко, В. А. Пономар, М. В. Єсіна // *Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. – К. : Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського",* 2017. – Випуск 2(34) – С. 9–32.
4. Горбенко І. Д. Генерація загальносистемних параметрів для криптосистеми Falcon для 256, 384, 512 біт безпеки / І. Д. Горбенко, С. О. Кандій, Є. В. Острянська //. *Радіотехніка*. – Вип. 202. – С. 32–52.

ГОРБЕНКО Юрій Іванович – к.т.н., перший заступник головного конструктора АТ «ІІТ», вулиця Бакуліна,12, Харків-22, Україна, 61022; e-mail: GorbenkoU@iit.kharkov.ua.

Наукові інтереси:

– *системи захисту інформації, постквантова криптографія.*

ПОНОМАР Володимир Андрійович – к.т.н., старший науковий співробітник кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: Laedaa@gmail.com.

Наукові інтереси:

– *системи захисту інформації, постквантова криптографія.*

ЄСІНА Марина Віталіївна – к.т.н., доцент кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: m.v.yesina@karazin.ua; ORCID: 0000-0002-1252-7606.

Наукові інтереси:

– *електронний підпис, постквантова криптографія*.

ГОРБЕНКО Іван Дмитрович – д.т.н., професор, професор кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: GorbenkoI@iit.kharkov.ua.

Наукові інтереси:

– *захист інформації електронний підпис, постквантова криптографія*.

УДК 004.855.5 **СЕКЦІЯ 3**

ГУЩІН І.В., СЕРДЮК С.А, СПОРОВ О.Є.

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ЗАСТОСУВАННЯ АРХІТЕКТУРИ НЕЙРОННОЇ МЕРЕЖІ GAN

Вступ

GAN - одне з найбільш багатообіцяючих досягнень в області машинного навчання за останнім часом. Дана архітектура нейронної мережі GAN широко використовується для генерації різноманітних зображень, а також відео, аудіофайлів, текстових даних.

Дана робота розглядає GAN (Generative Adversarial Network) як новий тип нейронної архітектури, а також його застосування у різноманітних задачах, її переваги та недоліки при використанні.

Визначення GAN

GAN (Generative Adversarial Network, укр. Генеративно-змагальна мережа) визначається як алгоритм машинного навчання, що будується на основі застосування комбінації двох моделей нейронних мереж, де одна нейронна мережа виступає у ролі генератора та генерує певні зразки на підставі якогось датасету, а друга нейронна мережа є дескриптором, задача якої полягає у тому, щоб знайти різницю між реальними зразками та тими, що створюються за допомогою генератора [1].

Архітектура GAN

GAN засновані на ідеї змагального навчання. Архітектура GAN в основному складається з двох нейронних мереж, які конкурують один з одним:

- генератор намагається перетворити випадковий шум або певні зображення у такі зображення, які виглядають так, як ніби вони були взяті з датасету;
- дискримінатор намагається передбачити, чи є зображення з датасету або воно було отримано за допомогою генератору.

Генератор та дискримінатор є моделями CNN (Convolution Neural Network).

Тренування GAN

Так, було розглянуто окремо моделі дискримінатора та генератора, тепер можна розглянути їх комбінацію як логістичну модель. Дискримінатор проходить навчання, щоб стати більш досконалим класифікатором, максимізуючи ймовірність присвоєння правильної мітки як до навчальних прикладів (реальним), так і до зображень, створеним генератором (підробка). Генератор же навчається щоб стати кращим фальсифікатором, щоб максимізувати свої шанси обдурити дискримінатор.

Процес навчання двох моделей складається з таких етапів [1]:

- Тренування дискримінатора. До дескриптора надходять реальні дані з навчального датасету та ті, що були згенеровані випадковим чином. Так вчиться дискримінатор.
- Тренування генератора. Модель генератора не може тренуватися самостійно, як модель дискримінатора. Йому потрібна модель дискримінатора щоб краще впоратись із підробкою зображень.

Так отримується логістична модель, що є комбінованою мережею для навчання генератора.

Застосування GAN для синтезу високоякісних зображень на основі текстового опису

Однією із складних задач, що зустрічається у CV (Computer Vision) є синтез високоякісних зображень на основі текстового опису, яка вирішується за допомогою деяких практичних застосувань. Така задача може вирішуватися із використанням StackGAN. GAN першого етапу малює примітивну форму і кольори об'єкта на основі заданого текстового опису,

© ГУЩІН І.В., СЕРДЮК С.А, СПОРОВ О.Є., 2021

у результаті чого виходять зображення з низькою роздільною здатністю. GAN другого етапу приймає результати етапу I і текстові описи в якості вхідних даних і генерує зображення з високою роздільною здатністю з фотореалістичними деталями [2].

Застосування дискримінатора GAN для задачі класифікації

Після навчання отримуються моделі генератора та дискримінатора, де остання може використовуватися для класифікації об'єктів. Таким чином, моделі генератора та дискримінатора навчаються на певному датасеті, де у процесі свого навчання дискримінатор вчиться відрізняти реальні зображення об'єкту.

Застосування GAN для перетворень між зображеннями

Окрім цього, використовуються Pix2Pix GAN, що здійснюють перетворення між зображеннями [1]. Такі GAN можуть застосовуватися для перетворення чорно-білого зображення у кольорове, ескізу - у реалістичне зображення [3] та комбінування зображень, а саме створення зображень у певному стилі, на якому попередньо відбувалось навчання GAN. Для реалізації перетворень між зображеннями використовується генератор після навчання на певному датасеті.

Так, за допомогою GAN можна здійснювати перетворення зображення з низькою роздільною здатністю до зображення з високою роздільною здатністю — із використанням SRGAN (Image super-resolution GAN) [1]. Приклад роботи зображено на рис. 1.

Рис. 1 — Демонстрація роботи SRGAN

Застосування GAN для задачі анімації зображень

Технологія на основі штучного інтелекту дозволяє створювати високоякісні, реалістичні відеокадри, що використовують будь-яке відео як алгоритм для керування рухами, виразами та мовою актора цілі чи об'єкта фотографії. Так, інструмент Deep Nostalgia від myHeritage використовується для анімації осіб на історичних фотографіях із використанням різноманітних драйверів, що складаються з фіксованої послідовності рухів і жестів із подальшим їх застосуванням для завдання рухів [4]. Для роботи із цим інструментом необхідні зображення у високій роздільній здатності, тому інструмент Deep поєднується із інструментом для збільшення роздільної здатності.

Чи створює GAN нове?

Із цього всього виникає питання: «Чи можна за допомогою GAN створити щось нове чи креативне?». Із аналізу теоретичних відомостей щодо GAN можна сказати, що ні. Все, що GAN навчиться робити — це видавати такі результати, які буде складно відрізнити від тих, на яких навчались нейронні мережі. Нічого нового нейронна мережа не вигадає як, наприклад, нової течії у мистецтві, оскільки генератору це не вигідно: тоді дискримінатор скаже, що з

результатом щось не так, він буде відрізнятися від тих даних, на яких здійснювалось навчання. У якому сенсі всі «нові» результати будуть сумішшю того, що використовувалось при навчанні.

Переваги та недоліки GAN

Так, можна виділити переваги GAN [5]:

• Не потребують анотації. Через те, що GAN — алгоритм навчання без учителя, нейронні мережі можуть навчатися без розмічення даних, що зазвичай забирає багато часу.

• Створення зразків, що складно відрізнити від реальних даних. Тепер можна створювати різноманітні зображення, текст, аудіо, відео, що схожі на оригінали. Полегшення роботи графічних дизайнерів та інших робітників у цій сфері.

• Навчений дискримінатор.

Недоліки:

• Обмежені дискримінатором. Не можуть створити щось креативне, що виходить за межі дискримінатора.

• Потребують великої кількості даних для навчання.

Висновки

У даній роботі було розглянуто GAN, їх застосування та різновиди, переваги та недоліки у використанні. Визначено архітектуру моделей генератору та дискримінатора, описано процес тренування та розглянуто питання можливості створення нового та креативного із використанням GAN.

ЛІТЕРАТУРА

1. Mohamed Elgendy Deep Learning for Vision Systems. Manning Publications Co, 2020. С. 12, 341-397.
2. Tao Hu, Chengjiang Long, Leheng Zhang , and Chunxia Xiao.VITAL: A Visual Interpretation on Text with Adversarial Learning for Image Labeling. 2019. С. 1.
3. Nvidia Research Mingyuliu. 2021. URL: http://nvidia-research-mingyuliu.com/gaugan (Дата доступу: 06.04.2021).
4. Deep Nostalgia. 2021. URL: https://www.myheritage.com.ua/deep-nostalgia (Дата доступу: 07.04.2021).
5. Generative Adversarial Networks Projects. Advantages of GANs. 2021. URL: https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781789136678/1/ch01lvl1sec14/advantages-of-gans (Дата доступу: 07.04.2021).

ГУЩІН Іван Валерійович – старший викладач кафедри штучного інтелекту і програмного забезпечення; ХНУ імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: gushchin.iv@gmail.com; ORCID: 0000-0002-1917-716X. Наукові інтереси: розробка програмного забезпечення в області аналізу даних, комп'ютерне моделювання фізичних процесів.

СЕРДЮК Софія Андріївна - студентка магістратури за спеціальністю «Комп'ютерні науки»; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: serdiuk2020ks52@student.karazin.ua; ORCID: 0000-0003-1512-0374. Наукові інтереси: машинне навчання, комп'ютерний зір.

СПОРОВ Олександр Євгенович – кандидат фізико-математичних наук, доцент кафедри штучного інтелекту та програмного забезпечення; ХНУ імені В.Н. Каразіна, майдан Свободи, 6, ауд. 5-20, Тел. 707-50-76, м. Харків, Україна, 61022; e-mail: sporov@karazin.ua; ORCID: 0000-0002-4610-9656. Наукові інтереси: штучний інтелект, розподілені обчислення, операційні системи, комп'ютерне моделювання фізичних процесів.

УДК 004.056.55 **СЕКЦІЯ 5**

ДЕРЕВ'ЯНКО Я.А., ГОРБЕНКО І.Д.

ПОРІВНЯННЯ ШВИДКОДІЇ РЕАЛІЗАЦІЙ АЛГОРИТМУ ЕЛЕКТРОННОГО ПІДПИСУ FALCON НА C ТА PYTHON

Вступ

Загроза з боку квантових комп'ютерів спричинила необхідність до якнайшвидшого дослідження постквантової криптографії. Про це наголошує Національний інститут стандартів і технологій США (NIST) у своєму нещодавньому заклику до стандартизації цього напрямку [1].

Криптографія на основі алгебраїчних решіток виступає в ролі одного з ефективних постквантових рішень. Гомоморфне шифрування та шифрування на основі ідентичності є безсумнівними перевагами використання саме алгебраїчних решіток. Підписи, засновані на решітках, мають високу швидкодію, порівняно невеликі розміри ключів, і, що найголовніше – надають захищеність від квантових атак. Це зумовлено тим, що задачі, які засновуються на припущеннях щодо решітки, мають змогу забезпечити достатній показник захищеності від середніх до найгірших випадків [2].

Алгоритм електронного підпису (ЕП) Falcon є одним з двох підписів на основі решітки серед учасників 3-го раунду стандартизації NIST США [1]. Дане дослідження пропонує порівняльний аналіз швидкодії двох реалізацій цього алгоритму – офіційної реалізацій на мові програмування C та авторської реалізації одного з розробників алгоритму на мові програмування Python.

Схеми ЕП, засновані на решітках та специфікація ЕП Falcon

Основою для припущення щодо безпеки криптографічних примітивів постквантової криптології є такі задачі решітки як: «коротке ціле рішення» (SIS), «задача найкоротшого вектора» (SVP) та «навчання з помилками» (LWE), які обчислювально є достатньо складними. У загальному випадку решітку можна визначити набором векторів, x_i, які формують її базис, *B*, який визначається як [2]:

$$L(B) = L\left(B\right) = \{\sum_i a_i x_i \mid a_i \in \mathbf{Z}, x_i \in B, 1 \le i \le n\}. \quad (1)$$

Для алгоритму Falcon припущення про стійкість генерації ключів базується на задачі NTRU. У ній стверджується, що обчислювально складною є процедура відновлення f і g для елемента кільця поліномів $h = g \cdot f^{-1}$. Стійкість процедури підписання залежить від SVP, в якій стверджується, що при відомому базисі решітки важко знайти короткий вектор у ній.

Всі операції виконуються в поліноміальному кільці $R_q = \mathbf{Z}_q[x]/(x^n + 1)$. N – розмір кільця, q – модуль. Елементи кільця R_q можуть бути представлені як поліноми ступеня N або вектори розмірності N. Параметри N, q, а також рівні безпеки для кожного набору параметрів наведено в таблиці 1.

Табл.1 Пропоновані параметри для схеми підпису Falcon

Набір параметрів	Рівень NIST	Рівень захищеності	Розмірність (*N*)	Модуль (*q*)
Набір 1	Level 1	AES128	512	12289
Набір 2	Level 4 Level5	SHA384 AES256	1024 1024	12289 12289

Основними елементами алгоритму Falcon є поліноми ступеня *n* з цілими коефіцієнтами. Ступінь *n* зазвичай є ступенем двійки (в основному 512 чи 1024) або невеликим числом, яке націло ділиться на ступені двійки (наприклад, 768). Обчислення виконується за модулем

© ДЕРЕВ'ЯНКО Я.А., ГОРБЕНКО І.Д., 2021

нормованого (одиничного) поліному ступеня n, який позначається як ϕ. Можна виразити більшість алгоритму Falcon з точки зору операцій над поліномами, навіть, якщо насправді обробляються матриці, які визначають решітку [3].

Генерація ключів для Falcon складається з наступних етапів: створення поліномів, які будуть заповнювати решітки (на цьому етапі відбувається вибір випадкових поліномів f, g шляхом відповідного розподілу), і побудова дерева Falcon (на цьому етапі вирішується рівняння NTRU для знаходження відповідних F і G). Після генерації ключів, отримуємо NTRU-поліноми, де f і g – особисті ключі, а $h = gf^{-1}$ – відкритий ключ. Процес підписання відбувається шляхом знаходження короткого вектору в решітці NTRU з використанням закритого ключа. Перевірка підпису виконується шляхом перевірки модуля, який розміщується в межах необхідної межі β, за допомогою відкритого ключа [3].

Порівняння швидкодії різних реалізацій на C та Python за однакових умов

Головною метою цієї роботи було порівняння швидкодії різних реалізації алгоритму на різних мовах програмування. Дослідження пропонує порівняльний аналіз швидкодії двох реалізацій алгоритму – офіційної реалізацій на мові програмування C та авторської реалізації одного з розробників алгоритму на мові програмування Python. Автор реалізації на Python – Thomas Prest – описує її як таку, що знаходиться в процесі покращення, та додає, що її не можна вважати безпечною чи придатною для застосування в реальних задачах [4]. Всі тести, наведені в даному розділі, виконувалися за однакових умов, з використанням однакових обчислювальних потужностей. Характеристики персонального комп'ютера, на якому виконувалося тестування наведено в таблиці 2.

Табл.2 Апаратне та програмне забезпечення ПК для тестування

Операційна система	Microsoft Windows 10 Home Single Language
CPU	AMD Ryzen 7 3750H 2.30 – 4.00 GHz (4 ядра / 8 потоків)
RAM	16.00GB DDR4 2400 MHz

Табл.3 Результати проходження тестів для реалізації на C

Test SHAKE256	Done
Test encode/decode	Done
Test verify	Done
Test RNG	Done
Test floating-point	Done
Test polynomials:	Done
Test half-Gaussian sampler	Done
Test sampler	Done
Test sign	Done
Test keygen	Done
Test external API	Done
Test NIST KAT (512)	Done
Test NIST KAT (1024)	Done

Табл.4 Результати проходження тестів для реалізації на Python

Test Sig KATs	Done
Test SamplerZ KATs	Done
Test FFT	Done
Test NTT	Done
Test NTRUGen	Done
Test ffNP	Done
Test Compress	Done
Test Signature	Done

Спочатку проведемо тестування правильності виконання всіх необхідних процедур для кожної з реалізацій. Тестування проводилося, для реалізації на C – у середовищі програмування CLion 2020.2.1, для реалізації на Python – у середовищі PyCharm 2020.3.3. Результати проходження тестів наводяться в таблиці 3 та 4 для C та Python відповідно.

Далі виконаємо заміри часових параметрів для генерації підпису, процесу підписування і перевірки підпису та порівняємо час виконання операцій для різних реалізацій. Порівняння часових показників виконання згаданих вище операцій наведено у таблиці 5.

Табл.5 Часові показники виконання операцій генерації підпису, підписування і перевірки підпису для реалізацій на C та Python

Ступінь (n)	Генерація, мс (для C)	Генерація, мс (для Python)	Підписання, мс (для C)	Підписання, мс (для Python)	Перевірка, мс (для C)	Перевірка, мс (для Python)
256	3,78	1055,78	0,13008	16,71	0,02396	5,56
512	7,44	4074,67	0,23719	35,52	0,05051	8,04
1024	20,77	24136,53	0,45896	74,95	0,09229	18,44

Як можна спостерігати з виконаних замірів часу, реалізація на Python є значно повільнішою за реалізацію на C. Це зумовлено тим, що дана реалізація є відносно новою і потребує значного покращення. Все ж тести показують, що вона працює коректно та здатна виконувати поставлені функції, а отже дана реалізація має всі шанси отримати повноцінний розвиток на рівні з реалізацією на мові C [4].

Для наочності проведених тестувань, наводяться графіки залежності часу виконання процедур в залежності від ступеня *n*. Для можливості відображення значень на одному графіку значення для реалізації C беруться у одиницях вимірювання, що є на 3 порядки нижчими, ніж для Python. Графік для генерації підпису:

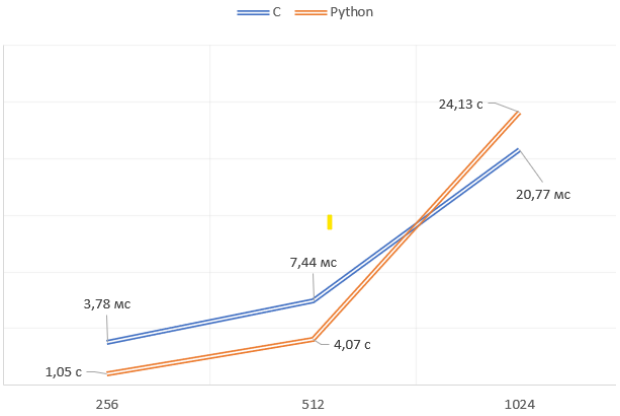


Рис. 1 Залежність часу генерації підпису від ступеня n

Графік для підписання:

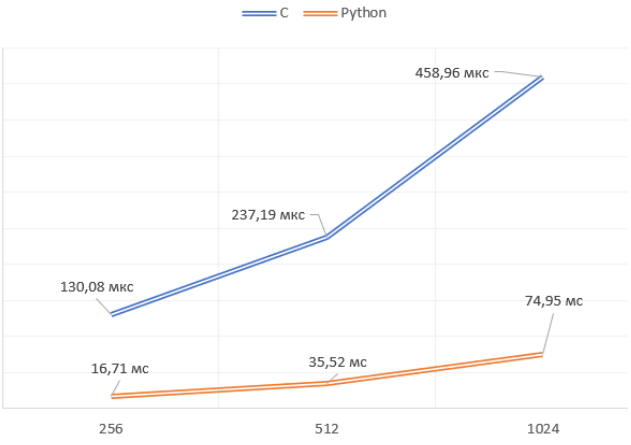


Рис. 2 Залежність часу підписання від ступеня n

Графік для перевірки підпису:

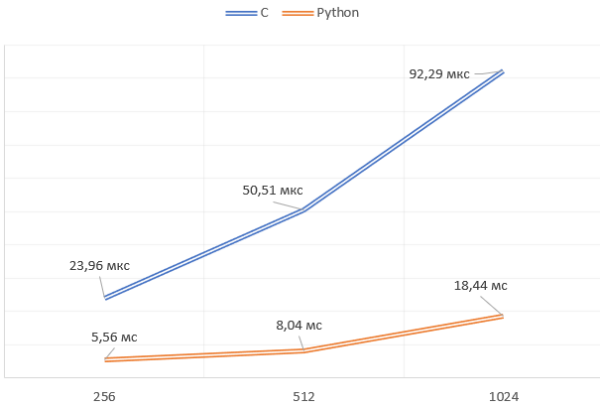


Рис. 3 Залежність часу перевірки підпису від ступеня n

Висновки

У даній роботі розглянуто один з перспективних алгоритмів електронного підпису, який є учасником 3-го раунду стандартизації NIST США. Можна з упевненістю сказати, що надійною математичною основою для алгоритмів ЕП, які можуть бути стійкими у постквантовий період, можуть бути алгебраїчні решітки. В реальних задачах алгоритм добре себе показав, швидко та без помилок виконуючи всі необхідні операції. Як можна спостерігати з проведених досліджень реалізація на Python є значно повільнішою за реалізацію на C. Це пояснюється тим, що дана реалізація не є фінальною та можливо буде покращуватися і отримає повноцінний розвиток на рівні з реалізацією на мові C.

ЛІТЕРАТУРА

1. Post-Quantum Cryptography. Round 3 Submissions, 2020. URL: https://csrc.nist.gov/Projects/post-quantum-cryptography/round-3-submissions.
2. Горбенко І.Д., Горбенко Ю.І. Прикладна криптологія: підручник. Харків: ХНУРЕ, 2012. 884 с.
3. Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Prest, Thomas Ricosset, Gregor Seiler, William Whyte, Zhenfei Zhang. Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU Specifications v1.2, 2020. 67 p. URL: https://falcon-sign.info/falcon.pdf.
4. Офіційний сайт ЕП Falcon. URL: https://falcon-sign.info.

ДЕРЕВ'ЯНКО Ярослав Андрійович – студент кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 6, Харків, Україна, 61000; e-mail: yarik0009258@gmail.com; ORCID: 0000-0002-3290-3373.

Наукові інтереси:

– *електронні підписи, постквантова криптографія, математика алгебраїчних решіток, постквантові алгоритми.*

ГОРБЕНКО Іван Дмитрович – доктор технічних наук, професор, головний конструктор АТ «Інститут інформаційних технологій», професор кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 6, Харків, Україна, 61000; e-mail: GorbenkoI@iit.kharkov.ua.

Наукові інтереси:

– *електронні підписи, постквантова криптографія, математика алгебраїчних решіток, постквантові алгоритми.*

УДК 535.4 + 621.373 **СЕКЦІЯ 2**

ДОЛЯ Г.М., БОНДАРЕНКО К. О.

ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЛАЗЕРНОГО ДЕТЕКТУВАННЯ АКУСТИЧНИХ КОЛИВАНЬ У ПОВІТРІ З ВИКОРИСТАННЯМ СВІТЛОПОВЕРТАЮЧИХ ПОВЕРХОНЬ

Вступ

Останнім часом для практичних застосувань з'явився новий клас світлоповертаючих поверхонь (СПП), що складаються з сукупності мініатюрних світлоповертачів. Істотною їх відмінністю від шорстких поверхонь або розсіюваючих частинок в потоці є закономірності просторового розподілу розсіяного випромінювання. З урахуванням цих відмінностей може бути побудований цілий ряд нових ефективних пристроїв, таких, наприклад, як датчики вібрацій [1] або акустичних коливань [2], а також вимірювачі лінійної або кутової швидкості руху [3, 4], тобто лазерні велосіметри. Серед інших, є також акусто-оптичні датчики – вони активно експлуатуються і мають потенціал для застосування в різних галузях промисловості, а використання ретрорефлекторів на додаток до них дозволяє створювати нові перспективні типи детекторів [5, 6], дослідження систем лазерного детектування акустичних коливань з використанням світлоповертаючих поверхонь є практично цінним і перспективним.

Раніше в роботах [2, 7] було проведено дослідження, яке показує можливість створення таких датчиків з використанням кутових відбивачів. Актуальним завданням є дослідження особливостей систем детектування акустичних коливань складаються з лазерного датчика і світлоповертаючої поверхні зі склокульок. На відміну від регулярної просторової структури решітки з ретрорефлекторів – кутових відбивачів, розташування мікрокульок на поверхні і їх розмір є випадковими, а значить для вирішення такого завдання доцільно розробити імітаційну модель системи та застосувати її для аналізу впливу варіації параметрів системи на якість детектування.

Опис фізичних принципів покладених в основу методу

Метод детектування звуку в повітрі шляхом зондування середовища його поширення лазерним променем і аналізу характеристик відбитого після цього від СПП лазерного випромінювання вперше був описаний в роботі [7]. В основі принципу детектування акустичних коливань в повітрі за допомогою лазера лежить факт модуляції оптичної щільності повітря при поширенні акустичної хвилі. По суті, така модуляція нерозривно пов'язана з природою механічних деформацій речовини в ході акустичного коливання, а тому несе в собі інформацію про властивості звукового сигналу. Це дозволяє використовувати лазерний пучок для детектування коливань: в загальному випадку, амплітудно-фазовий розподіл (АФР) його поля змінюється при проходженні через оптично неоднорідне середовище акустичного стовпа, а значить, спостерігаючи за зміною АФР світлового поля в часі, можна відновити акустичний сигнал, який його промодулював.

Базуючись на такому принципі детектування, для створення реального датчика необхідно вибрати спосіб реєстрації зміненого оптичного сигналу. З цією метою в оптичну систему додається ретрорефлектор на основі мікросклошариків. Розташована по іншу сторону акустичного стовпа від лазера, світлоповертаюча поверхня відбиває потужність оптичного сигналу назад у бік випромінювача, що дозволяє просторово поєднати фотоприймач з випромінювачем, роблячи систему детектування значно більш компактною.. Після відбиття від СПП світло проходить назад через звукову хвилю і його потужність інтегрується за допомогою лінзи на фотодетекторі. Закономірності зміни фотоструму за рахунок зміни діаграми розсіяного випромінювання в площині прийому і є відображення акустичних коливань.

Запропонована схема детектування корисна тим, що не потребує механічної взаємодії датчика з середовищем поширення акустичних коливань - детектор можна розташувати поза цим середовищем. Стає можливим уникнути спотворення акустичних властивостей спостережуємої системи впровадженням у неї датчика, якщо вважати прогрів та іонізацію

© ДОЛЯ Г.М., БОНДАРЕНКО К. О., 2021

повітря лазерним випромінюванням незначними. Також відпадає необхідність мати розташовувати датчик близько до джерела акустичних коливань - потужність звуку спостерігається опосередковано, а значить ступінь віддаленості детектора обмежується не потужністю предмета спостереження, а властивостями оптичної системи.

У зв'язку з цим інтерес представляє дослідження взаємозв'язку параметрів в системі, а також вплив зміни параметрів СПП на якість детектування. Для цього доцільно розробити імітаційну модель і дослідити її поведінку при різних умовах детектування.

Теоретичні основи процесів взаємодії лазерного випромінювання з акустичною хвилею

У середовищах з нелінійним показником заломлення промінь світла викривляється пропорційно градієнту показника заломлення. На цьому ефекті засновані системи сканування, в яких за допомогою електричних полів або пружних напружень створюється змінний в часі градієнт. Якщо промінь світла поширюється в середовищі в напрямку, перпендикулярному градієнту показника заломлення, то він відхиляється в бік зростання показника заломлення. У даній роботі передбачається, що виконується умова малості ширини лазерного пучка щодо довжини акустичної хвилі в повітрі - умова дифракції не виконується. Це означає, що звуковий стовп можна уявити як акустооптичний рефракційний дефлектор - механічне поширення звукової хвилі супроводжується зміною щільності повітря, що створює градієнт оптичної щільності і викликає відхилення променя від осі його первинного поширення. Тоді кут відхилення змінюється за синусоїдальним законом [8]. Обмеженням такого наближення є те, що пучок повинен падати на площину перпендикулярно, інакше, проходячи під великим кутом або малим кутом, але значній товщині акустичного стовпа, викривлений світловий промінь потрапляє в область звукової хвилі, де градієнт деформації змінює знак, і починається відхилення променя в протилежну сторону.

Аналіз результатів моделювання

Важливою характеристикою детектора акустичних коливань є амплітуда корисного сигналу (струму) на виході фотоприймача, яку зручно аналізувати в спектральної області в разі детектування монохроматичного звукового сигналу. При цьому було відмічено, що в деяких ситуаціях крім основної частоти спостерігається друга і більш високі гармоніки. Внесок гармонік в реєстрований сигнал можна оцінювати за допомогою коефіцієнта гармонік

$$k_f = \frac{\sqrt{u_2^2 + u_3^2 + ... + u_n^2}}{\sqrt{u_1^2 + u_2^2 + u_3^2 + ... + u_n^2}}, \quad (1)$$

де u_i – діюче значення i - ї гармоніки сигналу [9]. Очевидно, що чим більше величина коефіцієнта, тим гірше, тому що більше вплив гармонік на якість детектування монохроматичного сигналу.

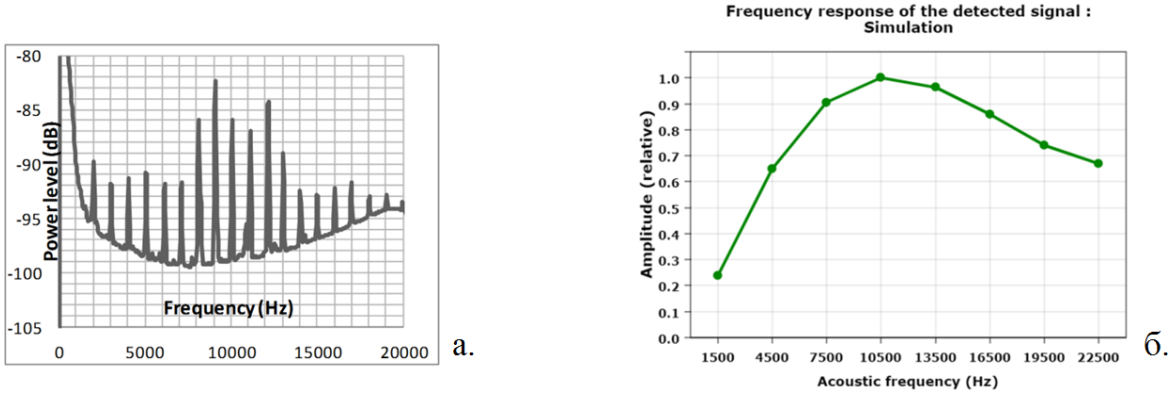


Рис. 1 АЧХ детектуємого сигналу: а) фізичний експеримент; б) комп'ютерне моделювання

Практичний інтерес представляє також залежність величини фотоструму від частоти звукових коливань (АЧХ). Лінійність такої характеристики означала б передачу без спотворень складного звукового сигналу, що складається з набору частот. Вид такої характеристики, виміряної експериментально шляхом аналізу фотоструму при послідовній зміні частоти звукового сигналу, наведено на рисунку 1 (а). З нього видно, що характеристика має максимум в області частот 9кГц і убуває в області більш низьких і більш високих частот.

Аналогічна залежність досліджувалася шляхом імітаційного моделювання при значеннях параметрів, відповідних умовам фізичного експерименту, результати наведені на рисунку 1 (б). Звернемо увагу на схожість форми кривих АЧХ, що свідчить про адекватність імітаційної моделі реальним фізичним процесам, що відбуваються при детектуванні.

Пояснити вигляд АЧХ на малюнках 1 (а) і 1 (б) можна наступним чином. За інших рівних, зростання частоти звукової хвилі спричиняє збільшення відхилення світлового пучка. При малому куті і невеликій відстані від акустичного стовпа до СПП пляма пучка періодично відхиляється від осі по поверхні СПП незначно - на величину порівнянну з діаметром самого пучка. Зовсім малі відхилення створюють малі флуктуації детектуємої потужності.

При збільшенні частоти збільшується амплітуда зміщення пучка по поверхні. При фіксованій частоті дискретизації апроксимація кривої зміни фотоструму стає все більш грубою, що адекватно збільшенню потужності гармонік основної частоти і, отже зменшення вкладу коливань на основній частоті в змінну складову фотоструму.

Далі звернемо увагу на результати імітаційного моделювання впливу діаметра пучка, що потрапляє на СПП (рисунок 2), при фіксованому середньому розмірі кульки. Тут по осі абсцис відкладено відношення (у відсотках) півширини гауссового пучка до маточікування діаметра кульки, а по осі ординат - коефіцієнт гармонік для спектральної складової фотоструму на частоті звукових коливань.

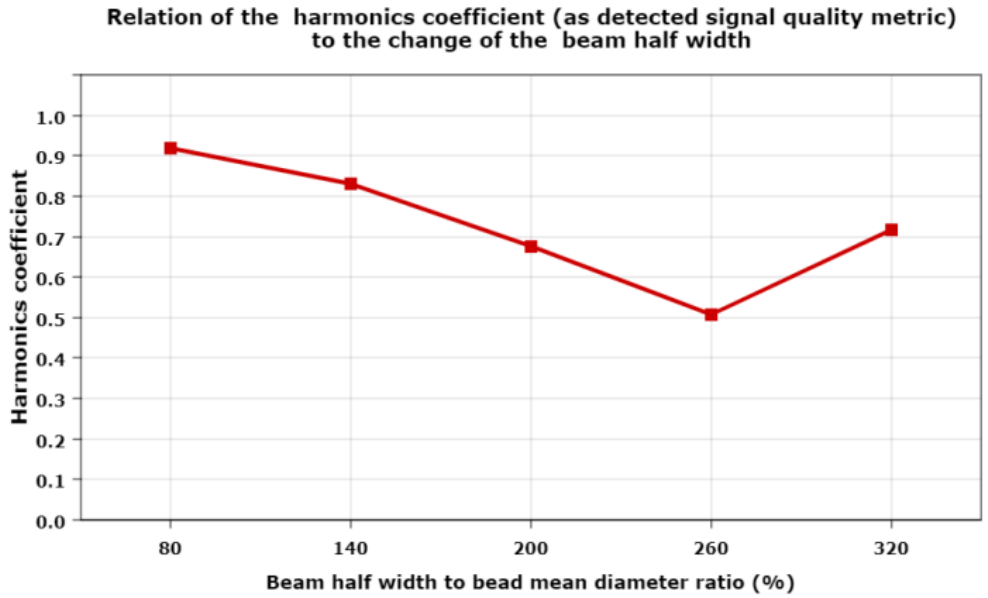


Рис. 2 Зміна коефіцієнта гармонік при збільшенні півширини пучка

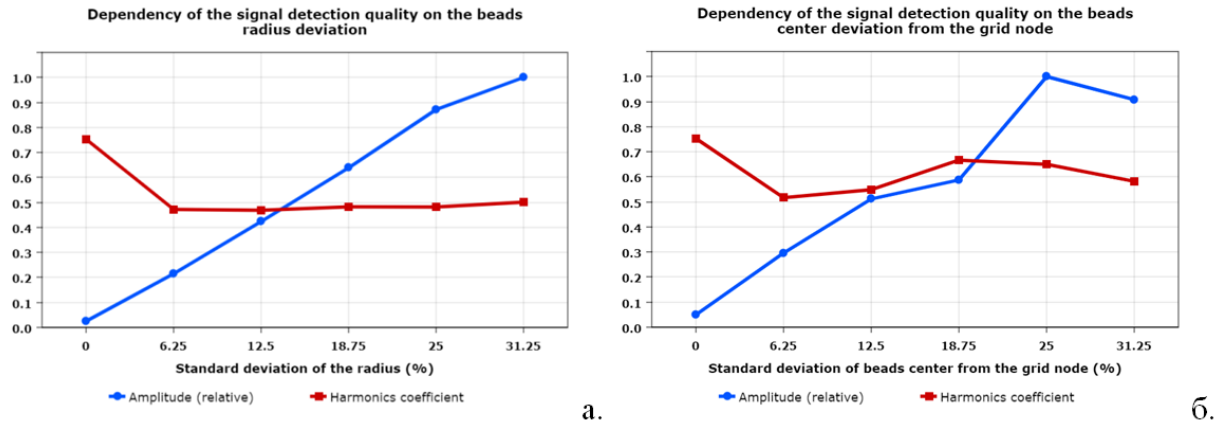
З графіка очевидно, що відношення діаметра пучка до діаметру кульки має важливе значення: можливо зменшити, а значить поліпшити, величину коефіцієнта гармонік, так як при його збільшенні відбувається згладжування спостережуємої у фотострумі синусоїди, що відповідає частоті розгортки акустичного дефлектора звукової хвилі - за рахунок засвітчення більшої кількості мікровідбивачів.

Звернемо увагу, що в разі, коли діаметр пучка можна порівняти з розміром кульки, буде створюватися ситуація, при якій пучок спочатку повністю накриває кульку, а потім - коли пучок теж не накриває взагалі ні однієї кульки. У цьому випадку крім періодичності процесу,

обумовленої звуковими коливаннями, в спектрі вихідного сигналу будуть виникати додаткові частоти, обумовлені (квазі)періодичним характером розташування кульок на СПП.

Якщо ж діаметр пучка буде занадто великий щодо діаметра однієї кульки, то змінна складова в детектуємій світловій потужності стає малою в порівнянні з середнім її значенням. Середнє значення не буде змінюватися істотно, тому що кількість засвічених кульок велика і при переміщенні пучка значно не змінюється.

На рисунку 3 (а) представлені результати імітаційного моделювання впливу середнього квадратичного відхилення (СКВ) розміру кульки при фіксованому його середньому розмірі. Тут по осі абсцис відкладені значення відносного СКВ в процентах до середнього розміру кульки, а по осі ординат - потужність корисного сигналу (синя лінія) і величина коефіцієнта гармонік (червона лінія).



а. б.

Рис. 3 Залежність якості детектування сигналу від: а) СКВ радіусу МК; б) СКВ відхилення центру МК від вузла решітки

З отриманої залежності видно, що випадковий розмір кульки більш бажаний, ніж детермінований. Це обумовлено тим, що при детермінованому розмірі кульок змінна складова, що несе інформацію про звукові коливання, відносно мала в порівнянні із середнім значенням детектуємої світлової потужності.

На рисунку 3 (б) представлені результати імітаційного моделювання впливу нерегулярності розташування кульок на поверхні. Тут по осі абсцис відкладені значення СКВ відносного розташування кульок при їх зміщенні від вузлів ідеальної квадратної решітки.

По осі ординат - потужність корисного сигналу (синя лінія) і величина коефіцієнта гармонік (червона лінія). З отриманих результатів видно, що використання ідеальної решітки менш доцільно ніж рандомізованної до певної величини. Причини такого характеру зміни корисної складової фотоструму аналогічні розглянутим вище при аналізі впливу рандомізації розміру кульки.

Підсумок

В ході аналізу результатів експериментів були виявлені залежності між параметрами системи, сформульовані рекомендації по їх підбору для підвищення якості детектування. Показано, що для практичних застосувань доцільно витримувати раціональне співвідношення діаметра світлового пучка і діаметра кульки/періоду їх розташування на поверхні. Показано, що рандомізація розмірів кульки і просторового періоду їх розташування є бажаною для збільшення потужності детектируемого сигналу. Отримані результати можуть бути використані при побудові конкретних акустооптичних детекторів для вирішення прикладних завдань, допомагаючи підібрати відповідні параметри СПП. Розвинути роботу в подальшому можливо в напрямку застосування створеної моделі до детектування комплексного (мультичастотного, складеного) акустичного сигналу, а також проведення обчисленоємних досліджень для визначення раціональних параметрів детектора акустооптичних коливань.

ЛІТЕРАТУРА

1. Dolya G. N., Zhyvchuk V. The appreciation of the influence of exactness of focusing on the work of the laser homodyne method of measuring the parameters of vibration // Proc. SPIE Vol. 5582, 2004. 45-52 c.
2. Dolya G.N., Litvinova E. S. Retroreflective laser detector of acoustic oscillations // Proceedings of 12th International Conference on Laser & Fiber-Optical Networks Modeling (LFNM'2016), Sept. 15-18, Odessa, Ukraine. 108-111 c.
3. Dolya G.N., Katunin A.N., Nad O.A., Bulay A.N. Laser speckle velocimetry for running objects with light reflective surface Systems of information processing // Kharkiv: KhUAF, Issue 1(126), 2015. 23-26 c.
4. Dolya G. M., Lytvynova O. S. Modeling the Method of Laser Doppler Speckle-Velocimetry for Flat Objects with Retroreflective Surface // Asian Engineering Review Vol.4, No. 2, 2017. 7-13 c.
5. Korpel, A. Acousto-Optics, Second Edition. Technology & Engineering, CRC Press, 1996. 360 c.
6. Magdich, L. N., Molchanov V. Ya. Acoustooptic Devices and Their Applications. Technology & Engineering, CRC Press, 1989. 160 c.
7. Dolya G. M., Lytvynova O. S. Modeling of fluctuations of laser radiation scattered on the reflector array in a turbulent atmosphere // Laser and Fiber-Optical Networks Modeling (LFNM), 2011. 108-111 c.
8. Ребрин Ю. К. Управление оптическим лучом в пространстве. М.:Сов. Радио, 1977. 336 с.
9. Шкаликов В.С. Измерение параметров вибрации и удара. – М.: Издательство стандартов, 1980. 280с.

ДОЛЯ Григорій Миколайович – д. т. н., професор; професор кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: gdolya@karazin.ua; ORCID: 0000-0001-8888-5189

Наукові інтереси:

– оптоінформатика, лазерна локація, спеклометрія

БОНДАРЕНКО Костянтин Олексійович - e-mail: konstantinb2016@gmail.com; ORCID: 0000-0002-2430-0716

Наукові інтереси:

–комп'ютерне моделювання оптичних процесів

УДК 004.052:681.3 **СЕКЦІЯ 2**

ДОРОШЕНКО М.І., БУЄВИЧ-СИСОЄВ В.М.

МОДЕЛЬ ОЦІНКИ ПАРАМЕТРІВ КОМП'ЮТЕРНОЇ МЕРЕЖІ

Вступ

У зв'язку з стрімким розвитком комп'ютерних технологій набуває актуальності тема оцінки параметрів і станів комп'ютерних мереж (КМ). Це, в першу чергу, необхідно для ефективної побудови КМ, а також для запобігання помилок, які можуть виникнути при функціонуванні мережі. Одним з основних показників при оцінюванні стану КМ є трафік, який дозволяє оцінити працездатність, простежити й оцінити характеристики працездатності мережі. А так, як на разі проходе перехід до мереж нового покоління виникає необхідність в удосконалені старих та розробці нових методів управління інформаційно-обчислювальной мережей. Однією з компонент системи управління є підсистема моніторингу. Проведений аналіз методів та технологій управління показав, що для більш повного аналізу мережі в такі системи необхідно включати додаткові модулі обробки статистичної інформації. Тому дослідження структури трафіку, оцінювання його основних характеристик є актуальним завданням сучасної науки при аналізі станів комп'ютерних мереж.

Вибір параметрів для оцінювання

Трафік є одним з основних показників роботи комп'ютерної мережі. Для оцінки КМ необхідно оцінити параметри трафіку, що передається в цій мережі, кожен з яких впливає на стан системи в цілому. В ході роботи були оцінені наступні параметри трафіку: джиттер, швидкість передачі інформації, затримка передачі пакета по мережі і втрати пакетів. Даний набір параметрів часто використовують при оцінці якості обслуговування мережі (QoS) [1], що дозволяє описати стан КС.

Підготовка та збір даних

Для оцінки необхідних параметрів було побудовано мережу в додатку GNS3, який призначений для побудови реальних імітаційних моделей КМ з емуляцією кожного елемента мережі (VMWare, Qemu, Dynamips) [2]. Альтернативним варіантом є додаток Packet Tracer, однак, він не дозволяє інтегрувати інші ПО в своє середовище. Топологія досліджуваної мережі містить 45 елементів (Рис 1.).

Така модель КМ може управляти трафіком, має можливість масштабуватись, а також проводити моніторинг трафіку мережі з використанням додаткових модулів.

Для генерації трафіку було використано безкоштовне ПО D-ITG, яке є на більшості Linux OS. Даний генератор трафіку дозволяє вибирати специфічні протоколи передачі даних, визначати розмір даних, швидкість передачі даних, а також має можливість збирати дані про переданий трафік в лог-файли для збору показників параметрів [3].

На розробленому експериментальному стенді було проведено по 100 автоматизованих вимірів параметрів трафіку на кожен з трьох станів КМ. Увесь трафік передавався по транспортному протоколу UDP, що дало можливість отримати важливий параметр - втрати пакетів при передачі потоку даних [4]. В ході експерименту були розглянуті наступні стани мережі: при малому навантаженні, при середньому навантаженні та при високому навантаженні.

При малому навантаженні мережі було вирішено використовувати швидкість передачі даних в 300 пакетів/сек, для середньої - 3000 пакетів/сек, а для високої - 30000 пакетів/сек. Об'єм кожного пакету мережі становив 512 байт та використовувався 1 потік передачі даних. Заміри проводилися за допомогою команди ITGRecv, яка дозволяє переглядати лог-файли, що містять інформацію про трафік мережі [5]. Таким чином було отримано вибірки параметрів для кожного стану системи, що надалі дозволило провести оцінку даних параметрів.

© ДОРОШЕНКО М.І., БУЄВИЧ-СИСОЄВ В.М., 2021

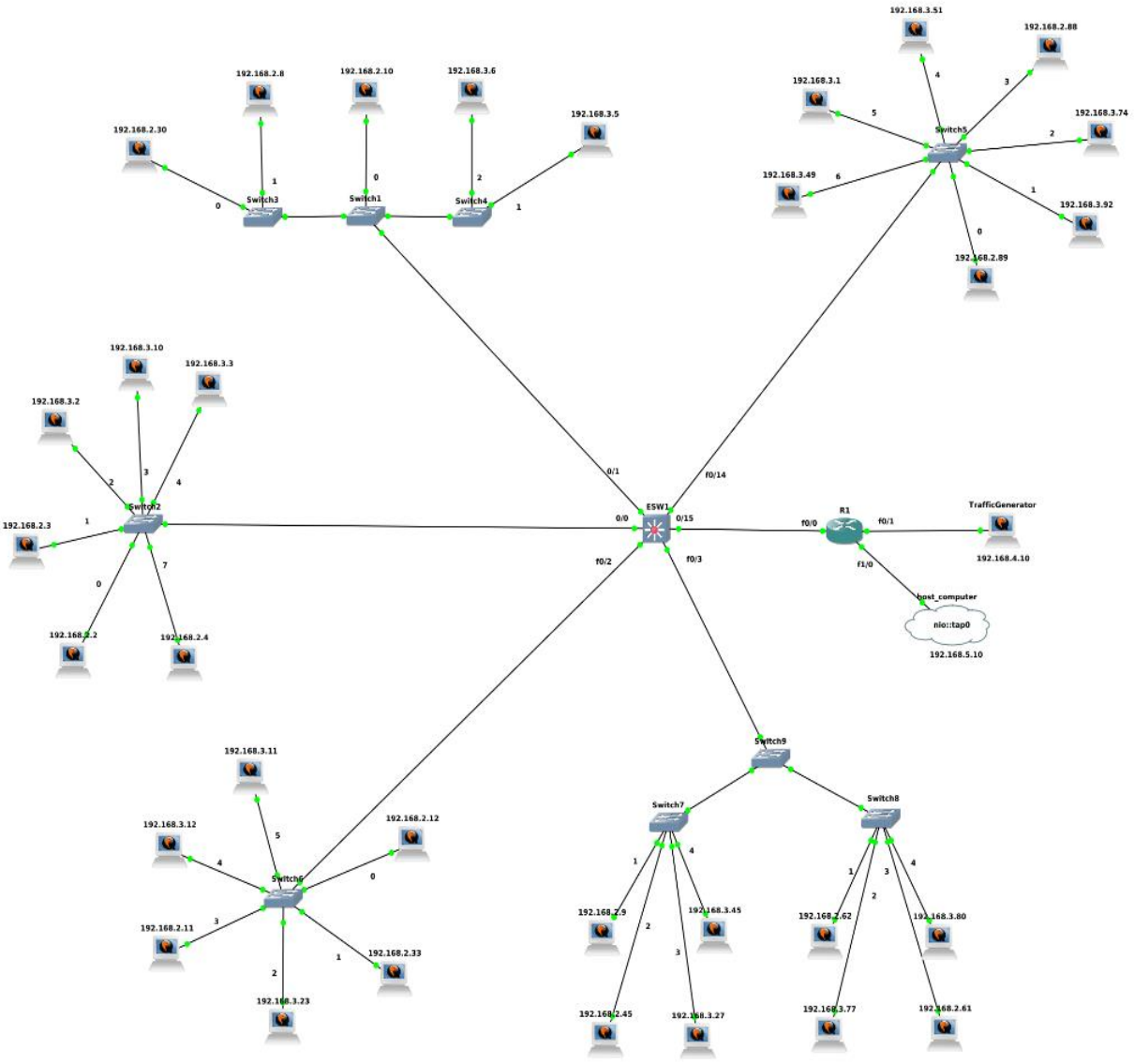


Рис. 1. Топологія досліджуваної КМ

Метод оцінки параметрів

Для оцінки стану мережі був сформований інтервал значень для кожного параметра, який характеризує відповідний параметр при кожному стані мережі. Інтервал містить значення мінімуму та максимуму показника параметра, який був експериментально отриманий. Результати дослідження зазначені в таблиці 1.

Табл.1 Оцінка параметрів трафіку КМ

Навантаженість мережі	Джиттер (сек)	Затримка передачі пакета (сек)	Швидкість передачи даних	Втрати пакетів (%)
300 пакетів/сек	[0; 0.001049]	[0; 0.528]	[0; 6232.59]	[0; 3.99]
3000 пакетів/сек	[0.00105; 0.001095]	[0	[6232.6; 6314.09]	[4; 28.33]
30000 пакетів/сек	[0.001096; 0.001141]	[0.541; 0.5492]	[6314.1; 6480.65]	[28.34; 100]

Таким чином, запропонований метод оцінки дозволяє отримати максимально наближені можливі показники параметрів для відповідного стану КМ й оцінити завантаженість мережі,

проводячи аналіз мережевого трафіку і зіставляючи отримані дані з наявними показниками параметрів для кожного рівня завантаження мережевого каналу.

Висновки

В рамках даного матеріалу були розглянуті основні параметри трафіку і запропоновано метод оцінки параметрів трафіку КС. Проведено дослідження роботи мережі при різних навантаженнях, в результаті чого було отримано значення основних показників мережевого трафіку. Використання даного методу дозволяє оцінити продуктивність мережі в цілому, а також запобігати збоям в системі, що у свою чергу збільшить ефективність КМ. Описана модель також може використовуватись для класифікації станів КМ.

ЛІТЕРАТУРА

1. Вегешна Шринивас. Качество обслуживания в сетях IP. :Пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 368 с.
2. Графічний симулятор мережі gns3 [Електронний ресурс] - https://pcuawiki.ru/rizne/9170-grafichnij-simuljator-merezhi-gns3.html
3. Avallone S., Guadagno S., Emma D. [and other] D-ITG distributed Internet traffic generator // First International Conference on the Quantitative Evaluation of Systems, QEST 2004, 27 – 30 Sept. 2004: proceedings of the conference. – P. 316 – 317.
4. Sawashima, H., Hori, Y., Sunahara, H., Oie, Y.: Characteristics of UDP packet loss: effect of TCP traffic. In: Proc. INET 1997 (June 1997)
5. D-ITG 2.8.1 Manual [Електронний ресурс] - http://traffic.comics.unina.it/software/ITG/manual/index.html

ДОРОШЕНКО Максим Ігорович – студент групи КІ-41 факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, 61022; e-mail: xa11867769@student.karazin.ua; ORCID: 0000-0003-3645-252X.

Наукові інтереси:

– *Програмування, структури даних.*

БУЄВИЧ-СИСОЄВ Владислав Миколайович – ст. викладач; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, 61022; e-mail: Vladbuevich1993@gmail.com; ORCID: 0000-0002-4195-0526.

Наукові інтереси:

– *Програмування, структури даних.*

УДК 004.056.55 **СЕКЦІЯ 5**

Д'ЯЧЕНКО А.С., ГОРБЕНКО І.Д.

ДОСЛІДЖЕННЯ МАТЕМАТИЧНОЇ СТРУКТУРИ ТА ПРОЦЕСУ ГЕНЕРАЦІЇ КЛЮЧОВОЇ ПАРИ У АЛГОРИТМІ ЕП RAINBOW

Вступ

Створення криптографічних систем, які могли б забезпечити безпечність у постквантовий період - одна з найважливіших цілей сучасної криптографії. Упродовж останніх років спостерігається значний розвиток у напрямку створення квантових комп'ютерів і однією з найбільших загроз є те, що обчислювальна потужність даних комп'ютерів ставить під загрозу усі досягнення у криптографії минулих років, завдяки більшим можливостям для застосування криптоаналізу. Саме через це актуальною є задача створення та розробки таких криптографічних систем, що будуть захищені не тільки від вже відомих класичних атак, а й від атак, реалізованих для квантових комп'ютерів та квантової математики.

Досить великих зусиль для досягнення поставлених цілей докладає Національний інститут стандартів та технологій США, який організував найбільш багатообіцяючий конкурс за цим напрямком - NIST PQC [1]. Конкурс ставить своєю ціллю дослідження запропонованих алгоритмів для знаходження можливих кандидатів для затвердження нових стандартів за основними напрямками криптографії (симетричний шифр, асиметричний, електронний цифровий підпис). На даний момент проходить вже третій раунд даного конкурсу і його фіналістами за напрямком електронного підпису стали FALCON, Rainbow та CRYSTAL-DILITHIUM.

Схема електронного підпису Rainbow відрізняється від конкурентів математичною базою, оскільки заснована на мультиваріативних перетвореннях та є узагальненням структури Unbalanced Oil-Vinegar, що за рахунок додаткової алгебраїчної структури забезпечує ефективну параметризацію. Rainbow заявляє стійкість EUF-CMA та пропонує дуже малі підписи (528 біт для I рівня безпеки NIST), що э дуже гарним результатом у порівнянні з іншими кандидатами. Крім того, оскільки Rainbow використовує лише прості операції над невеликими скінченними полями, генерація та верифікація підпису є надзвичайно ефективними.

Метою статті є надання опису ключової пари, що використовується у алгоритмі ЕП Rainbow, та опису алгоритму генерації даної ключової пари, а також наведення певного покращення для пришвидшення генерації цієї пари.

Математична база та основні параметри алгоритму

У ЕП Rainbow відкритий ключ задається набором нелінійних багатовимірних поліномів над скінченним полем, так як сам алгоритм належить до мультиваріативних схем з відкритим ключем. Відкритий ключ такої системи – це система багатовимірних квадратичних поліномів, усі коефіцієнти та змінні якого походять з F_q – скінченного поля з q елементами. Приклад такої системи наведено нижче:

$$p^{(1)}(x_1,\ldots,x_n)=\sum_{i=1}^{n}\sum_{j=1}^{n}p_{ij}^{(1)}\cdot x_i x_j+\sum_{i=1}^{n}p_i^{(1)}\cdot x_i+p_0^{(1)}$$

$$p^{(2)}(x_1,\ldots,x_n)=\sum_{i=1}^{n}\sum_{j=1}^{n}p_{ij}^{(2)}\cdot x_i x_j+\sum_{i=1}^{n}p_i^{(2)}\cdot x_i+p_0^{(2)}$$

$$p^{(3)}(x_1,\ldots,x_n)=\sum_{i=1}^{n}\sum_{j=1}^{n}p_{ij}^{(3)}\cdot x_i x_j+\sum_{i=1}^{n}p_i^{(3)}\cdot x_i+p_0^{(3)}$$

$$\vdots$$

$$p^{(m)}(x_1,\ldots,x_n)=\sum_{i=1}^{n}\sum_{j=1}^{n}p_{ij}^{(m)}\cdot x_i x_j+\sum_{i=1}^{n}p_i^{(m)}\cdot x_i+p_0^{(m)}$$

© Д'ЯЧЕНКО А.С., ГОРБЕНКО І.Д., 2021

Тобто сукупність загальновідомих поліномів

$$\mathrm{P}\left(x_1,\ldots,x_n\right)=\left(p^{(1)}\left(x_1,\ldots,x_n\right),p^{(2)}\left(x_1,\ldots,x_n\right),\ldots,p^{(m)}\left(x_1,\ldots,x_n\right)\right),$$

математично являє собою відображення F_q^n до F_q^m. Операція верифікації підпису полягає у оцінюванні $\mathrm{P}\left(x_1,\ldots,x_n\right)$, тоді як процес генерації підпису відповідає інверсії зазначеного відображення, що є еквівалентним вирішенню проблеми MQ.

У загальному вигляді мультиваріативної криптосистеми з відкритим ключем, спочатку обирається центральне відображення F з m квадратичних поліномів та n змінних, які можна легко інвертувати. Потім, для того, щоб приховати структуру центрального відображення F у відкритому ключі, обираються два афінні інвертивні відображення S і T. Відкритим ключем криптосистеми є складене квадратичне відображення $\mathrm{P}=\mathrm{S}\circ\mathrm{F}\circ\mathrm{T}$ і головною умовою є те, що його досить важко інвертувати. Секретний ключ у свою чергу складається з S, F і T, що дозволяє нам P. У даному випадку з відкритим ключем, відображення S і T використовуються для захисту відображення F, яке, як зазначалося вище, складається з $m=n-\upsilon_1$ мультиваріативних квадратичних поліномів $f^{(\upsilon_1+1)},\ldots,f^{(n)}$ виду

$$f^{(k)}\left(\mathbf{x}\right)=\sum_{i,\,j\,\in\,V_\ell}\alpha_{ij}^{(k)}x_ix_j+\sum_{i\,\in\,V_\ell,\,j\,\in\,O_\ell}\beta_{ij}^{(k)}x_ix_j+\sum_{i\,\in\,V_\ell\,\cup\,O_\ell}\gamma_i^{(k)}x_i+\eta^{(k)},$$

де $\ell\in\{1,\ldots,u\}$ – єдине ціле число, таке що $k\in O_\ell$.

Зауважимо, що в кожному поліномі $f^{(k)}$ при $k\in O_\ell$ немає квадратного члена x_ix_j, де i і j знаходяться в O_ℓ. Цей факт використаний авторами [2] для генерації підпису, через що такі поліноми отримали назву поліномів Oil-Vinegar [3].

Для реалізації схеми підпису важливою умовою є нерівність $m\le n$. Генерація підпису для документа d починається з використання геш-функції $\mathrm{H}:\{0,1\}^*\to\mathrm{F}^m$ для приведення вхідного повідомлення до потрібної довжини та, як наслідок, отримання значення $\mathbf{w}=\mathrm{H}\left(d\right)\in\mathrm{F}^m$. Тоді обчислюються $\mathbf{x}=\mathrm{S}^{-1}\left(\mathbf{h}\right)$, $\mathbf{y}=\mathrm{F}^{-1}\left(\mathbf{x}\right)$ і $\mathbf{z}=\mathrm{T}^{-1}\left(\mathbf{y}\right)$. Підпис документа d – це $\mathbf{z}\in\mathrm{F}^n$. Тут $\mathrm{F}^{-1}\left(\mathbf{x}\right)$ означає пошук одного (з можливо багатьох) попередніх зображень $\mathbf{x}$ під центральним відображення F. Оскільки $m\le n$, ми можемо бути впевнені, що таке попереднє зображення існує. Тому кожне повідомлення має підпис. Для верифікації підпису в свою чергу просто обчислюється $\mathbf{w}'=\mathrm{P}\left(\mathbf{z}\right)$ і геш-значення $\mathbf{w}=\mathrm{H}\left(d\right)\in\mathrm{F}^m$ документа. Якщо виконується $\mathbf{w}'=\mathbf{w}$, підпис є валідним, в іншому випадку – він відхиляється. Схематично цей процес зображено на Рис 1.

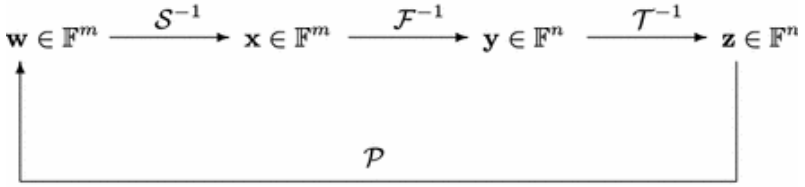


Рис.1 Процес генерації та верифікації підпису Rainbow

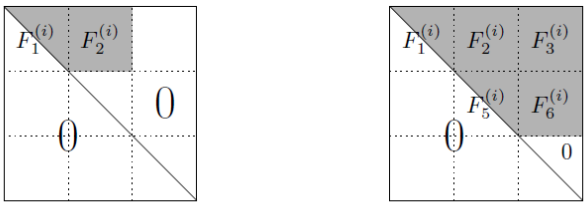
Основними системними параметрами є наступні змінні: кінцеве поле $\mathrm{F}=\mathrm{F}_q$ з елементами q; цілі числа $0<\upsilon_1<\ldots<\upsilon_u<\upsilon_{u+1}=n$; множина індексів $V_i=\{1,\ldots,\upsilon_i\},O_i=$ $=\{\upsilon_i+1,\ldots,\upsilon_{i+1}\}\left(i=1,\ldots,u\right)$, де кожне $k\in\{\upsilon_1+1,\ldots,n\}$ міститься рівно в одному з наборів

$O_i (\upsilon > u)$; маємо $|V_i| = \upsilon_i$ та множину $o_i := |O_i| (i = 1, ..., u)$; згадану раніше кількість рівнянь $m = n - \upsilon_1$; кількість змінних n. [3]

У наведеному прикладі змінними υ та u – незбалансовані поліноми Oil-Vinegar або їх відповідні коефіцієнти, кількість яких залежить від параметрів n та m.

Внесені пропозиції щодо покращення продуктивності

Для прискорення генерації ключової пари та відповідного підвищення швидкості роботи алгоритму, було прийнято рішення щодо введення проміжного відображення $\mathrm{Q} = \mathrm{F} \circ \mathrm{T}$. Відображення F, Q і P можна представити відповідними матрицями як показано на Рисунку 2.



$v_1 + 1 \le i \le v_2$ $\quad$ $v_2 + 1 \le i \le n$

Рис.2 Матриці $F^{(i)} (i = \upsilon_1 + 1, ..., n)$

Після того, як було обрано початкове 256-бітне значення, випадково згенеровано матриці та ненульові коефіцієнти відповідних центральних відображень, ми обчислення матриці $\mathrm{Q}^{(i)} = T^T \square F^{(i)} \square T$. Після обчислення усіх частин матриць $\mathrm{Q}^{(i)}$, обчислюється структура MQ [4], де $\frac{n \cdot (n+1)}{2}$ ненульових елементів матриці $\mathrm{Q}^{(i)}$ вставляються в $(i - \upsilon_1)$-й рядок MQ таким чином:

- Перші $\frac{\upsilon_1 \cdot (\upsilon_1 + 1)}{2} + \upsilon_1 o_1$ позиції $(i - \upsilon_1)$-го рядка заповнені елементами матриці $\mathrm{Q}_1^{(i)} \square \mathrm{Q}_2^{(i)}$ (зліва направо та зверху вниз).
- Наступні $\upsilon_1 \cdot o_2$ позиції заповнені елементами $\mathrm{Q}_3^{(i)}$ (так само зліва направо та зверху вниз).
- Наступні $\frac{o_1 \cdot (o_1 + 1)}{2} + o_1 o_2$ позицій заповнені елементами $\mathrm{Q}_5^{(i)} \square \mathrm{Q}_6^{(i)}$ (зліва направо та зверху вниз).
- Останні $\frac{o_2 \cdot (o_2 + 1)}{2} + o_1 o_2$ позиції рядка заповнені елементами матриці $\mathrm{Q}_9^{(i)}$.

Останньою обчислюється матриця, що містить відкритий ключ, – MP. Обидві матриці мають структуру, що складається з декількох шарів, які схематично зображено на Рисунку 3.

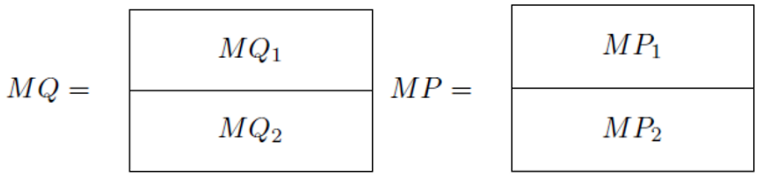


Рис.3 Матриці MQ *та* MP *для класичної схеми підпису Rainbow*

Наскільки таке впровадження є доцільним можна побачити на Рисунку 4. Заміри продуктивності алгоритму виконувалися на автономному комп'ютері, який обладнано процесором Intel(R) Core(TM) i7-9750H CPU.

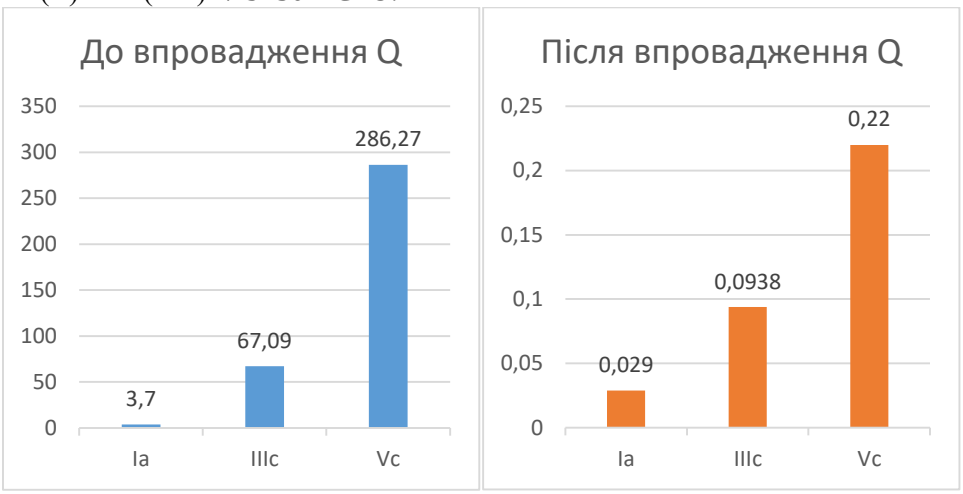


Рис. 4 Заміри продуктивності алгоритму до та після впровадження проміжної матриці Q

Висновки

Rainbow та багатовимірна математика загалом є досить перспективними напрямками у постквантовій криптології, що чудово демонструє конкурс NIST PQC. Сам алгоритм усе ще потребує певних досліджень, адже на даний момент точно можна зробити висновок щодо покращення продуктивності, за рахунок «розвантаження» обчислювальних операцій та введення структур матриць певного типу, але питання щодо збереження показників криптостійкості після введення таких структур відкрите, адже такі структури передбачають заповнення їх полів нулями, що повинно негативно вплинути на показники безпеки. Евристика цього впровадження залишає відкритим для дослідження питання щодо появи нових вразливостей та ін.

ЛІТЕРАТУРА

1. PQC Standardization Process: Third Round Candidate Announcement. July 22, 2020. [Electronic resource]. – Access mode: https://csrc.nist.gov/News/2020/pqc-third-round-candidate-announcement
2. J. Ding, D. Schmidt: Rainbow, a new multivariable polynomial signature scheme. ACNS 2005, LNCS vol. 3531, Springer, 2005, pp. 164 - 175.
3. A. Kipnis, J. Patarin, L. Goubin: Unbalanced Oil and Vinegar schemes. EUROCRYPT 1999, LNCS vol. 1592, Springer, 1999, pp. 206 – 222.
4. Albrecht Petzoldt, Stanislav Bulygin, Johannes Buchmann: Selecting Parameters for the Rainbow Signature Scheme - Extended Version. PQCrypto`10, 2010, pp. 21
5. Rainbow Signature / Ding J., and other. 2020. pp. 16-22. [Electronic resource]. – Access mode: https://www.pqcrainbow.org/

Д'ЯЧЕНКО Андрій Сергійович – студент кафедри безпеки інформаційних систем та технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, Харків, Україна, 61108; e-mail: andrey.090220@gmail.com; ORCID: 0000-0002-2342-4231.

Наукові інтереси:

– *електронний підпис, мультиваріативна криптографія;*

ГОРБЕНКО Іван Дмитрович – доктор технічних наук, професор, головний конструктор АТ «Інститут інформаційних технологій», професор кафедри безпеки інформаційних систем та технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, Харків, Україна, 61108; e-mail: GorbenkoI@iit.kharkov.ua;

Наукові інтереси:

– *електронний підпис, мультиваріативна криптографія*.

УДК 004.942 **СЕКЦІЯ 2**

ДЯЧЕНКО В.Г., МІНЯЙЛОВ Є.С., УГРЮМОВ М.Л.

ІМОВІРНІСТНІ МЕТОДИ ГЕНЕРАЦІЇ ПРОБНИХ ВИБІРОК, ЩО МАЮТЬ ЗАДАНИЙ РОЗПОДІЛ

Вступ і актуальність

Розглядаються імовірнісні методи генерації пробних вибірок, що мають заданий розподіл. Розглянуто методи генерації випадкових і псевдовипадкових послідовностей. Охарактеризовано кожен із методів та описані їхні переваги та недоліки. За результатами складено графік залежності відхилення від кількості елементів вибірки для одновимірних випадків.

Формування вихідних тестових вибірок даних є однією з головних проблем, що виникають у момент побудови алгоритмів стохастичної оптимізації. Звертаючи увагу на масштабність застосування машинного навчання у різних сферах життя сучасної людини, динаміку ускладнення задач, які вирішуються в області аналізу даних, створення систем штучного інтелекту, потреби в ефективності, якості, чіткості будуть зростати.

Тому виникає потреба в дослідженні, порівнянні та аналізі імовірнісних методів генерації пробних вибірок, що мають заданий розподіл.

Проблема генерації випадкових послідовностей

В умовах сьогодення, наукові дослідження у сфері імовірнісних методів генерації пробних вибірок, що мають заданий розподіл є доволі масштабними. Автори підходять до розгляду теми як із теоретичного, так і з практичного аспектів, формуючи коло вирішених задач та спонукаючи до появи нових.

Однак, незважаючи на масштабність наукових досліджень, дослідження імовірнісних методів генерації пробних вибірок, що мають заданий розподіл залишається відкритим та потребує детального опрацювання в напрямку, зокрема, використання квазівипадкових генераторів та інших методів покращення сходження вибірки до параметрів розподілів.

Основна проблема генерації пробних вибірок пов'язана з відхиленням практичних значень параметрів розподілів від заданих для теоретичних розподілів, як в більшу сторону так і в меншу.

Було розглянуто одновимірний генератор рівномірного розподілу з різними генераторами з метою дослідження залежності відносної похибки вибірки від кількості її елементів і типу розподілу. [1]

Для одновимірного випадку були розглянуті наступні генератори випадкових чисел:

1. Стандартний генератор псевдовипадкових послідовностей з рівномірним розподілом бібліотеки random середовища Python (random.uniform). [2]
2. Генератор псевдовипадкових послідовностей операційних систем сімейства Unix dev/urandom. [3]

Метою дослідження є аналіз алгоритмів генерації вихідних тестових вибірок даних, що мають заданий розподіл, для підвищення якості діагностування за сферою впровадження.

Генерація одновимірних послідовностей

Для обчислення відносної похибки одномірної вибірки в залежності від кількості елементів у ній було задано наступні параметри: E (Expected value) – математичне очікування; D – відхилення; N_min – мінімальна кількість елементів вибірки; N_max – максимальна кількість елементів вибірки; step – крок збільшення кількості елементів від N_min до N_max; required_error – максимально допустима помилка. Потрібна для знаходження мінімальної кількості елементів, більше котрої помилка завжди менша required_error; randomizer – об'єкт генератора випадкових чисел, наслідує абстрактний метод generate.

© ДЯЧЕНКО В.Г., МІНЯЙЛОВ Є.С., УГРЮМОВ М.Л., 2021

Для E = 1; D = 1; N_min = 100; N_max; step = 100; required_error = 1.5 (%) і стандартного генератора random.uniform, були отримані наступні результати (Рисунок 1, Рисунок 2).

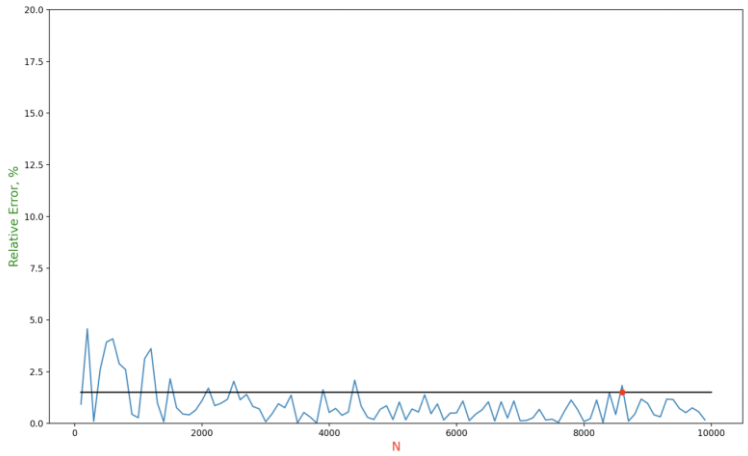


Рис. 1 Графік залежності відносної похибки математичного очікування від кількості елементів у вибірці для одновимірного випадку з генератором random.uniform

Затребувана похибка не більше заданої при кількості елементів N = 8600, відповідно до результату роботи програми.

```
vitalydiachenko@Vitaliis-MacBook-Pro Python % python3 generators

1: "single_dimension_uniform"
2: "multiple_dimensions_uniform"

Enter task or exercise number: 1
The task is found!

Enter expected value E = 1
Enter variance value D = 1
Enter min number of elements in the generated sequence N_min = 100
Enter min number of elements in the generated sequence N_max = 10000
Enter step for number of elements in the generated sequence step = 100
Enter number required relative error Relative error, % = 1.5
Choose randomizer:

#0. native

#1. urandom

Randomizer #0
Suitable N for required delta is 8600
```

Рис. 2 Результат роботи програми для одновимірного випадку з генератором random.uniform

Наступним кроком була генерація послідовності для таких же вхідних параметрів за відмінністю генератора – тепер це квазівипадковий dev/urandom (Рисунок 3, Рисунок 4).

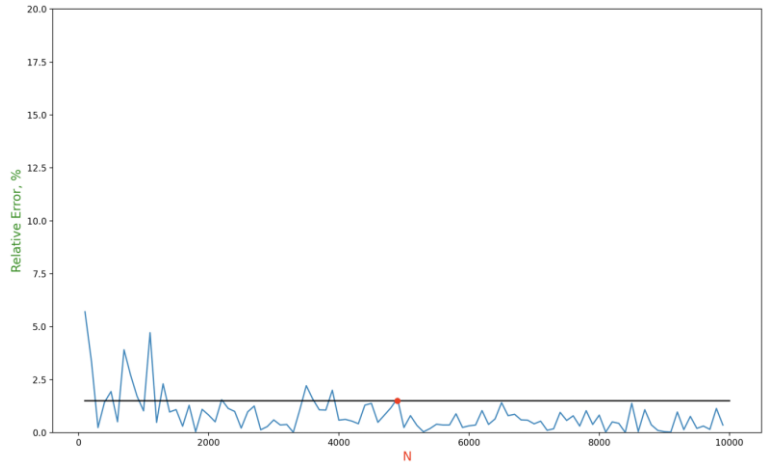


Рис. 3 Графік залежності відносної похибки значень математичного очікування від кількості елементів у вибірці для одновимірного випадку з генератором urandom

Затребувана похибка не більше заданої при кількості елементів N = 4900 відповідно до результату роботи програми, що суттєво швидше, ніж з random.uniform.

```
vitalydiachenko@Vitaliis-MacBook-Pro Python % python3 generators

1: "single_dimension_uniform"
2: "multiple_dimensions_uniform"

Enter task or exercise number: 1
The task is found!

Enter expected value E = 1
Enter variance value D = 1
Enter min number of elements in the generated sequence N_min = 100
Enter min number of elements in the generated sequence N_max = 10000
Enter step for number of elements in the generated sequence step = 10
Enter number required relative error Relative error, % = 1.7
Choose randomizer:

#0. native

#1. urandom

Randomizer #1
Suitable N for required delta is 8340
vitalydiachenko@Vitaliis-MacBook-Pro Python %
```

Рис. 4 Результат роботи програми для одновимірного випадку з генератором dev/urandom

Генерація багатовимірних послідовностей

В якості генератора багатовимірних послідовностей, був розроблений генератор послідовностей Соболя. Послідовності Соболя є прикладом квазівипадкових послідовностей з низькою розбіжністю. [4]

Для побудови послідовності Соболя на одиничному інтервалі задаються наступні параметри: Dim – розмірність послідовності, n – кількість точок послідовності. [5]

Нижче наведено приклад генерації двовимірної послідовності на 700 точок (Рисунок 5) і виведено графік такої послідовності (Рисунок 6).

```
vitalydiachenko@Vitaliis-MacBook-Pro Python % python3 generators

1: "single_dimension_uniform"
2: "multiple_dimensions_uniform"
3: "lp_tau_uniform"
4: "lp_tau_normal"

Enter task or exercise number: 3
The task is found!

Enter dimensions number Dim = 2
lp_tau.points_number n = 700
[[0.5        0.5       ]
 [0.75       0.25      ]
 [0.25       0.75      ]
 ...
 [0.90527344 0.22363281]
 [0.40527344 0.72363281]
 [0.28027344 0.09863281]]
```

Рис. 5 Генерація двовимірної послідовності Соболя на 700 точок

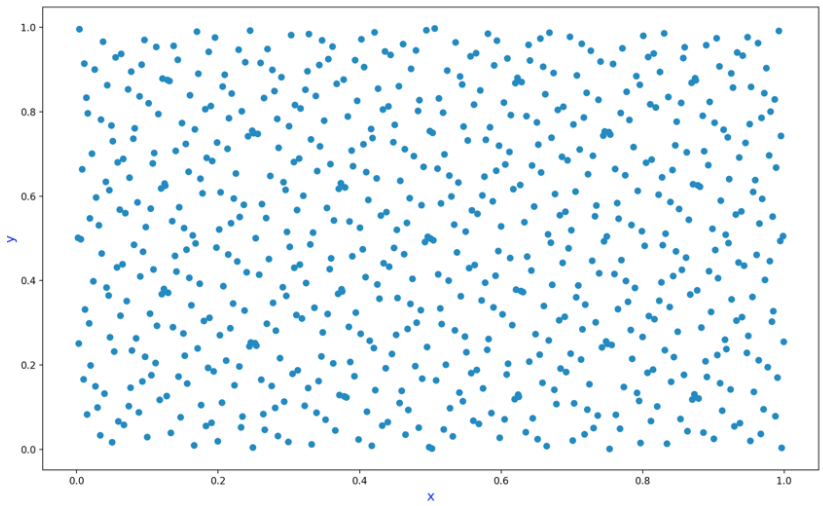


Рис.6 Виведення графіку двовимірної послідовності Соболя на 700 точок

Підрахунок функції розподілу імовірності за допомогою scipy.stats.uniform.pdf [6] показує низьку розбіжність, що відповідає вимогам висунутим до генератора.

Висновки

У процесі дослідження було розглянуто імовірнісні методи генерації пробних вибірок, що мають заданий розподіл. Розглянуто методи генерації випадкових і псевдовипадкових послідовностей. За результатами дослідження було складено графіки залежності відносної похибки значень математичного очікування від кількості елементів у вибірці для одновимірного і багатовимірного випадку. Експериментальним шляхом було досліджено, що використання квазівипадкових генераторів замість псевдовипадкових сприяє зменшенню відносної похибки вибірки для такої ж кількості елементів і, відповідно, прискорює зближення практичного значення математичного очікування і теоретичного. Було розглянуто генератор послідовностей Соболя та реалізовано програмно. Аналіз якісних показників багатовимірних послідовностей планується розглянути детально в наступних дослідженнях.

ЛІТЕРАТУРА

1. https://www.researchgate.net/publication/300745862_Discrete_Uniform_Distribution (date of access: 10.04.2021).
2. https://docs.python.org/3/library/random.html (date of access: 10.04.2021).
3. https://man7.org/linux/man-pages/man4/random.4.html (date of access: 10.04.2021).
4. Sobol,I.M. (1967), "Distribution of points in a cube and approximate evaluation of integrals". Zh. Vych. Mat. Mat. Fiz. 7: 784–802 (in Russian); U.S.S.R Comput. Maths. Math. Phys. 7: 86–112 (in English).
5. https://people.sc.fsu.edu/~jburkardt/cpp_src/sobol/sobol.html (date of access: 10.04.2021).
6. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.uniform.html#scipy.stats.uniform (date of access: 10.04.2021).

ДЯЧЕНКО Віталій Геннадійович – аспірант 3-го року кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: v.di4chenko@gmail.com; ORCID: 0000-0001-7966-8596.

Наукові інтереси:

– *методи обчислювального інтелекту в багатокритеріальних задачах робастного оптимального проектування та інтелектуального діагностування систем і процесів в умовах невизначеності;*

МІНЯЙЛОВ Євген Сергійович – старший викладач кафедри теоретичної та прикладної інформатики, Харківського національного університету імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: j.menyailov@khai.edu; ORCID: 0000-0002-9440-8378.

Наукові інтереси:

- *методи обчислювального інтелекту в багатокритеріальних задачах робастного оптимального проектування та інтелектуального діагностування систем і процесів в умовах невизначеності;*

- *системи підтримки прийняття рішень.*

УГРЮМОВ Михайло Леонідович - д.т.н., професор, професор кафедри теоретичної та прикладної системотехніки Харківського національного університету імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: m.ugryumov@karazin.ua; ORCID: 0000-0003-0902-2735.

Наукові інтереси:

- *методи обчислювального інтелекту в багатокритеріальних задачах робастного оптимального проектування та інтелектуального діагностування систем і процесів в умовах невизначеності.*

- *системи підтримки прийняття рішень.*

УДК 004.056 **СЕКЦІЯ 5**

ЄСІН В.І, СОКОЛОВСЬКИЙ К.О.

КОНТРОЛЬ ЦІЛІСНОСТІ ЗБЕРЕЖЕНИХ ПРОГРАМ БАЗ ДАНИХ НА ОСНОВІ ВИКОРИСТАННЯ СТРУКТУРИ ҐЕШ-ДЕРЕВА

Вступ

У наш час бази даних (БД) є невід'ємною частиною кожної інформаційної системи (ІС). БД стали необхідним інструментом практично у всіх сферах діяльності людини. Що в свою чергу відбилося і на зворотному боці цього процесу. А саме, зріс інтерес до циркулюючої всередині ІС інформації не тільки з боку законних користувачів і власників, а й з боку зловмисників. Тому зараз гостро стоїть питання їх захисту.

У ситуації, що склалася, беручи до уваги сучасний стан розвитку техніки, кваліфікацію зловмисників, які постійно удосконалюють можливості шкідливого впливу за допомогою відповідного програмного забезпечення (malware), проаналізувавши різні нормативно-правові акти, принципи і системи забезпечення безпеки інформації, було прийнято рішення про доцільність захисту не тільки даних (фактів), що зберігаються в БД, а й про забезпечення безпеки деяких інших важливих об'єктів бази даних, які здійснюють управління даними. Постійний моніторинг цих об'єктів бази даних дуже важливий, оскільки деякі з атак на БД можуть бути виявлені саме на основі аналізу зміни (навмисного або випадкового) цих об'єктів (порушення їх цілісності, автентичності), або їх набору (збільшення або зменшення їх числа) на сервері БД. Мова йде про так званих постійно збережених модулях [1], спеціальним чином оформлених програм, які зберігаються в базі даних, можуть викликатися застосунками і виконуються всередині СУБД. На жаль, розробки методів верифікації набору збережених програм і моніторингу їх цілісності не повною мірою відповідають сучасним вимогам, обумовлених об'єктивною потребою раціонального витрачання ресурсів. Більшість існуючих підходів базується на використанні операцій ґешування. Але чим більший об'єм бази даних, тим більше ресурсів потрібно на розрахунки. Тому знаходження рішення, завдяки якому можна істотно зменшити обсяг ресурсів, необхідних для верифікації набору збережених програм (ЗП) і моніторингу їх цілісності, є актуальним напрямком дослідження.

Нижче пропонується рішення, що ґрунтується на концепції ґеш дерев.

Алгоритми побудови дерева Меркла

Доцільність використання структури ґеш-дерева [2], кореня Меркла, механізму цифрового підпису для контролю цілісності і справжності збережених в конкретній схемі БД об'єктів в пропонованому підході обумовлена об'єктивною потребою раціонального витрачання ресурсів, що веде до економії обсягу збережених для цього даних і обчислювального ресурсу процесора. А саме, для того щоб здійснити контроль цілісності та автентичності конкретної ЗП в конкретній схемі БД звичайним способом було б потрібно виконати процедури ґешування і цифровий підпис зі збереженням відповідних даних для кожного з них. До того ж у багатьох випадках кінцевий користувач, володіючи обмеженими обчислювальними ресурсами, за наявною в його розпорядженні інформації не зможе перевірити самостійно цілісність коду.

Дерева Меркла – це бінарні дерева, що містять криптографічні ґеш-значення [3]. Дерева Меркла створюються шляхом багаторазового ґешування пар вузлів до тих пір, поки не залишиться тільки один ґеш (цей ґеш називається кореневим ґешем, або коренем Меркла). Вони будуються знизу вгору з ґешів окремих блоків даних, в ролі яких в даному випадку виступають ґеш-значення коду збережених програм. Кожен листовий вузол є ґешем блоку даних, а кожен нелистовий вузол – ґешем суми своїх попередніх ґешів.

Коли N елементів даних ґешуються і конкатенуються в дереві Меркла, то перевірка того, чи є такий елемент даних (ЗП БД) в дереві, потребують не більше, ніж $2 \times \log_2 N$ обчислень, що робить цю структуру даних дуже ефективною [3]. Набір ґешів, які необхідно визначити (а саме, $\log_2 N$ ґешів) для доказу приналежності ЗП до блоку блокчейна, становить так званий шлях автентифікації або шлях Меркла, що з'єднує конкретний елемент блоку даних з коренем дерева.

© ЄСІН В.І, СОКОЛОВСЬКИЙ К.О., 2021

Таким чином, досить перевірити тільки корінь дерева Меркла, і в результаті будуть верифіковані все блоки даних, що містяться в дереві. Тому дерева Меркла знайшли широке застосування для безпечної та ефективної валідації великих структур даних (перевірки їх цілісності) [3-5].

Для побудови дерева Меркла на основі збережених програм БД було розроблено дві версії алгоритму. Перша версія алгоритму (рис. 1) приймає на вхід масив збережених програм і функцію ґешування. Масив сортується в алфавітному порядку і для кожної програми рахується ґеш-значення її коду. Далі рівень за рівнем формуються нові вузли, доки на одному з рівнів не залишиться один. Це і буде корінь дерева Меркла. Для формування вузлів нового рівня потрібно послідовно брати пару з попереднього і, використовуючи функцію ґешування, об'єднувати їх ґеші для отримання нового.

```
Constructing Merkle tree algorithm v1

Input: SP = [SP_1, … , SP_n], hash
Output: Merkle root hash – h_root

 1: Sort stored objects SP_k alphabetically (depending on
    their name and type)
 2: h_k = hash(0x00 || SP_k), k = 1..n
 3: hashes = [h_1,…,h_n]
 4: prev_hashes = hashes
 5: parents_hashes = []
 6: while size_of(prev_hashes) != 1
 7:     for i = 0 to size_of(prev_hashes) - 1
 8:         left = prev_hashes[i]
 9:         right = prev_hashes[i + 1] ?? null
10:         if (right == null):
11:             h = left
12:         else:
13:             h = hash(0x01 || left || right)
14:         parents_hashes.add(h)
15:         i += 2
16:     prev_hashes = parents_hashes
17:     parents_hashes = []
18: h_root = prev_hashes[0]
19: Save serialized version of the tree
20: return h_root
```

Рис. 1 Алгоритм 1 побудови дерева Меркла

У разі виникнення ситуації, коли кількість вузлів на попередньому рівні непарна, ця версія алгоритму не рахує новий ґеш, а встановлює значення ґешу для нового вузла рівне значенню лівого дочірнього. На виході з алгоритму маємо значення ґешу кореня побудованого дерева Меркла.

Друга версія алгоритму (рис. 2) принципово не відрізняється від першої. У разі виникнення ситуації, коли кількість вузлів на попередньому рівні непарна, ця версія алгоритму, на відміну від першої, не встановлює значення ґешу для нового вузла рівне значенню лівого дочірнього.

```
Constructing Merkle tree algorithm v2

Input: SP = [SP_1, … , SP_n], hash
Output: Merkle root hash – h_root

 1: Sort stored objects SP_k alphabetically (depending on
    their name and type)
 2: h_k = hash(0x00 || SP_k), k = 1..n
 3: hashes = [h_1,…,h_n]
 4: prev_hashes = hashes
 5: parents_hashes = []
 6: while size_of(prev_hashes) != 1
 7:     for i = 0 to size_of(prev_hashes) - 1
 8:         left = prev_hashes[i]
 9:         right = prev_hashes[i + 1] ?? null
10:         if (right == null):
11:             h = hash(0x01 || left || left)
12:         else:
13:             h = hash(0x01 || left || right)
14:         parents_hashes.add(h)
15:         i += 2
16:     prev_hashes = parents_hashes
17:     parents_hashes = []
18: h_root = prev_hashes[0]
19: Save serialized version of the tree
20: return h_root
```

Рис. 2 Алгоритм 2 побудови дерева Меркла

Алгоритм побудови шляху автентифікації

Алгоритм побудови шляху автентифікації (рис. 3) приймає на вході дані збереженої програми: назву і тип, а також версію дерева Меркла, яку було отримано шляхом десеріалізації збереженої серіалізованої версії. Спочатку потрібно отримати посилання на потрібний листовий вузол дерева за допомогою даних збереженої програми. Далі послідовно, доки не дійдемо до кореня, отримуємо ґеш-значення сестринського вузла і його розташування відносно теперішнього вузла: зліва або справа.

```
Constructing Merkle tree auth path algorithm

Input: prog_name, prog_type, tree
Output: path

 1: Get reference to the tree node which represents
    hash of the program with name equals to prog_name
    and type equals to prog_type and save it in node
    variable
 2: path = []
 3: while node.parent != null:
 4:     parent = node.parent
 5:     if (parent.right == node):
 6:         path.add(["RIGHT", node])
 7:     else:
 8:         path.add(["LEFT", node])
 9:     node = parent
10: return path
```

Рис. 3 Алгоритм побудови шляху автентифікації

На виході з алгоритму маємо шлях автентифікації у вигляді масиву, який містить усі потрібні значення для перевірки вузла на цілісність.

Алгоритм верифікації збереженої програми

Для верифікації збереженої програми за допомогою дерева Меркла було розроблено дві версії алгоритму.

Перша версія алгоритму (рис. 4) приймає на вхід отриманий на попередньому кроці шлях автентифікації у вигляді масиву, код збереженої програми, функцію ґешування і значення кореня дерева Меркла. Далі, проходячи по всім елементам зі шляху автентифікації, перевіряється розташування сестринського вузла і виходячи з нього розраховується значення для наступного вузла. У разі виникнення ситуації, коли правого сестринського вузла немає, в даній версії алгоритму не рахується новий ґеш, а встановлюється значення ґешу для нового вузла рівне значенню лівого сестринського.

```
Verifying Merkle tree auth path algorithm v1

Input: path, SP, hash, root
Output: status

 1: h = hash(SP)
 2: for each entry in path:
 3:     if (entry[0] == "RIGHT"):
 4:         left = h
 5:         right = entry[1]
 6:     else:
 7:         left = entry[1]
 8:         right = h
 9:     if (right == null):
10:         h = left
11:     else:
12:         h = hash(0x01 || left || right)
13: if (root == h):
14:     return 1
15: else:
16:     return 0
```

Рис. 4 Алгоритм верифікації збереженої програми

На виході маємо статус перевірки: 1, якщо перевірка успішна і 0, якщо ні.

Друга версія принципово не відрізняється від першої. Але у разі виникнення ситуації, коли правого сестринського вузла немає, ця версія алгоритму, на відміну від першої, не встановлює ґеш-значення для нового вузла рівне значенню лівого сестринського.

Висновки

Запропоноване рішення дозволяє більш ефективним способом здійснювати контроль цілісності збережених в БД програм і верифікацію їх набору. За рахунок використання структури ґеш дерева забезпечується контроль цілісності не тільки конкретної програми, що перевіряється, а й усіх інших збережених програм обраної схеми БД. Так як цей один фрагмент даних входить в загальну структуру і зміна хоча б одного біта в ньому спричинить за собою повну зміну значення кореня дерева Меркла.

Крім того, у багатьох випадках кінцевий користувач, володіючи обмеженими обчислювальними ресурсами, за наявною в його розпорядженні інформації не зможе перевірити самостійно цілісність коду. Пропоноване ж рішення дозволяє вирішити цю проблему.

ЛІТЕРАТУРА

1. Garcia-Molina H., Ullman J. D., Widom J. Database Systems. The Complete Book, 2th ed. – Pearson Prentice Hall, 2009, – 1203 p.
2. Patent 4,309,569, United States, Method of providing digital signatures / Ralph C. Merkle, Mountain View, Calif.; The Board of Trustees of the Leland Stanford Junior University, Stanford, Calif. – No. 72,363; Jan. 5, 1982.
3. Antonopoulos A. M. Mastering Bitcoin: Programming the Open Blockchain 2nd Edition. – O'Reilly Media, 2017. – 416 p.
4. Wei W., Yu T. Integrity Assurance for Outsourced Databases without DBMS Modification // IFIP Annual Conference on Data and Applications Security and Privacy. – Springer, Berlin, Heidelberg, 2014. – pp. 1-16.
5. Niaz M. S., Saake G. Merkle Hash Tree based Techniques for Data Integrity of Outsourced Data // Proceedings of the 27th GI-Workshop Grundlagen von Datenbanken, Gommern, Germany, May 26-29, 2015. – pp. 66-71.

ЄСІН Віталій Іванович – д. т. н., доцент; професор кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: v.i.yesin@karazin.ua; ORCID: 0000-0003-1977-7269

Наукові інтереси:

– моделі і методи розробки баз даних інформаційних систем і забезпечення їх безпеки.

СОКОЛОВСЬКИЙ Костянтин Олександрович – студент факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: sokolovskyi.konstantin@gmail.com; ORCID: 0000-0002-6593-6534

Наукові інтереси:

– безпека даних в інформаційних і комунікаційних системах;

– криптографічний захист інформації.

УДК 65.0(075.8) **СЕКЦІЯ 2**

ЖМИРОВ Д.А, БЕРДНІКОВ А.Г.

МОДЕЛЮВАННЯ РИЗИКІВ ПРИ РЕАЛІЗАЦІЇ IT-ПРОЕКТІВ

Вступ

В даний час вимоги ринку до якості представлених товарів і послуг дуже високі, а час, який мають виробники для випуску на ринок нової продукції, прагне до мінімуму.

При організації роботи по проекту в скорочені терміни необхідно враховувати всі критичні фактори, що впливають на розробку. У зв'язку з цим необхідно вміти працювати з ризиками підприємства і враховувати можливі наслідки, які вони можуть спричинити за собою.

Проблема дослідження ризиків в процесах розробки програмного забезпечення ускладнюється внаслідок зростання різноманітності і складності розроблюваних програмних продуктів. Створення програмного забезпечення характеризується неможливістю чітко описати продукт проекту на початкових стадіях його реалізації. Нездатність стратегічно вірно керувати ризиками може призвести до відчутних економічних витрат, які могли б бути завчасно зменшені шляхом своєчасного аналізу, прогнозування ризиків і вжиттям заходів щодо їх зниження. Це робить задачу управління ризиками однією з найважливіших та найактуальніших.

Метою даної роботи є розробка моделі оцінки ризиків в IT-проектах, яку після апробації на практичному занятті в навчальному процесі можна рекомендувати для аналізу і ідентифікації ризиків у відповідній компанії.

Постановка задачі

IT-компанії потрібна CRM-система, для їх внутрішнього продукту, щоб якісно займатися управлінням взаємин з клієнтами. Проект треба закінчити швидко, бо від цього залежить яку долю ринку продукту може зайняти компанія, тому керівництво компанії хоче отримати цей продукт через 4 місяці (960 годин).

Продукт компанії - це система інтернет-бронювання місць для оренди готелів, котеджів, апартаментів і т.д. в різних країнах світу. Розроблена CRM-система допоможе автоматизувати комунікацію з клієнтами, що спричинить поліпшення якості і підвищення попиту на продукт.

CRM - прикладне програмне забезпечення, призначене для автоматизації взаємодії з покупцями (замовниками), зокрема для збільшення рівня продажів, оптимізації маркетингу і вдосконалення обслуговування замовників. Досягається це за рахунок зберігання інформації про покупців, історії відносин з ними, удосконалення відповідних бізнес-процесів і майбутнього аналізу результатів.

На цей проект виділяється певні людські, часові та фінансові ресурси, а також висуваються певні вимоги до якості.

Для прийняття оптимального рішення необхідно класифікувати ризики, кількісно і якісно оцінити можливі наслідки для компанії при обох варіантах, сформулювати пропозиції керівництву з моніторингу ситуації та проведення заходів з управління ризиками.

Класифікація ризиків в IT-проектах

Невизначеність при прийнятті рішень по організації роботи над проектом характеризується наступними аспектами: неповним знанням всіх обставин ситуації; неможливістю адекватного і точного обліку всієї інформації; факторами, які апріорі неможливо передбачити і спрогнозувати. Названі умови пов'язані між собою і приводять до потенційних ризиків, які є джерелами несприятливих ситуацій і пов'язаних ними наслідків у вигляді втрат і збитків.

У роботі зроблена класифікація ризиків IT-компанії, яка дозволяє систематизувати фінансові та часові ризики, а також ризики зниження якості продукції. Дана класифікація представлена на схемі як дерево ризиків, утворене групами ризиків різної природи.

© ЖМИРОВ Д.А, БЕРДНІКОВ А.Г., 2021

Дерево ризиків представляє собою ієрархічну структуру. У якості елементів структури ризиків виступають різні ризики по значимості та характером. Дерево ризиків може бути використано для проведення якісного та кількісного аналізу ризиків та забезпечує здійснення процесу систематичної ідентифікація ризиків в залежності від рівня деталізації і в зв'язку з іншими елементами проекту.

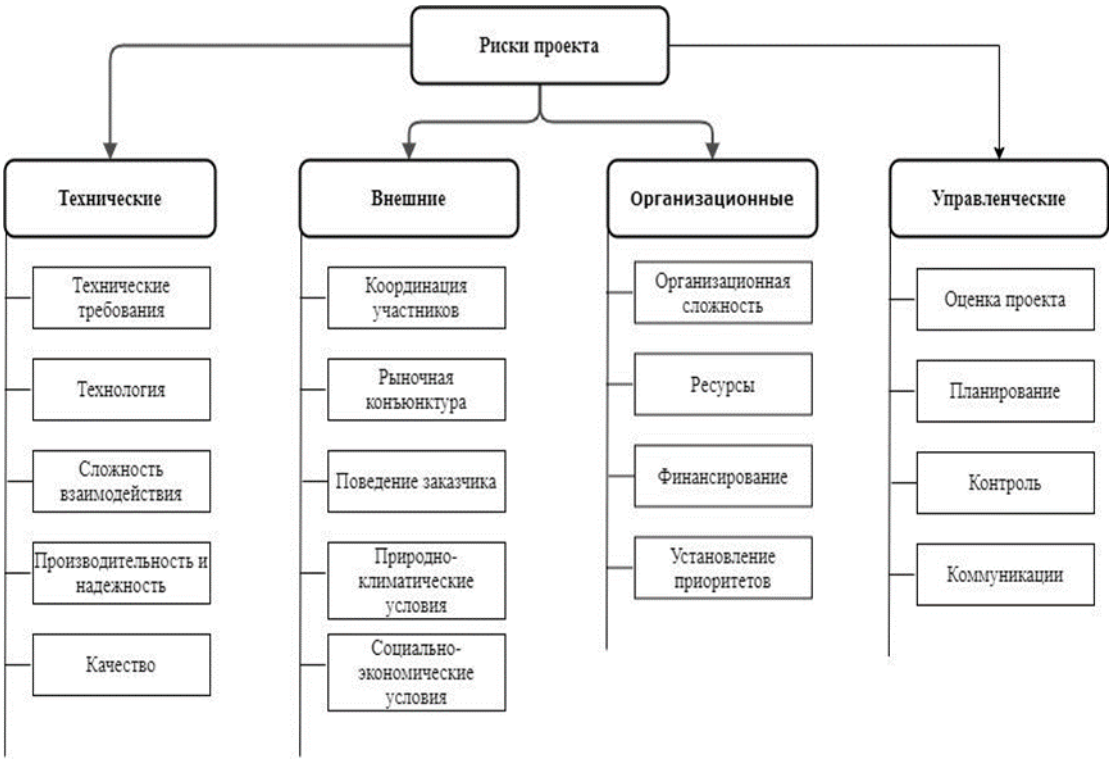


Рис. 1 Дерево Ризиків

Принцип оцінки ризиків передбачає врахування взаємозв'язку кількісних і якісних показників.

Якісний аналіз ризиків дозволяє менеджеру виявити причину появи ризику і визначити, на яких стадіях є загроза його виникнення. Таким чином, встановлюються можливі області ризику і обчислюються можливі збитки при його виникненні.

При кількісному аналізі ризиків проводиться оцінка ризиків, які фігурують у відповідній операції алгоритму прийняття рішення.

Менеджер, що приймає рішення, отримує числові значення параметрів по кожному ризику і по об'єкту в цілому, обчислює потенційна втрата, виробляє його вартісну оцінку і визначає дії для нейтралізації наслідків з фінансовим розрахунком.

Для ефективного виконання цього завдання менеджеру необхідний відповідний інструментарій, яким і є розроблена модель оцінки ризиків.

Суть розробленої моделі оцінки ризиків ІТ-проекту

Для оцінки фінансових і тимчасових ризиків проекту розроблена модель, яка включає в себе комплексне використання методу дерева рішень і методу PERT (Project Evaluation and Review Technique), що на нашу думку робить оцінку більш точною.

Дерево рішень являє собою модель, яка дозволяє розбити складну проблему прийняття рішень в умовах ризику на сукупність менших проблем, кожна з яких може бути розглянута окремо, а потім в сукупності. Застосування цього методу передбачає, що тимчасові параметри процесу є детермінованими.

Однак на практиці тимчасові показники проекту можуть характеризуватися випадковими параметрами, які підкоряються власним законом розподілу зі своїми числовими характеристиками, а невизначеність термінів виконання окремих операцій означає, що загальна тривалість робіт також схильна до невизначеності. Тому з метою підвищення точності

визначення часових ризиків пропонується використовувати метод PERT, який дозволяє оцінювати часові параметри проекту в умовах випадкового характеру тривалості робіт.

Хоча в нинішній час існує багато різних методів оцінки ризиків і кількість методів з часом зростає, дана модель має змогу конкурувати з ними за рахунок прозорості та легкості в інтегруванні, але за нижчою ціною. Через це дана модель є універсальною та може використовуватися не тільки в ІТ-проектах. Одним з основних мінусів проекту – це важкість інтегрування у достатньо великі проекти. Основне застосування моделі передбачається для малих та середніх за розмірами проектів.

Протягом року над розробкою моделі, вона стала більш точною ефективною в зв'язку з тим, що пов'язує найбільш небезпечні і ймовірні ризики з можливістю керувати ними та формувати певні коригувальні впливи на проект.

Основною процедурою етапу управління ризиками є вибір методу управління ризиками і подальше його застосування. Підприємство має розглядати всі групи методів управління ризиками і застосовувати їх як на фазі планування, так і на фазі реалізації проекту. Безпосередньо під час застосування методів управління ризиками відбувається моніторинг за цим процесом, який може виявити необхідність різних коригувань.

Таким чином, розроблена модель управління проектними ризиками в ІТ-проекту дозволить структурувати саму систему управління ризиками. Впровадження централізованої системи управління ризиками здатне також вирішити проблему появи непередбачених ризиків для підприємства і його проектів, спростити прийняття рішень в області ризиків і знизити витрати на управління ризиками в порівнянні з існуючим рівнем.

ЛІТЕРАТУРА

1. Управление проектом. Основы проектного управления: учебник / коллектив авторов под редакцией М.Л. Разу. – М.: изд КНОРУС, 2006 – 768.
2. Теория принятия решений: учебник / Черноморов Г. А; изд: Ред. Журнал «Изв. вузов, Электромеханика», 2002 - 276.

ЖМИРОВ Данило Андрійович Олексій Миколайович – студент групи КІ-51 факультету Комп'ютерних Наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: reus11fan@gmail.com; ORCID 0000-0003-3661-2642.

Наукові інтереси:

– *Розробка Single Page Application,*

– *Управління ризиками ІТ-проектів.*

БЕРДНІКОВ Анатолій Георгійович – к. т. н., доцент; доцент кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені. В.Н. Каразіна; м. Харків, майдан. Свободи, 6, 61022; e-mail:a.berdnikov@karazin.ua, ORCID 0000-0001-7923-0754.

Наукові інтереси:

– *застосування методів теорії ухвалення рішень при управлінні складними телекомунікаційними системами, управління проектами*.

УДК 004.41.02 **СЕКЦІЯ 6**

ЗІНОВ'ЄВ Д.В., ТКАЧУК М.В.. ТРІЩЕНКО І.В.

МОДЕЛІ ТА ТЕХНОЛОГІЇ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ СЕРВІС-ОРІЄНТОВАНИХ ПРОГРАМНИХ СИСТЕМ: СУЧАСНИЙ СТАН ТА ПЕРСПЕКТИВНІ НАПРЯМКИ ДОСЛІДЖЕНЬ

Актуальність теми та мета дослідження

На теперішній час в ІТ-галузі постійно зростає складність процесів розробки програмного забезпечення (ПЗ) внаслідок того, що бізнес-вимоги користувачів передбачають інтеграцію окремих програмних систем (ПС), можливість їх швидкої модернізації, адаптації, та підвищення показників їх якості. Традиційні підходи до створення розподілених ПС (РПС) на основі клієнт-серверних архітектур з 2-х або 3-х рівнями розміщення їх компонентів [1], які отримали загальну назву монолітних архітектур (МА), за багатьма показниками вже не встигають за динамікою зростання бізнес-потреб сучасних підприємств. Це привело до появи нових концепцій і технологій створення ПЗ, і однією з широко використовуваних в останній час методологій розробки РПС є використання сервіс-орієнтованої архітектури (СОА) [2]. Застосування СОА при розробці РПС забезпечує підвищення гнучкості корпоративної ІТ-інфраструктури, зниження витрат на розробку ПЗ; здатність до модернізації, конфігурації і супроводу РПС. В той же час в РПС, які створюються на основі парадигми СОА, актуальною і досить складною науково-практичною задачею є забезпечення відповідного рівня їх якості, тобто таких показників як продуктивність, надійність, супроводжуваність та деяких інших [3]. Саме тому метою цього дослідження є стислий аналіз сучасних проблем у цій області та визначення можливих перспективних підходів до їх вирішення.

Особливості побудови та застосування розподілених програмних систем на основі сервіс-орієнтованих та мікросервісних архітектур

При використанні СОА (service-oriented architecture – SOA) програмні компоненти, що реалізують бізнес-логіку РПС, оформляються як окремі програмні сервіси (service). Основною відмінністю сервісу від звичайного компонента ПЗ є його стандартний і платформо-незалежний інтерфейс. Застосунок-клієнт, який звертається до сервісу, не залежить від особливостей реалізації сервісу: якою мовою і в якій моделі програмування він створений, на яких апаратних засобах, в якому операційному середовищі, на якій платформі проміжного програмного забезпечення він виконується. Еволюційним розвитком СОА стала поява мікросервісної архітектури (МСА) [2], яка розглядається як набір незалежно розміщених невеликих функціональних сервісів, кожен з яких має свій процес і які спілкуються через чітко визначений протокол їх взаємодії. На рис. 1 схематично зображені особливості трьох основних типів архітектур: МА, СОА та МСА.

До переваг МСА можна віднести: швидке внесення змін; спрощене розгортання та тестування; більшу стабільність завдяки тому, що помилка в одному мікросервісі не виведе з ладу всю систему; можливість використання в одній РПС мікросервісів на різних мовах і платформах та деякі ін. [2]. Слід також зазначити, що застосування МСА є одним з ефективних методів проведення реінжинірингу успадкованих монолітних ПС, а на модельно-технологічному рівні вона пов'язана з такими сучасними напрямками досліджень як розробка адаптивних ПС та забезпечення властивостей варіабельності ПЗ в лінійках програмних продуктів [4, 5].

В той же час розробка РПС на основі МСА має такі проблеми як додаткові витрати на обмін даними між мікросервісами, для чого потрібен правильний вибір протоколів спілкування між компонентами; необхідність забезпечити незалежність одного мікросервіса від можливого збою в іншому компоненті; необхідність забезпечення узгодженості функціонування окремих мікросервісів та деякі інші [2].

© ЗІНОВ'ЄВ Д.В., ТКАЧУК М.В.. ТРІЩЕНКО І.В., 2021

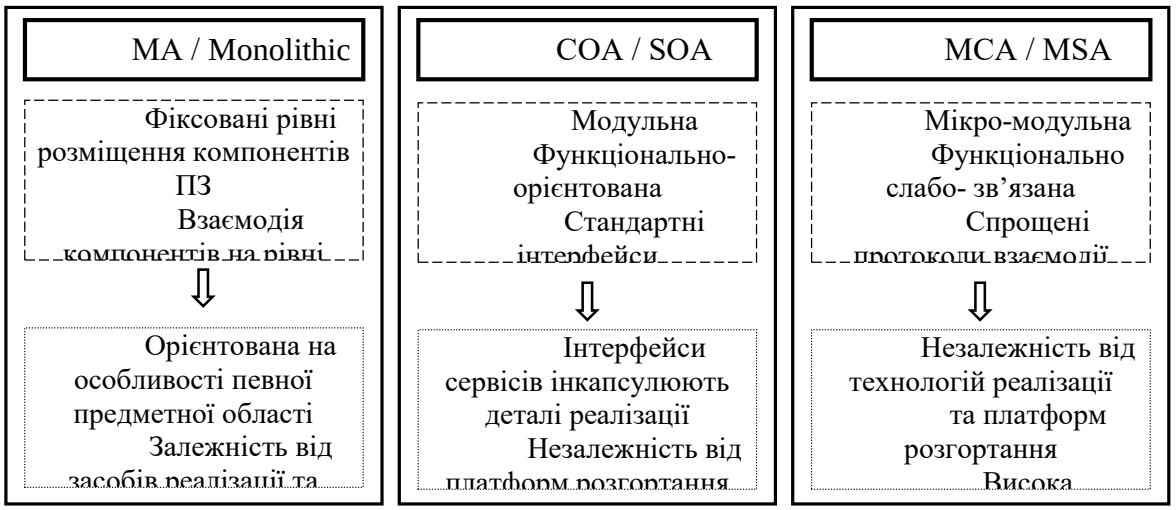


Рисунок 1 – Особливості побудови та функціонування 3-х основних типів системних архітектур

Стислий огляд існуючих підходів до оцінки якості розподілених програмних систем на основі мікросервісних архітектур

За останній період у публікаціях стосовно аналізу найбільш актуальних та перспективних напрямках досліджень у галузі МСА значна увага приділена саме питанням розробки нових підходів до оцінки якості. Так, наприклад, в [6] розглядаються такі атрибути якості МСА як *деталізація* (*granularity*) – середній розмір коду мікросервісів, *зчеплення* (*coupling*) – ступінь функціональної зв'язності між окремими мікросервісами та *щільність* (*cohesion*) – ступінь зв'язності програмного коду (напр., через виклики методів їх окремих класів) всередині певного мікросервісу. Це дозволяє, перш за все, кількісно оцінити якість архітектурних рішень у такій системі, але в той же час залишає поза увагою вплив на неї інфраструктурних чинників, напр., певних засобів контейнеризації окремих компонентів МСА.

В [7] розглядається підхід до оцінки якості МСА, у якому застосовується такий агрегований показник якості МСА як *супроводжуванність* (*maintainability*), який поєднує в собі метрики для окремих показників якості: *складність, зчеплення* та *щільність*. Цікавим рішенням тут є запропонована авторами нечітка модель для оцінки цього узагальненого показника шляхом визначення системи лінгвістичних змінних, напр.,: *Супроводжуванність* = «*Висока*», якщо *Зчеплення* = «*Низьке*» і *Щільність* = «*Висока*», або якщо *Складність* = «*Низька*» і *Зчеплення* = «*Низьке*» і та інш. Але цей підхід також дозволяє оцінити лише якість статичних аспектів відповідної МСА, і за висновками самих авторів, не враховує показників якості РПС на етапах її виконання (run-time aspects).

Ще один підхід [8] на основі обчислення кількісних метрик, а саме: *зчеплення між мікросервісами, кількість класів* в окремому мікросервісі (classes per microservices) і *частота викликів* (*frequency of external calls*) окремих мікросервісів дозволяє розробити алгоритм до проведення ефективної декомпозиції наявного монолітного застосунку та побудову відповідної цільової МСА. В [9] наведено перелік значної кількості атрибутів для опису як статичних так і динамічних аспектів якості МСА, але нажаль, для їх конкретного визначення не запропоновані конкретні кількісні метрики. У [10-13] представлені лише приклади реалізації конкретних ПС з визначенням їх окремих показників якості, але при цьому практично не розглядаються питання розробки загального підходу до визначення комплексних показників якості та МСА.

Підсумовуючи результати стислого аналізу вищенаведених досліджень [6-13], можливо зробити висновок, що для більшості існуючих підходів до оцінки якості РПС на основі МСА притаманними є наступні недоліки:

- спрощений підхід до розгляду та моделювання архітектурних особливостей побудови МСА та інфраструктурних вимог, які є необхідними до їх якісного функціонування: напр., практично відсутні дослідження впливу на якість МСА такого суттєвого чиннику як можливість застосування різних механізмів контейнеризації мікросервісів (таких як Docker, Vagrant та ін.) [14];

- недостатньо уваги приділено розробці кількісних метрик та відповідних експериментальних методик для комплексного визначення показників якості РПС на основі МСА;
- не досліджується вплив на рівень якості таких важливих характеристик РПС на основі МСА як їх загальна функціональна складність та ступінь деталізації окремих мікросервісів.

Для усунення цих недоліків потрібно запропонувати перспективні підходи до дослідження проблем якості МСА, які стисло представлені нижче.

Перспективні напрямки дослідження проблем забезпечення якості розподілених програмних систем на основі мікросервісних архітектур

Враховуючі наведені вище певні обмеження та недоліки у підходах до визначення показників якості МСА, можливо запропонувати наступні напрямки перспективних досліджень:

1) розробка моделей та методів побудови багатовимірного інформаційного простору для структурування чинників впливу та параметрів якості РПС з МСА: «*Функціональна складність МСА » / «Метрики якості МСА» / «Рівень вимог до якості РПС)»;*
2) розробка сукупності алгоритмічних моделей [15] для процесу оцінки якості МСА з застосуванням методів експертних оцінок для метрик якості МСА;
3) створення комплексної методики експериментальної оцінки якості РПС на основі МСА із застосуванням ресурсів багатовимірного інформаційного простору та алгоритмічних моделей з виразами (1) та (2).

Практична реалізація запропонованого підходу також передбачає розробку інтелектуальних інструментальних CASE-засобу для проведення обчислювальних експериментів та аналізу їх результатів.

Висновки

Обґрунтована актуальність вирішення проблеми забезпечення відповідного рівня якості при розробці та супроводі сервіс-орієнтованих розподілених програмних систем, зокрема, з мікросервісною архітектурою (МСА). На основі огляду сучасних джерел по моделям і технологіям інженерії якості МСА зроблено мотивований висновок щодо необхідності пошуку нових підходів до комплексного вирішення цих проблем. Запропоновано такі напрямки перспективних досліджень як побудова концепції багатовимірного інформаційного простору та розробка сукупності алгоритмічних моделей і метрик для експертної оцінки якості МСА.

ЛІТЕРАТУРА

1. Фаулер М. Архитектура корпоративных программных приложений – М., Изд. дом Вильямс, 2006. – 544 с.
2. Ньюмен С. Создание микросервисов. – СПб.: Питер, 2016. – 304 с.
3. Андон Ф.И., Коваль Г.И., Коротун Т.М. и др. Основы инженерии качества программных систем. – 2-е изд. - К.: «Академпериодика», 2007. – 632с.
4. Nabor C. Mendonca, Pooyan Jamshidi, David Garlan. Developing Self-Adaptive Microservice Systems: Challenges and Directions. *IEEE Software*, Vol. 38 (Issue 2), 2019. - pp. 70-79.
5. Hamzehloui, M., Sahibuddin, S. and Ashabi, A. A Study on the Most Prominent Areas of Research in Microservices. *International Journal of Machine Learning and Computing*, Vol. 9, No. 2, 2019. - pp. 242-247.
6. Michel Cojocaru;Alexandru Uta;Ana-Maria Oprescu. MicroValid: A Validation Framework for Automatically Decomposed Microservices. *IEEE International Conference on Cloud Computing Technology and Science (CloudCom - 2019)*, 2019. - pp. 78 – 86.
7. Cardarelli M., Iovino L., Francesco P., Salle A., Malavolta I., Lago P. An Extensible Data-Driven Approach for Evaluating the Quality of Microservice Architectures. *34th ACM/SIGAPP Symposium on Applied Computing*. 2019. - pp. 1225–1234. DOI: 10.1145/3297280.3297400. URL: https: //doi.org/10.1145/3297280.3297400 (Last accessed: 10.04.2021).
8. Davide Taibi, Kari Systa. A Decomposition and Metric-Based Evaluation Framework for Microservices. *International Conference on Cloud Computing and Services Science, (CLOSER-2019)*, Greece, May 2–4, 2019. - pp 133-149.

9. Michel Cojocaru, Ana-Maria Oprescu, Alexandru Uta. Attributes Assessing the Quality of Microservices Automatically Decomposed from Monolithic Applications. *18th International Symposium on Parallel and Distributed Computing (ISPDC-2019)*, Amsterdam, Netherlands, 2019. - pp. 84 – 93.
10. Приходченко С., Родна К., Поштак Р. Обґрунтування вибору мікросервісної архітектури в порівнянні з монолітною. *Проблеми використання інформаційних технологій в освіті, науці та промисловості*. 2018. № 3. С. 70 – 73.
URL: http://ir.nmu.org.ua/handle/123456789/153177 (Останній перегляд: 10.04.2021).
11. Марценюк В., Андрущак І. Про розробку інформаційної системи для медичного рятівництва на основі мікросервісної архітектури. *Медична інформатика та інженерія.* 2017. Вип. 1. DOI: 10.11603/mie.1996-1960.2017.1.7667. URL: https://doi.org/10.11603/mie.1996-1960.2017.1.7667 (Останній перегляд: 10.04.2021).
12. Бондаренко Н. Побудова системи моніторингу та керування станом навколишнього середовища на основі технології IoT. *Інформаційні технології та комп'ютерне моделювання.* 2017. С. 65–68. URL: http://itcm.comp-sc.if.ua/2017/Bondarenko.pdf (Останній перегляд: 10.04.2021).
13. Тульчинський В., Лавренюк С., Роганов В. та ін. Фактори продуктивності застосування моделей штучного інтелекту у хмарі з використанням GPU. *Cybernetics and Computer Technologies*. 2020. Вип. 1. С. 74–82. DOI: 10.34229/2707-451X.20.1.8. URL: https://doi.org/10.34229/2707-451X.20.1.8 (Останній перегляд: 10.04.2021).
14. Docker і технологія контейнерів Linux [Electronic resource]. URL: https://vps.ua/blog/docker-and-linux-containers/ (Останній перегляд: 10.04.2021).
15. Ткачук М.В., Векшин О.В., Місюта Д.М. Алгоритмічна модель адаптивного управління ресурсами мобільної системи доповненої реальності. *Зб. наук. Праць «Системи обробки інформації», ХУПС ім. Івана Кожедуба.* – Харків, – 2015. - № 12 (137), с.151- 155.

ЗІНОВ'ЄВ Дмитро Володимирович – старший викладач кафедри моделювання систем і технологій факультету комп'ютерних наук, Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, 61022; zinoviev@karazin.ua, ORCID: https://orcid.org/0000-0003-1862-9803
Наукові інтереси:
– знання-орієнтовані моделі та методи розробки та інженерії якості програмного забезпечення, проєктування розподілених інформаційних систем.

ТКАЧУК Микола Вячеславович – д.т.н., професор, завідувач кафедри моделювання систем і технологій факультету комп'ютерних наук, Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, 61022; mykola.tkachuk@karazin.com, ORCID: https://orcid.org/0000-0003-0852-1081
Наукові інтереси:
– концептуальне моделювання інформаційних систем і технологій, проєктування проблемно-орієнованих програмних систем, знання-орієнтовані моделі і методи розробки та інженерії якості програмного забезпечення.

ТРІЩЕНКО Ілля Володимирович – студент кафедри моделювання систем і технологій факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, м. Харків, Майдан Свободи 6, 61022; e-mail: illia.trishchenko@gmail.com, ORCID: https://orcid.org/0000-0002-6694-4982
Наукові інтереси:
– об'єктно-орієнтоване програмування, розробка Web – базованих застосунків, моделі і методи інженерії якості програмного забезпечення.

УДК 004.89 **СЕКЦІЯ 3,7**

КАМИШНІК Д.О, СПОРОВ О.Є, ГУЩИН І.В.

АНАЛІЗ СИСТЕМ ГЕНЕРАЦІЇ ТЕКСТУ НА БАЗІ НАУКОВИХ ТЕЗИСІВ

Вступ

Обробка текстів на природній мові (Natural-language processing, NLP) – є одним із пріоритетних напрямків розвитку штучного інтелекту. В межах цього підходу розглядаються як задачі аналізу існуючих текстів, так і задача генерації текстів, що мають сенс. Метою даної роботи є аналіз існуючих архітектур систем генерації тексту, що можуть бути використані для розв'язку задачі автоматичного створення наукових статей на основі наданих тезисів, визначення їх особливостей, дослідження переваг та недоліків.

На першому етапі дослідження необхідно визначитись з тематикою текстів та мовою тезисів та статей. На початку роботи оберемо англійську мову; передбачається, що такий вибір зробить значно простішим генерацію тексту через відсутність відмінків, родів, закінчень – слово приймає значно менше форм. Це дозволить отримувати більшу точність. Крім того, на англійській мові доступно значно більше наборів даних. В дослідження будуть використані статті на гуманітарні та філософські теми. Статті в галузі математики, техніки та фізики значно складніші для аналізу та генерації через наявність у статтях значної кількості формул, графіків та таблиць, що є невід'ємною складовою тексту.

Доволі поширеним підходом, що використовується для генерації текстів, є рекурентні нейронні мережі (далі РНМ) [1-4], їх також називають нейронними мережами з пам'яттю. РНМ мають різновиди(архітектури), що застосовуються в задачах обробки природної мови, такі як: LSTM, GPT-2, GRU, рекурсивні нейронні мережі тощо. Явною перевагою усіх архітектур РНМ є можливість обробляти текст різної довжини, враховувати порядок слів та попередню інформацію.

Рекурентні нейронні мережі

Вихід нейронної мережі можна представити як функцію від вхідних даних та вихідних даних на попередньому кроці:

$$\mathrm{h_t} = \mathrm{f}(\mathrm{h_{t-1}}, \mathrm{x_t}) \quad (1)$$

де h_t – вихід поточного кроку;
x_t – вхід поточного кроку;
h_{t-1} – вихід попереднього кроку.

Недоліками такої мережі є:
1. Загалом погана здатність працювати з великими послідовностями.
2. Накопичення помилок, вибухові градієнти.
3. Зникаючі градієнти.

LSTM

Довга короткочасна пам'ять(long short-term memory, LSTM) є більш досконалою архітектурою РНМ. LSTM може забувати пам'ять та визначає на якій пам'яті треба фокусуватися на поточному кроці, а яку треба просто зберігати.

Вихід нейронної мережі залежить від вентилів(gates) забуття, зберігання та фокусування:

$$\begin{cases} remember = f_1(W_r x_t + U_r m_{t-1}) \\ save = f_2(W_s x_t + U_s m_{t-1}) \\ focus = f_3(W_f x_t + U_f m_{t-1}) \end{cases} \quad (2)$$

де remember – вентиль забуття;

© КАМИШНІК Д.О, СПОРОВ О.Є, ГУЩИН І.В., 2021

$W_r, W_s, W_f,\ U_r, U_s, U_f$ – матриці параметрів;
x_t – вхідні дані;
save – вентиль додавання, або вхідний вентиль;
focus – вентиль уваги, або вихідний вентиль;
m_{t-1} – пам'ять на попередньому кроці;
f_1, f_2, f_3 – деякі функції активації, зазвичай це tanh.

Після розрахунку вентилів знаходиться нова довгочасна пам'ять та нова робоча(короткочасна) пам'ять:

$$\begin{cases} l_t = remember * l_{t-1} + save * f_4(W_l x_t + U_l m_{t-1}) \\ m_t = focus * f_5(l_t) \end{cases} \quad (3)$$

де l_t – довгочасна пам'ять на поточному кроці;
l_{t-1} – довгочасна пам'ять на попередньому кроці;
$f_4,\ f_5$ – функції активації;
m_t – робоча пам'ять на поточному кроці;
m_{t-1} – робоча пам'ять на попередньому кроці;
W_l, U_l – матриці параметрів.

Переваги:
1. Зменшення впливу вибухових градієнтів.
2. Зменшення впливу зникаючих градієнтів.
3. Більша швидкість навчання ніж у РНМ якщо оцінювати швидкість навчання з точки зору кількості епох.
4. Можливість контролювати відображення пам'яті[6].
Недоліки:
1. Більші затрати оперативної пам'яті.
2. Більша складність обчислень.
3. Проблема зникаючих градієнтів все ще присутня.
4. Є потреба зберігати попередню інформацію довше.
5. Схильні до перенавчання[5].
Архітектура має досить велику кількість модифікацій, в статті [1] описано застосування TAV-LSTM.

GRU

GRU (Gated recurrent units) дещо схожі до LSTM, але мають лише два вентилі – сбросу та оновлення:

$$\begin{cases} z_t = f_1(W_z x_t + U_z h_{t-1}) \\ r_t = f_2(W_r x_t + U_r h_{t-1}) \\ h_t = z_t \odot h_{t-1} + (1 - z_t) \odot f_3\big(W_h x_t + U_h(r_t \odot h_{t-1})\big) \end{cases} \quad (4)$$

де z_t – вентиль оновлення;
r_t – вентиль сбросу;
f_1, f_2, f_3 – функції активації;
$W_z, W_r, W_h, U_z, U_r, U_h$ – матриці параметрів;
x_t – вхідні дані на поточному кроці;
$\odot$ – добуток Адамара;
h_t – вихід на поточному кроці;
h_{t-1} – вихід на попередньому кроці.

Переваги:
1. Менша обчислювальна складність, якщо порівнювати з LSTM.
2. Простота, якщо порівнювати з LSTM.
3. Зменшення впливу зникаючих градієнтів.
4. Краща продуктивність на менших датасетах, що містять менш повторювані дані[6].
Недоліки:
1. Є потреба зберігати попередню інформацію довше. Поступається в здатності зберігання інформації LSTM.
2. Зникаючі градієнти все ще присутні.

Порівняння архітектур

Класичні РНМ суттєво поступаються їх модифікаціям – таким як LSTM та GRU. Основною причиною є вплив зникаючих градієнтів. Вибухові градієнти можуть бути уникнені, на відміну від зникаючих.

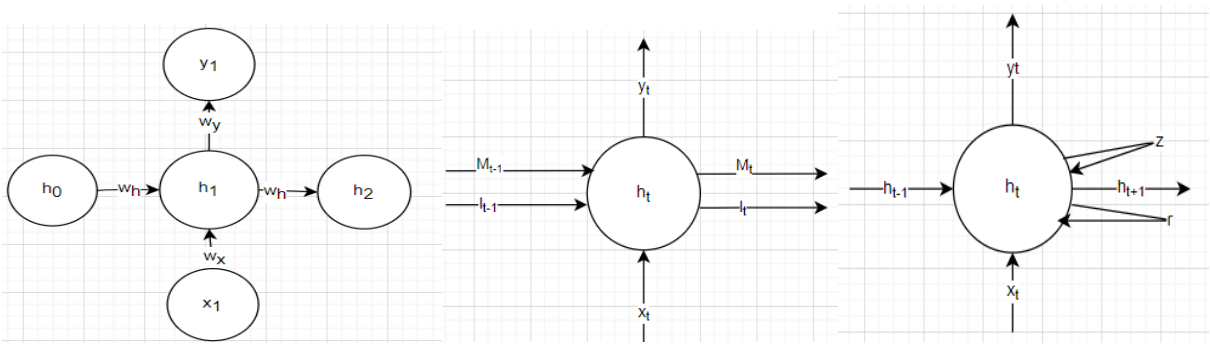


Рис.1 Рекурентна нейронна мережа

LSTM виникла у 1997 році, GRU була створена у 2014 році. Обидві застовуються досить часто та в деяких випадках є взаємозамінними.

GRU та LSTM загалом показують дуже добре результати і фактор вибухових градієнтів впливає менше ніж у звичайних РНМ, хоча він є також присутнім. GRU простіше за LSTM та має меншу обчислювальну складність, краще підходить для менших датасетів, менше схильна до перенавчання. LSTM має досить суттєві переваги – здатність довгочасного зберігання та можливість вибіркового зберігання.

Обидві архітектури показують значно кращі результати за класичні РНМ, хоча у випадках невеликих послідовностей РНМ може бути вигіднішою з точку зору простоти і давати кращі результати. При збільшенні складності GRU та LTSM є більш бажаними архітектурами для задач генерації тексту.

Висновки

В роботі проведено аналіз існуючих систем, що можуть бути використані для розв’язку задачі генерації тексту на базі наданих тезисів. Основну увагу приділено рекурентним нейронним мережам, розглянуто їх особливості та ті основні архітектури, що використовуються для генерації тексту. Проведено аналіз архітектур LSTM, GRU, наведено принципи їх роботи, переваги та недоліки, відміни від класичних архітектур рекурентних нейронних мереж. Зроблено висновок щодо їх можливостей та умов застосування.

ЛІТЕРАТУРА

1. Xiaocheng Feng, Ming Liu, Jiahao Liu, Bing Qin, Yibo Sun, Ting Liu. Topic-to-Essay Generation with Neural Networks.

URL: http://ir.hit.edu.cn/~xcfeng/xiaocheng%20Feng's%20Homepage_files/final-topic-essay-generation.pdf (Last accessed: 11.04.2021).

2. Tarasov D. S. Natural Language Generation, Paraphrasing and Summarization of User Reviews with Recurrent Neural Networks.

URL: http://www.meanotek.ru/files/TarasovDS(2)2015-Dialogue.pdf (Last accessed: 11.04.2021).

3. Alec Radford, Jeffrey Wu , Rewon Child, David Luan, Dario Amodei, Ilya Sutskever. Language Models are Unsupervised Multitask Learners.

URL:https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (Last accessed: 11.04.2021).

4. James Martens, Geoffrey Hinton, Ilya Sutskever. Generating Text with Recurrent Neural Networks.

URL: https://icml.cc/Conferences/2011/papers/524_icmlpaper.pdf (Last accessed: 11.04.2021).

5. Andrej Karpathy, Justin Johnson, Li Fei-Fe. Visualizing and Understanding Recurent Networks.

URL: https://arxiv.org/pdf/1506.02078 (Last accessed: 11.04.2021).

6. Junyoung Chung Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling

URL: https://arxiv.org/pdf/1412.3555v1.pdf (Last accessed: 11.04.2021).

КАМИШНІК Денис Олегович – студент кафедри моделювання систем на технологій; ХНУ імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: kamyshnik2020zks51@student.karazin.ua; ORCID: 0000-0002-3036-1080.

Наукові інтереси:

– *розробка програмного забезпечення в області аналізу даних, штучній інтелект.*

ГУЩІН Іван Валерійович – старший викладач кафедри штучного інтелекту і програмного забезпечення; ХНУ імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: gushchin.iv@gmail.com; ORCID: 0000-0002-1917-716X.

Наукові інтереси:

– *розробка програмного забезпечення в області аналізу даних, комп'ютерне моделювання фізичних процесів.*

СПОРОВ Олександр Євгенович – кандидат фізико-математичних наук, доцент кафедри штучного інтелекту та програмного забезпечення; ХНУ імені В.Н. Каразіна, майдан Свободи, 6, ауд. 5-20, Тел. 707-50-76, м. Харків, Україна, 61022; e-mail: sporov@karazin.ua; ORCID: 0000-0002-4610-9656.

Наукові інтереси:

– *штучний інтелект, розподілені обчислення, операційні системи, комп'ютерне моделювання фізичних процесів.*

УДК 004.9 **СЕКЦІЯ 7**

КАПЛУН В.М., ТОЛСТОЛУЗЬКА О.Г.

РОЗРОБКА КОМП'ЮТЕРНОЇ МОДЕЛІ АВТОМАТИЗОВАНОГО РОБОЧОГО МІСЦЯ ПСИХОЛОГА

ІТ фахівцями сфера розробки веб-додатків для психологів та соціальних працівниками як замовників не сильно розвинена. Це обумовлено тим що в період до пандемії COVID-19 психологам для надання допомоги або консультацій пацієнтам було достатньо особистого спілкування. Після введення карантинних норм робота психологів значно ускладнилася, оскільки був накладений ряд обмежень. У той час поки лікарі по всьому світу борються з коронавірусом, потреба пацієнтів, з психічними та іншими захворюваннями, в медичної допомозі не знижується. Зростає попит у тих людей, які раніше не так активно ходили до лікарів - ситуація спонукає турбуватися про своє здоров'я.

Дану проблему вирішують програмісти різних напрямків. Ними надаються послуги розробки веб-сервісів, які створюють можливість психологу проводити онлайн відеоконференції з використанням необхідного інструментар ію для взаємодії онлайн з пацієнтом, на базі оригінального програмно-реалізованого рішення, при цьому дотримуючись усіх карантинні норми.

Деякі ІТ фахівці на цій хвилі залучають користувачів в нові сервіси. За час пандемії психологи, соціальні працівники та пацієнти зможуть адаптуватися до цифрового консультування і не будуть сприймати його як непотрібне нововведення. Коли пандемія пройде, всі ці користувачі вже навряд чи підуть в повний офлайн.

Для утримання клієнтів при поліпшенні ситуації з карантином ІТ фахівцям необхідно зуміти перелаштуватися під час підвищеного попиту. А саме: запропонувати нові онлайн-продукти замість недоступних під час кризи, дослідити звички користувачів, адаптувати консультації, автоматизувати процеси техпідтримки або збільшити пропускну здатність сервісів.

Загальна діаграма «Комп'ютерної моделі автоматизованого робочого місця психолога» зображена на Рис 1.

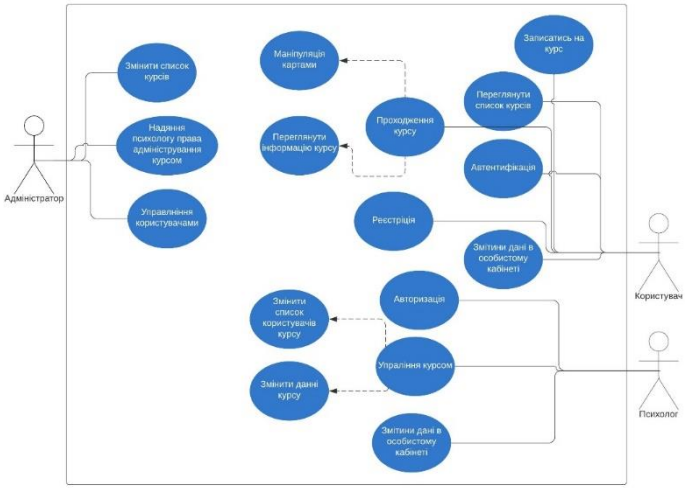


Рис. 1. Діаграма варіантів використання (UseCase diagram) "Комп'ютерної моделі автоматизованого робочого місця психолога"

© КАПЛУН В.М., ТОЛСТОЛУЗЬКА О.Г., 2021

Рекомендації по використанню "Комп'ютерної моделі автоматизованого робочого місця психолога".

Реєстрація, вхід та відновлення облікового запису користувача

Реєстрація користувача на сайті здійснюється введенням поштової адреси, пароля та імені. Для появи форми реєстрації необхідно натиснути на посилання «Увійти» з правого боку у верхній частині сайту (в шапці сайту). Після натиснути на посилання «Реєстрація», потім після заповнення форми реєстрації та натиснувши на кнопку «Реєстрація», за умови введення вірного повторного пароля, користувач реєструється в системі. На рис.1 зображено інтерфейс реєстрації користувача.

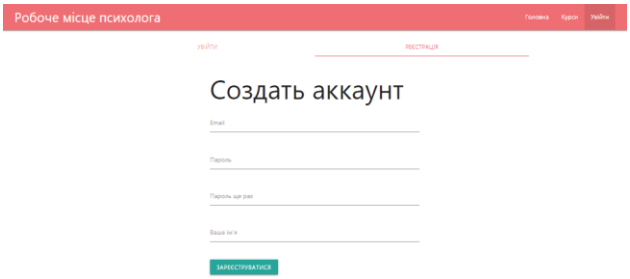


Рис. 1 "Інтерфейс реєстрації користувача".

Для появи форми входу необхідно натиснути на посилання «Увійти» з правого боку у верхній частині сайту (в шапці сайту). Після натиснути на посилання «Увійти», потім після заповнення форми реєстрації та натиснувши на кнопку «Увійти», за умови введення вірних поштової адреси пароля, користувач авторизується на сайті і може використовувати меню сервісу. На рис. 2 зображено інтерфейс входу користувача в обліковій запис.

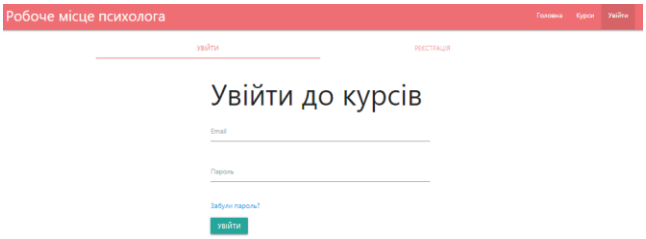


Рис. 2 "Інтерфейс входу користувача в обліковий запис *".*

Якщо користувач не може увійти в свій обліковий запис у нього є можливість відновити пароль. Натиснувши на посилання «забули пароль?» користувач переходить на сторінку відновлення пароля. На рис. 3 зображено інтерфейс відновлення пароля.

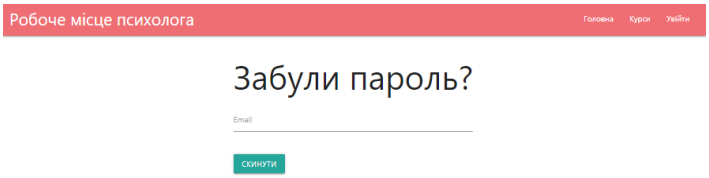


*Рис. 3 "Інтерфейс відновлення пароля".*Меню зареєстрованого користувача

Для зареєстрованого користувача доступні наступні пункти меню: головна, курси, особистий кабінет, кошик, вихід. Докладніше про всі пункти меню, буде описано нижче.

Головна сторінка містить всю основну інформацію даного курсу.

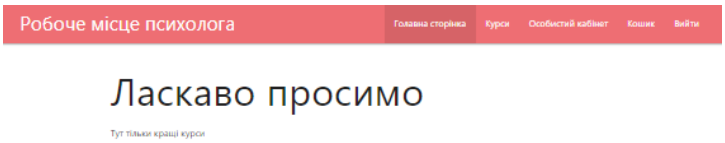


Рис. 5 "Голова сторінка веб сайту".

Сторінка курсів містить список доступних курсів. Натиснувши на посилання "Відкрити курс" користувач має можливість ознайомитися с інформацією курсу та маніпулювати інструментами даного курсу при у мові що йому надано дозвіл до даного курсу. На рис. 6 зображено список доступних курсів.

Рис. 4 "Сторінка перегляду доступних курсів".

На сторінці *"Особистий кабінет"* користувач має змогу переглянути особисті дані. Також є можливість завантажити зображення та змінити Ім'я.

ЛІТЕРАТУРА

1. HTML & CSS, and JavaScript & JQuery (2 book set) by Jon Duckett.
2. Front-End Developer Handbook — 2019.
3. The Pragmatic Programmer: Your Journey to Mastery.
4. Don't Make Me Think: A Common Sense Approach to Web and Mobile Usability.
5. Learn JavaScript VISUALLY with Interactive Exercises.

КАПЛУН Владислав Миколайович – кафедра прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: xa11867768@student.karazin.ua; ORCID 0000-0001-5787-4786.

Наукові інтереси:

- *технології створення та супроводження веб-сайтів;*

ТОЛСТОЛУЗЬКА Олена Геннадіївна – д. т. н., с. н. с. ; Харківський національний університет імені. В.Н. Каразіна; м. Харків, майдан. Свободи, 6, 61022; e-mail: elena.tolstoluzka@karazin.ua, ORCID 0000-0001-2741-180.

Наукові інтереси:

- *технологія паралельного розподіленого програмування.*

УДК 004.056.55 **СЕКЦІЯ 5**

КАПТЬОЛ Є.Ю., КУЗНЕЦОВА К.О.

АНАЛІЗ ПОСТКВАНТОВОГО АЛГОРИТМУ ЕЛЕКТРОННОГО ПІДПИСУ RAINBOW ТА ПОСТКВАНТОВИХ АТАК НА НЬОГО

Вступ

Нині, через розвиток в напрямку розробки квантових комп'ютерів, здатних на здійснення квантового криптоаналізу, постає проблема необхідності представлення та стандартизації нових постквантових електронних підписів. У зв'язку зі зростаючим інтересом до квантового криптоаналізу, в межах конкурсу NIST PQC було проведено відбір, за яким до третього раунду було допущено 3 електронних підписи, що претендують на використання в постквантовий період. Згідно з [1,2] це: CRYSTAL-DILITHIUM, FALCON, Rainbow. Так як Rainbow є одним з фіналістів, варто розглянути атаки на нього, зокрема ті, що використовують переваги квантового комп'ютера. Разом зі звичайним алгоритмом Rainbow було також представлено CZ-Rainbow та стислий алгоритм Rainbow.

1. Сутність алгоритму Rainbow

Згідно з [3], Генерація та верифікація підпису за алгоритмом Rainbow мають вигляд, наведений далі.

Генерація підпису. Маємо документ d, що необхідно підписати, використовується хеш-функція $\mathrm{H}:\{0,1\}\to\mathrm{F}^m$ для обчислення хеш-значення $\mathrm{h}=\mathrm{H}(d)\in\mathrm{F}^m$. Далі підпис $\mathrm{z}\in\mathrm{F}^n$ документа d обчислюється наступними послідовними кроками: обчислюється $\mathrm{x}=\mathrm{S}^{-1}(\mathrm{h})\in\mathrm{F}^m$; обчислюється прообраз $\mathrm{y}\in\mathrm{F}^n$ від x над центральною мапою F; обчислюється підпис $\mathrm{z}\in\mathrm{F}^n$ з $\mathrm{z}=\mathrm{T}^{-1}(\mathrm{y})$.

Верифікація підпису. Маємо документ d, та підпис $\mathrm{z}\in\mathrm{F}^n$, справжність підпису перевіряється наступними кроками: використовується хеш-функція H для обчислення хеш-значення $\mathrm{h}=\mathrm{H}(d)\in\mathrm{F}^m$; обчислюється $\mathrm{h}'=\mathrm{P}(\mathrm{z})\in\mathrm{F}^m$. Якщо виконується рівність $\mathrm{h}'=\mathrm{h}$, підпис z приймається, в іншому випадку він відхиляється.

2. Набори параметрів Rainbow

Спочатку потрібно розглянути параметри електронного підпису. У зв'язку з тим, що на конкурс NIST PQC було представлено 3 варіанти схеми електронного підпису та кожен з них має свої набори параметрів для різних категорій безпеки, їх варто розглядати окремо. Згідно з [3], було представлено такі варіанти схеми електронного підпису Rainbow: звичайний алгоритм Rainbow, CZ-Rainbow та стислий алгоритм Rainbow.

Набори параметрів для зазначених варіантів електронного підпису відповідають різним категоріям безпеки NIST. Так, кожен варіант Rainbow має 3 набори параметрів: I, III та V. Відповідність наборів параметрів категоріям безпеки NIST наступна [3]: набір параметрів I відповідає категоріям безпеки I та II, набір параметрів III відповідає категоріям безпеки III та IV, набір параметрів V відповідає категорії безпеки V.

Таблиці 1, 2 та 3 містять набори параметрів для звичайного алгоритму Rainbow, CZ-Rainbow та стислого алгоритму Rainbow відповідно [3] (Так ми маємо параметри (F, v_1, o_1, o_2), розміри відкритого та закритого ключів, розмір гешу та розмір підпису).

Табл. 1 Розміри ключів та підписів для звичайного алгоритму Rainbow

Набір параметрів	Параметри (F, v_1, o_1, o_2)	Відкритий ключ (кВ)	Закритий ключ (кВ)	Розмір гешу (біт)	Розмір підпису (біт)
I	(GF(16), 36, 32, 32)	157.8	101.2	256	528
III	(GF(256), 68, 32, 36)	861.4	611.3	576	1,312
V	(GF(256), 96,36, 64)	1,885.4	1,375.7	768	1,696

© КАПТЬОЛ Є.Ю., КУЗНЕЦОВА К.О., 2021

Таблиця 2 Розміри ключів та підписів для CZ-Rainbow

Набір параметрів	Параметри (F, v_1, o_1, o_2)	Відкритий ключ (кВ)	Закритий ключ (кВ)	Розмір гешу (біт)	Розмір підпису (біт)
I	(GF(16), 36, 32, 32)	58.8	101.2	256	528
III	(GF(256), 68, 32, 48)	258.4	611.3	576	1,312
V	(GF(256), 96,36, 64)	523.6	1,375.7	768	1,696

Таблиця 3 Розміри ключів та підписів для стислого алгоритму Rainbow

Набір параметрів	Параметри (F, v_1, o_1, o_2)	Відкритий ключ (кВ)	Закритий ключ (кВ)	Розмір гешу (біт)	Розмір підпису (біт)
I	(GF(16), 36, 32, 32)	58.8	0.06	256	528
III	(GF(256), 68, 32, 48)	258.4	0.06	576	1,312
V	(GF(256), 92,36, 64)	523.6	0.06	768	1,696

3. Атаки на Rainbow

Згідно з [3], всі відомі атаки, що можуть бути застосовані до Rainbow, є класичними. Проте також слід зазначити, що деякі з них можуть використовувати переваги, що надаються квантовим комп'ютером, шляхом застосування алгоритму Гровера до деяких кроків атаки. До атак на Rainbow можна віднести [3]: атаки знаходження колізій на геш-функції, прямі атаки, атаки MinRank, атаки HighRank, атаки "Rainbow-Band-Separation" (RBS), атаки UOV, атаки на диференціальне поле та квантові атаки «грубої сили».

Серед зазначених атак є атаки, котрі використовують квантові методи (а саме метод Гровера) для пришвидшення. До них відносяться наступні атаки: прямі атаки, атаки HighRank, атаки UOV та квантові атаки «грубої сили». Розглянемо атаки, що можуть використовувати квантові методи.

Прямі атаки. Найбільш прямолінійною атакою на Rainbow вважається пряма алгебраїчна атака, де рівняння $\mathrm{P}(\mathbf{z}) = \mathbf{h}$ розглядається як приклад задачі MQ. У зв'язку з особливостями Rainbow найефективнішим вважається отримання визначеної системи, що має рівно одне рішення, шляхом фіксації $n-m$ змінних перед застосуванням такого алгоритму подібного до алгоритму обчислення базисів Гробнера (F_4) як в [4]. Також використовують «гібридний підхід», що передбачає відгадування додаткових змінних до вирішення системи, як це було показано в [5]. В [3] складність вирішення такої системи оцінюється як кількість множень у полі, що наведена у формулі (1), де d_{reg} – це так звана ступінь регулярності системи.

$$\text{Complexity}_{\text{direct; classical}} = \min_k \left(q^k \cdot 3 \cdot \binom{m-k+d_{\text{reg}}}{d_{\text{reg}}} \cdot \binom{m-k}{2} \right) \quad (1)$$

Застосування квантового комп'ютера надає перевагу при використанні «гібридного підходу». В [3] складність такої атаки оцінюється як кількість множень у полі наведена у формулі (2).

$$\text{Complexity}_{\text{direct; quantum}} = \min_k \left(q^{k/2} \cdot 3 \cdot \binom{m-k+d_{\text{reg}}}{d_{\text{reg}}}^2 \cdot \binom{m-k}{2} \right) \quad (2)$$

Атаки HighRank. Згідно з [6], атака HighRank полягає у виявленні змінних, що з'являються найменшу кількість разів у центральних поліномах (вони відповідають Oil-змінним останнього рівня Rainbow).

Складність цієї класичної атаки оцінюється як кількість множень у полі, наведена в формулі (3).

$$\text{Complexity}_{\text{HighRank; classical}} = q^{O_a} \cdot \frac{n^3}{6}. \quad (3)$$

Застосування квантового комп'ютера дозволяє зменшити кількість множень у полі до кількості наведеної у формулі (4).

$$\text{Complexity}_{\text{HighRank; quantum}} = q^{O_a/2} \cdot \frac{n^3}{6} \quad (4)$$

Атаки UOV. У зв'язку з тим, що Rainbow можна розглядати як продовження добре відомої схеми підписів Oil and Vinegar [7], можна стверджувати, що її можна атакувати, з використанням всіх відомих атак UOV. Таким чином, наприклад, може бути застосована атака UOV «Oil Subspace» Кіпніса та Шаміра [8]. Окрім цього, Rainbow також можна розглядати як екземпляр UOV з $\upsilon = \upsilon_1 + o_1$ та $o = o_2$.

Мета цієї атаки заключається в пошукі попереднього зображення так званого Oil підпростору O афінного перетворення T, де $\mathrm{O} = \left\{ \mathrm{x} \in \mathrm{F}^n : x_1 = \cdots = x_\upsilon = 0 \right\}$. Знаходження цього простору дозволяє відокремити Oil від змінних Vinegar та відновити закритий ключ.

Складність цієї класичної атаки оцінюється як кількість множень у полі, наведена в формулі (5).

$$\text{Complexity}_{\text{UOV-Attack; classical}} = q^{n-2_{o_2}-1} \cdot o_2^4 \quad (5)$$

Кількість множень у полі, до якої застосування квантового комп'ютера та алгоритма Гровера зменшує складність наведено на формулі (6).

$$\text{Complexity}_{\text{UOV-Attack; classical}} = q^{\frac{n-2_{o_2}-1}{2}} \cdot o_2^4 \quad (6)$$

Квантові атаки «грубої сили». За умови застосування для криптоаналізу квантового комп'ютера, атака «грубої сили» може бути суттєво прискореною шляхом використання алгоритму Гровера. Так, наприклад, в роботі [9] було продемонстровано можливість розв'язанна системи з $m-1$ двійкових квадратних рівнянь з $n-1$ двійковими змінними, використовуючи $m+n+2$ кубітів та обчислюючи схему з $2^{n/2} \cdot \left(2m\left(n^2+2n\right)+1\right)$ квантових логічних елементів.

Разом з цим, також було продемонстровано варіант вирішення системи, що передбачає використання меншої кількості кубітів але зі збільшенням схеми таким чином, що вона має приблизно в 2 рази більше квантових логічних елементів. Так при $n = m$ може бути вирішена двійкова система з m рівнянь та з m змінними з використанням $2^{m/2} \cdot 2 \cdot m^3$ бітових операцій.

Очікується, що використання алгоритму Гровера при проведенні квантового криптоаналізу методом «губої сили» призведе до квадратичного прискорення атаки «грубої сили». Таблиці 4, 5 та 6 демонструють складність різних атак (серед яких прямі атаки, атаки HighRank та атаки UOV) на підпис для різних наборів параметрів (I, III та V відповідно)[3]. У першому рядку наведено кількість необхідних для здійснення атаки класичних логічних елементів, а у другому – кількість необхідних для здійснення атаки квантових логічних елементів.

Таблиця 4 Складність класичних та квантових атак на підпис з набором параметрів I

Набір параметрів	Параметри (F, v_1, o_1, o_2)	$\log_2$ (#гейтів)		
		прямі	HighRank	UOV
I	(GF(16), 36, 32, 32)	164	150	157
		122	86	91

Таблиця 5 Складність класичних та квантових атак на підпис з набором параметрів III

Набір параметрів	Параметри (F, v_1, o_1, o_2)	$\log_2$ (#гейтів)		
		прямі	HighRank	UOV
III	(GF(256), 68, 32, 48)	234	410	437
		200	218	233

Таблиця 6 Складність класичних та квантових атак на підпис з набором параметрів V

Набір параметрів	Параметри (F, v_1, o_1, o_2)	$\log_2$ (#гейтів)		
		прямі	HighRank	UOV
V	(GF(256), 96, 36, 64)	285	539	567
		243	283	299

Висновки

Наведені сутність генерації та верифікації електронного підпису за алгоритмом Rainbow та набори параметрів схеми електронного підпису Rainbow, що відповідають рівням безпеки NIST вказують на підготовку до застосування Rainbow у посткантовий період та протистояння окрім класичних ще й квантовим атакам.

Однак, враховуючи відомості про атаки на електронний підпис та оцінки складності атак на набори параметрів можна зробити висновок про наявність можливості реалізації атаки стосовно електронного підпису Rainbow та про можливість застосування квантового комп'ютера для зменшення складності атак.

Проте також слід зауважити, що пришвидшення атак за допомогою квантового комп'ютера є нерівномірним. Через це для різних наборів параметрів оптимальними є різні атаки. Так, наприклад для набору параметрів I найкращою атакою з використанням квантового комп'ютера є атака HighRank, а для наборів параметрів III та V – пряма атака.

Також важливим моментом є те, що повністю квантовою атакою з наведених є лише квантова атака «грубої сили». Інші атаки використовують квантовий комп'ютер лише для виконання певного кроку атаки.

ЛІТЕРАТУРА

1. Post-Quantum Cryptography PQC. Round 3 Submissions. NIST Computer Security Resource Center (CSRC). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions (last accessed on 15.02.2021).
2. PQC Standardization Process: Third Round Candidate Announcement. NIST Computer Security Resource Center (CSRC). July 22, 2020. URL: https://csrc.nist.gov/News/2020/pqc-third-round-candidate-announcement (last accessed on 13.02.2021).
3. Jintai Ding. Rainbow – Algorithm Specification and Documentation. The 3d round Proposal. Department of Mathematical Sciences, University of Cincinnati.
4. J.-C. Faugere. A new efficient algorithm for computing Grobner Bases (F4). Journal of Pure and Applied Algebra, 139 (1999) 61-88. DOI: https://doi.org/10.1016/S0022-4049(99)00005-5 (last accessed on 17.02.2021).
5. L. Bettale, J.-C. Faugere, L. Perret. Hybrid approach for solving multivariate systems over finite fields. Journal of Mathematical Cryptology, 3, pp. 177-197, 2009.
6. D. Coppersmith, J. Stern, S. Vaudenay. Attacks on the birational signature scheme. CRYPTO 1994, pp. 435-443. Springer, 1994.
7. A. Kipnis, J. Patarin, L. Goubin. Unbalanced Oil and Vinegar schemes. EUROCRYPT 1999, pp. 206-222. Springer, 1999.
8. A. Kipnis, A. Shamir. Cryptanalysis of the Oil and Vinegar signature scheme. CRYPTO 1998, pp. 257-266. Springer,1998.
9. P. Schwable, B. Westerbaan. Solving Binary MQ with Grover's Algorithm. SPACE 2016, pp. 303-322. Springer 2016.

КАПТЬОЛ Євгеній Юрійович – аспірант кафедри безпеки інформаційних систем і технологій ХНУ імені В.Н. Каразіна.

Наукові інтереси:

– криптологія, квантовий криптоаналіз, використання квантових комп'ютерів в криптології.

КУЗНЕЦОВА Катерина Олександрівна – студентка факультету комп'ютерних наук ХНУ імені В.Н. Каразіна.

Наукові інтереси:

– криптологія, електронні підписи, постквантова криптографія.

УДК 621.865.8 **СЕКЦІЯ 2**

КАПУСТА М.А.

МОБІЛЬНИЙ РОБОТ ДЛЯ РОЗПІЗНАВАННЯ ТА СПОСТЕРЕЖЕННЯ

Уявити сучасне життя без використання роботизованих механізмів і систем на виробництвах, у медичній сфері або захисті населення неможливо. Розвиток і поява нових роботів полегшують повсякденне життя і дають нові можливості.

Дане дослідження присвячене проблемі використання роботів в небезпечних місцях для зменшення ризиків травмування людини, наприклад при пошукових роботах у надзвичайних ситуаціях. Метою дослідження є розробка програмного забезпечення для віддаленого керування роботом і розпізнавання людей.

Проаналізувавши декілька роботів з схожим функціоналом можна скласти вимоги до апаратного та програмного забезпечення мобільного робота для розпізнавання образів (табл. 1). Для забезпечення системи управління та коректної роботи мобільного робота необхідні мікроконтролер, приводи, драйвери, шасі, камера та відповідне програмне забезпечення.

Табл.1 Функціональні вимоги до апаратних складових мобільного робота

Складова частина	*Перелік вимог*
Мікроконтролер	1) компактність; 2) дешевизна; 3) наявність Bluetooth або Wi-fi модуля в платі; 4) наявність оперативної пам'яті не менше 1 ГБ; 5) наявність не менше 20 GPIO; 6) низьке енергоспоживання; 7) можливість легкого налаштування веб серверу.
Приводи	1) надійність; 2) мала маса і малі габарити приводу; 3) наявність стандартного посадкового місця для зручності розміщення і монтажу; 4) висока енергоефективність; 5) доступність.
Драйвери	1) доступність; 2) можливість контролювати мінімум 2 двигуни; 3) компактність.
Шасі	1) надійність; 2) компактність; 3) зручність монтажу компонентів; 4) гусениці; 5) амортизація підвіски.
Камера	1) передача відео в якості мінімум 720 на 480 пікселів; 2) просте налаштування відеотрансляції.
Клієнт-сервер	1) підключення з будь якого пристрою, що знаходяться в одній мережі; 2) затримка передачі запитів з сервера складає від 1 до 15мс; 3) всі запити повинні мати статус запиту 200; 4) частота передачі потокового відео не менше 15 кадрів в секунду.
Алгоритм розпізнавання обличь	1) швидкість розпізнавання обличчя максимум 3 секунди; 2) алгоритм повинен працювати і при малій швидкості інтернет-з'єднання.

Також були визначені загальні вимоги до характеристик мобільного робота:

1) швидкість роботи на рівній поверхні – 0,3 метра в секунду;
2) градус підйому – 20 градусів;
3) час роботи від батареї мінімум 10 годин;
4) дальність роботи робота від роутера 20 метрів;
5) швидкість повороту вліво і вправо 0.5 секунди.

© КАПУСТА М.А., 2021

Обґрунтування та вибір основних компонентів системи

Основним мікроконтролером для керування роботом було вибрано Raspberry Pi 3b[1]. На Raspberry Pi 3B встановлений чотирьохядерний 64-х бітний процесор Broadcom BCM2837B0 на архітектурі ARMv8 Cortex-A53 з тактовою частотою 1,4 ГГц та модуль оперативної пам'яті LPDDR2 SDRAM на 1 ГБ. BCM2837B0 включає в себе також двохядерний графічний співпроцесор Video Core IV® Multimedia, який підтримує OpenGL ES 2.0, OpenVG, та здатний кодувати, декодувати і виводити Full HD відео (1080p, 30 FPS) .

В даній роботі була використана модель 3B+ – з портів встановлено HDMI вхід, слот для SD-карти, чотири порти USB 2.0, Gigabit Ethernet, GPIO (General Purpose I / O), роз'єм 3.5 для навушників. Із безпровідних інтерфейсів Raspberry Pi має Bluetooth 4.2, а також Wi-Fi 2,4 і 5ГГц. Однією з особливостей Raspberry Pi є порти GPIO (загального вводу / виводу) [2].

Для проекту використана 8-мегапіксельна камера Raspi CAM v2 базі сенсора Sony IMX219. Камера сумісна з усіма версіями Raspberry. Підтримувані формати відео – 1080p з швидкістю 30 кадрів за секунду(fps), 720p з 60 fps і 480p з 90 fps.

Робот рухається за допомогою двох двигунів з номінальною робочою напругою 6В, для контролю цих двигунів використовується драйвер LN298N [7].

Для даного проекту було використане шасі DFRobot Devastator v2.Шасі виконано із алюмінію. Однією з особливостей цього робота є амортизація підвіски, а також гусениці. На рис. 1 зображено готове шасі робота.

Рис. 1 Шасі мобільного робота

Функціональна схема та схема з'єднань системи керування

В передній частині розміщений батарейний відсік для живлення драйвера двигунів, і сам драйвер, що буде регулювати подачу струму на двигуни.

У середині системи буде розміщено два двигуни, що під'єднані до гусениць – це буде приводом платформи. Всі частини з'єднані болтами та гайками.

В задній частині знаходиться Raspberry Pi 3B+ який служить головною платою управління, а також веб сервером. На рис. 2 показана схема електричного кола робота.

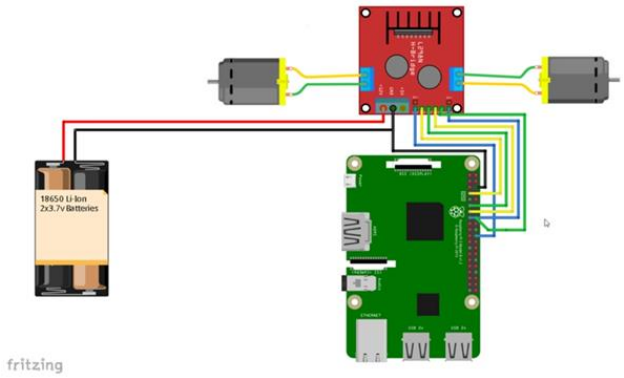


Рис. 2 Схема електричного кола робота

Для управління мобільним роботом було написано клієнт-серверне ПО. Воно представляє локальний сервер для отримання запитів та клієнт у вигляді веб-сайту з JavaScript. Сервер створюється на Raspberry Pi. А сам клієнт динамічно завантажується на сервер. Для налагодження має бути налаштований HTTP зв'язок в локальній мережі.

Для сервера було використано веб-фреймворк Flask-мікрофреймворк для створення веб-додатків [3].

Для контролю портів GPIO було використано бібліотеку Rpi.GPIO. Ця бібліотека створена для легкого контролю GPIO. Контроль реалізовано таким чином. Коли користувач натискає кнопку відправляється GET запит на сервер, де він оброблюється і на відповідний GPIO порт віддається високий сигнал. Коли користувач відпускає кнопку на сервер, де GET запит, де він опрацьовується і зупиняє подачу високого сигналу.

Швидкість регулюється за допомогою ШІМ (широтно імпульсна модуляція). У табл. 2 показаний результат подачі високого сигналу на входи драйвера двигуна.

Табл.2 Подача високого сигналу на входи драйвера двигуна

Вхід		Вихід		
1A	1B	2A	2B	Опис
0	0	0	0	Вимкнутий
0	1	0	1	Вперед
1	0	1	0	Назад
1	0	0	1	Вліво
0	1	1	0	Вправо
1	1	1	1	Вимкнутий

Фактичний напрямок залежить від того як двигуни змонтовані і підключені. Напрямок можна змінити шляхом переставлення проводів.

В даному проекті було використано алгоритм 2D розпізнавання через потокове відео за допомогою бібліотеки OpenCV [5]. OpenCV, або бібліотека з відкритим кодом Computer Vision, почалася як дослідницький проект в Intel. На даний момент це найбільша бібліотека комп'ютерного зору за рівнем кількості функцій ,які вона виконує. OpenCV містить реалізацію понад 2500 алгоритмів [6].

Для даного проекту було використано каскадний класифікатор. Етапами алгоритму Віоли і Джонса є класифікатори бустінга над деревами рішень, які використовують в якості ознак характеристики Хаара. Загальний підхід Віоли-Джонса до детектування об'єктів на зображенні комбінує чотири концепції:

1. Використання простих прямокутна функцій, так званні функцій Хаара;
2. Інтегральне представлення зображення за цими ознаками для Швидкого Виявлення Функції;
3. Використання методу побудова класифікатора на основі алгоритму адаптивного бустінга AdaBoost, що реалізує концепцію машинного навчання;
4. Застосування методу комбінування класифікаторів в каскадну структуру для ефективного поєднання множини функцій.

Саме ці ідеї дозволяють здійснювати детектування об'єкта в режимі реального часу.

Розпізнавання йде в відеопотоком, тому відео кожну секунду розбивається на кадри і кожний з них проходить цей алгоритм і навантаження на процесор максимальне [4]. Для зменшення навантаження і прискорення обробки потокове відео використовувалось в якості 720x480 пікселів, а частота кадрів не перевищувала 30. Для перевірки можливостей алгоритму було проведено декілька тестів роботи алгоритму.

Результати тестування роботи мобільного робота надані у табл. 3.

Таким чином можна зробити висновок, що використання створеного мобільного робота з розпізнаванням обличчя можливе, але потребує додатково доопрацювання для покращення його характеристик.

Табл.3 Результати тестування роботи мобільного робота

Отримані результати	Рекомендовані результати	Опис тесту
0.25м/с	0.3 м/с	Швидкість робота на рівній поверхні
30 градусів	30 градусів	Кут перешкоди на яку може піднятись робот
16 годин	10 годин	Час роботи
15 метрів	20 метрів	Дальність роботи від роутера
0,5 секунд	0,5 секунд	Швидкість повороту вліво, вправо на 180 градусів
1-12мс	1-15мс	Час передачі запитів на сервер
Від 5 до 25 секунд	15 кадрів в секунду	Частота передачі потокового відео
40%	45%	Кут розпізнавання алгоритмом обличя
Від 1 до 5 секунд	До 3 секунд	Швидкість розпізнавання
2-3 метри	5 метрів	Дистанція розпізнавання

ЛІТЕРАТУРА

1. Модельний ряд Raspberry Pi // URL: https://raspberrypi.ru/modeli_raspberry_pi (дата звернення: 10.03.2021).
2. GPIO ports в Raspberry Pi // URL: https://www.explainingcomputers.com/rasp_pi_robotics.html (дата звернення: 24.03.2021)
3. Документація Flask // URL:https://flask-russian-docs.readthedocs.io/ru/latest/(дата звернення: 13.03.2021).
4. Video translation behind Flask + OpenCV // URL: https://blog.miguelgrinberg.com/post/flask-video-streaming-revisited (дата звернення: 21.03.2020).
5. Документація OpenCV // URL: https://opencv.org/ (дата звернення: 20.05.2020).
6. How face recognition works? // URL: https://www.popmech.ru/technologies/402352-na-lice-napisano-kak-rabotaet-kompyuternoe-raspoznavanie-lic/ (дата звернення: 24.03.2020).
7. Monk S. Raspberry Pi Cookbook: Software and Hardware Problems and Solutions / S. Monk, 2019. – 660 p.

КАПУСТА Максим Анатолійович – студент кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: kapusta2020ku51@student.karazin.ua; ORCID: 0000-0001-9232-1757.

Наукові інтереси:

– *робототехніка;*

– *автоматизація та комп'ютерно-інтегровані технології.*

УДК 004.056.55 **СЕКЦІЯ 5**

КАЧКО О.Г., КАНДІЙ С.О.

ОПТИМІЗАЦІЯ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ЕП FALCON З ВИКОРИСТАННЯМ AVX512 ІНСТРУКЦІЙ

Вступ

Однією з важливих задач сучасної криптографії є розробка криптографічних перетворень, що є стійкими до атак на квантових комп'ютерах. У 2016 році NIST США оголосили про початок конкурсу NIST PQC, метою якого є створення нових постквантових криптографічних стандартів. До фінального третього етапу з електронних підписів (ЕП) вийшли Falcon, Crystals-Dilithium та Rainbow.

Спеціалісти NIST зазначають [1], що одним з основних критеріїв на третьому етапі є наявність надійної та швидкої реалізації для широкого спектру існуючих платформ. Механізм ЕП Falcon відрізняється від інших схем складністю створення програмної реалізації. Проте, завдяки використанню AVX2 інструкцій, авторам вдалося зробити надзвичайно швидку реалізацію.

Набір інструкції AVX512 у багатьох випадках дозволяє значно підвищити швидкодію програмних засобів. У даній доповіді ми розглядаємо особливості створення програмної реалізації Falcon з використанням AVX512 інструкцій.

Механізм ЕП Falcon

У механізмі ЕП Falcon [3] відкритим ключем є базис $A \in \mathbb{Z}_q^{n \times m}$ решітки Λ. У якості закритого ключа використовується редукований базис $B \in \mathbb{Z}_q^{m \times m}$ дуальної решітки $\Lambda^{\perp}$. Схема Falcon використовує NTRU решітки:

$$A = \begin{bmatrix} 1 & h \\ 0 & q \end{bmatrix}, B = \begin{bmatrix} f & g \\ F & G \end{bmatrix}, \quad (1)$$

де f, g, h, F, G є поліномами у кільці $\mathbb{Z}_q[X]/(\phi(X))$ і для них виконується рівняння

$$fG - gF = q \bmod \phi. \quad (2)$$

Якщо f, g є малими поліномами, то задача вирішення рівняння (2) стає складною. Використання структурованих решіток дає змогу зменшити розмір ключів та передавати не всю матрицю базису, а лише поліноми, що її формують.

Для вироблення підпису для повідомлення m семплується пара поліномів (s_1, s_2), для якої виконується рівняння

$$s_1 + s_2 h = H(r \| m), \quad (3)$$

де r – сіль, H – деяка криптографічна геш-функція, що відображає бінарні данні на вектори. Таких (s_1, s_2) існує багато і знайти їх не важко, проте, якщо ввести обмеження $\|s\| < \beta$, де β має достатньо мале значення, то задача стає важкою і зводиться до проблеми SIS. Таким чином, основна проблема, на якій ґрунтується безпека схеми Falcon – SIS на NTRU-решітках.

Оптимізація програмної реалізації

Оскільки у ЕП Falcon основними операціями є множення та додавання поліномів, то доцільно оптимізувати ці операції. Для оптимізації операції множення доцільно використати метод NTT. NTT можливо обчислити за $O(n \log n)$ операцій за допомогою алгоритму Кулі-Тьюкі [4]. Для зворотного перетворення відповідно за допомогою алгоритму Джентльмен-Садне. Алгоритми на кожній ітерації обчислюють відображення

© КАЧКО О.Г., КАНДІЙ С.О., 2021

$$f \mapsto (f \bmod X^{n/2} - \xi^{n/2},\ f \bmod X^{n/2} + \xi^{n/2})$$
$$\mathbf{Z}_q[X]/(X^n - \xi^n) \to \mathbf{Z}_q[X]/(X^{n/2} - \xi^{n/2}) \times \mathbf{Z}_q[X]/(X^{n/2} + \xi^{n/2}) \quad (4)$$

При цьому коефіцієнти відповідних поліномів на кожній ітерації обчислюються як

$$c'_i = c_i + \xi^{n/2} c_{n/2+i}$$
$$c''_i = c_i - \xi^{n/2} c_{n/2+i} \quad (5)$$

Через те, що коефіцієнти поліномів лежать у кільці $\mathbf{Z}_q$, під час обчислень в стандартній реалізації відбувається багато приведень за модулем q. Ця операція є дорогою. Щоб зменшити кількість її використань при множенні коефіцієнтів автори використовують редукцію Монтгомері, яка залежить від фактора r. На фазі предобчислень всі примітивні корені з одиниці заносяться до масиву. Кожен корінь помножується на фактор $r \bmod q$. Іншою оптимізацією є використання лінивої редукції: під час складань значення зростає доволі повільно, то коефіцієнти можливо не приводити за модулем у кожній ітерації, а приводити лише в кінці.

На кожній ітерації фактично масив можливо розглядати як сукупність блоків. На рис. 1 зображено таке представлення.

Рис 1. Представлення взаємодії коефіцієнтів поліномів як взаємодії блоків (кожен рівень є ітерацією алгоритму Кулі-Тьюкі)

При обробці кожен блок взаємодіє лише з одним іншим блоком, що знаходиться справа. При використанні AVX512 на перших ітераціях розмір блоків занадто великий, щоб їх повністю розмістити в регістрах zmm. Відповідно, стратегія обчислень буде відрізнятися. Кожен коефіцієнт зберігається в 32-бітній змінній. Один регістр zmm може зберігати 8 коефіцієнтів. З врахуванням всіх покращені, основне перетворення NTT матиме вигляд:

```
uint32_t x, y; x = *r1;y = *r2;
*r1 = modp_add(x, y, p);
*r2 = modp_montymul(modp_sub(x, y, p), s, p, p0i);
```

де r1, r2 є вказівниками на відповідні елементи в блоках; p – одне з простих чисел, що використовується у системі залишкових класів у якості модуля; p0i – зворотне до числа p.

При цьому вказівники r1, r2 змінюються з деяким кроком в залежності від підполя, в якому відбуваються перетворення. Це ускладнює використання векторизованих інструкцій, так як дані неможливо зчитати одним блоком. Тож, для використання векторизованих інструкцій необхідно спочатку перегрупувати дані в залежності від підполя. Операцію складання за модулем можливо реалізувати як

```
static inline __m512i modp_add_avx512(__m512i a,__m512i b,__m512i p){
  __m512i tmp1,tmp2;
  tmp1 = _mm512_add_epi32(a,b);
  tmp1 = _mm512_sub_epi32(tmp1,p);
  tmp2 = _mm512_srli_epi32(tmp1,31);
  tmp2 = _mm512_sub_epi32(_mm256_set1_epi32(0),tmp2);
  tmp2 = _mm512_and_si256(tmp2,p);
```

```
    tmp1 =  _mm512_add_epi64(tmp1,tmp2);
    return tmp1;
}
  Операцію віднімання відповідно
static inline __m512i modp_sub_avx512(__m512i a,__m512i b,__m512i p){
    __m512i tmp1,tmp2;
    tmp1 = _mm512_sub_epi32(a,b);
    tmp2 = _mm512_srli_epi32(tmp1,31);
    tmp2 = _mm512_sub_epi32(_mm256_set1_epi32(0),tmp2);
     tmp2 = _mm512_and_si256(tmp2,p);
    tmp1 = _mm512_add_epi64(tmp1,tmp2);
    return tmp1;
}
  І редукцію Монтгомері як
static inline __m512i modp_montymul_avx512(__m512i a,__m512i b,
                                __m512i p,__m512i p0i){
    __m512i tmp1,tmp2,tmp3;
    __m512i mask = _mm512_set1_epi64x(0x7FFFFFFF);
    tmp1 =_mm512_mul_epu32(a,b);
    tmp2 =_mm512_mul_epu32(tmp1,p0i);
    tmp2 =_mm512_and_si256(tmp2,mask);
    tmp2 =_mm512_mul_epu32(tmp2,p);
    tmp1 = _mm512_add_epi64(tmp1,tmp2);
    tmp1 = _mm512_srli_epi64(tmp1,31);
    tmp1 = _mm512_sub_epi32(tmp1,p);
    tmp2 = _mm512_srli_epi64(tmp1,31);
    tmp2 = _mm512_sub_epi32(_mm256_set1_epi32(0),tmp2);
    tmp2 = _mm512_and_si512(tmp2,p);
    tmp1 = _mm512_add_epi64(tmp1,tmp2);
    return tmp1;
}
```

Множення в спектральній області, відповідно до методу NTT, відбувається покомпонентно. Розглянемо загальну ситуацію, де потрібно знайти значення виразу

$$a_1 * b_1 + ... + a_l * b_l, \tag{4}$$

де $a_1, ..., a_l, b_1, ..., b_l$ є поліноми у NTT представленні (в спектральній області).

Щоб не використовувати дорогу операцію приведення за модулем, скористаємося редукцією Монтгомері: кожен елемент помножимо на фактор 2^{32} і виконаємо операцію редукції у кінці. Тож, для обчислення виразу (4) виконаємо наступні кроки:

- Оскільки вираз (4) є сумою, то необхідні регістри, що слугуватимуть у ролі акумулятора. Відведемо під цю роль регістри zmm2-zmm9.
- Нехай a та b є вказівниками типу _m512i на відповідні представлення поліномів в пам’яті. Зчитаємо з пам’яті поточні блоки a та b в регістри zmm10, zmm12, zmm14, zmm16 та zmm18, zmm20, zmm22, zmm24 відповідно.
- Помножимо на фактор 2^{32}:
  zmm11 = _mm512_srli_epi64(zmm10, 32);
  ….
  zmm25 = _mm512_srli_epi64(zmm24, 32);
- Виконаємо покомпонентне множення
  zmm10 = _mm512_mul_epu32(zmm18, zmm10);
  …
  zmm17 = _mm512_mul_epu32(zmm25, zmm17);
- Запишемо результати до обраних акумуляторів
  zmm2 = _mm512_add_epi64(zmm2, zmm10);

…
zmm9 = _mm512_add_epi64(zmm9, zmm17);

- На останньому кроці виконаємо редукцію Монтгомері над акумуляторами (в регістрах zmm0, zmm1 містяться відповідні константи для редукції.

```
zmm10 = _mm512_mul_epu32(zmm2, zmm0);
zmm10 = _mm512_mul_epu32(zmm10, zmm1);
zmm2 = _mm512_add_epi64(zmm10, zmm2);
zmm2 = _mm512_srli_epi64(zmm2, 32);
… (для інших блоків аналогічно)
```

Висновки:

Програмна реалізація ЕП Falcon, незважаючи на свою складність, гарно розпаралелюється. Це дає змогу використовувати векторизовані набори інструкції для підвищення швидкодії. У даній роботі було розглянуто застосування інструкцій AVX512 для оптимізації основної операції множення поліномів з урахуванням особливостей схеми ЕП Falcon.

ЛІТЕРАТУРА

1. Gorhan Alagic Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process. NISTIR 8309 / Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, John Kelsey, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, Ray Perlner
2. Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, Peter Schwabe, Gregor Seiler and Damien Stehlé CRYSTALS-Dilithium: Algorithm Specifications and Supporting Documentation. – Access mode: https://pq-crystals.org/dilithium/data/dilithium-specification.pdf.
3. Thomas Prest et Al. aFlcon: Fast-Fourier Lattice-basedCompact Signatures over NTRU – Access mode: https://falcon-sign.info/falcon.pdf.
4. Gregor Seiler Faster AVX2 optimized NTT multiplication for Ring-LWE lattice cryptography – Access mode: https://crypto.ethz.ch/publications/files/Seiler18.pdf.
5. Качко О.Г. Осика О.Ф. Використання SIMD команд для паралельних обчислень. Навчальний посібник з дисципліни Паралельне програмування. Харків, ХНУРЕ, 2020, 274 с.

КАЧКО Олена Григорівна – канд. техн. наук, професор, Харківський національний університет радіоелектроніки, професор кафедри програмної інженерії, начальник відділу програмування АТ «Інститут інформаційних технологій», вул. Бакуліна, 12, 61000, Харків, (050) 32-39-026, elena.kachko@nure.ua; ORCID: 0000-0001-9249-0497

Наукові інтереси:
– *Криптографія, паралельні алгоритми, оптимізація програм.*

КАНДІЙ Сергій Олегович – технік-конструктор АТ «Інститут Інформаційних технологій» вул. Бакуліна, 12, 61166, Харків, e-mail: sergeykandy@gmail.com; ORCID: 0000-0003-0552-8341.

Наукові інтереси:
– *Електронний цифровий підпис, програмування, постквантова криптографія.*

УДК 621.394 **СЕКЦІЯ 7**

КИРИЛОВА Є.О.

МАТЕМАТИЧНА МОДЕЛЬ СИСТЕМИ АВТОМАТИЗОВАНОГО РАДІОМОНІТОРИНГУ

На сьогоднішній день радіочастотний моніторинг є найефективнішим інструментом отримання об'єктивної інформації стосовно стану електромагнітної обстановки .Та, сучасний рівень радіомоніторингу не завжди встигає за стрімким розвитком сфери телекомунікацій. Упровадження новітніх радіотехнологій, які грунтуються на використанні складних видів радіосигналів, зокрема, сигналів з розширенням спектра та шумоподібних сигналів, а також освоєння вищих діапазонів частот вимагають нагального вирішення питань стосовно методів радіомоніторингу і вимірювання параметрів таких сигналів.

Постановка задачі дослідження

Метою доповіді є аналіз методів автоматичного радіомоніторингу (АР) та розпізнавання сигналів. Та на основі отриманих знань розробка моделі автоматичного радіомоніторингу. Тому що, на сьогоднішній день радіомоніторинг це найефективніший інструмент отримання об'єктивної інформації стосовно стану електромагнітної обстановки.

Вдамося трохи у визначення, щоб розібратися краще. Радіомоніторинг - комплекс організаційних і технічних заходів щодо збирання, оброблення, аналізу та збереження даних про параметри та характеристики сигналів радіовипромінювання РЕЗ, ВП та інших ДРВ із метою отримання необхідної інформації для прийняття управлінських рішень у сфері використання РЧР.

Структурно автоматизована система радіомоніторингу складається зі стаціонарної та мобільної компонент (рис. 1).

Стаціонарна компонента представлена стаціонарними станціями радіоконтролю різного призначення, які вирішують завдання радіомоніторингу в межах визначеного регіону (зони радіодоступності). Їм можуть бути підпорядковані додаткові станції радіомоніторингу, зокрема: стаціонарні станції, які управляються дистанційно, а також рухомі (мобільні) станції радіомоніторингу.

Мобільна компонента представлена мобільними станціями радіоконтролю та спеціалізованими мобільними (рухомими) станціями технічного радіоконтролю.

За структурою автоматизована система радіомоніторингу у смугах радіочастот загального користування в Україні є розподіленої трьохрівневої.

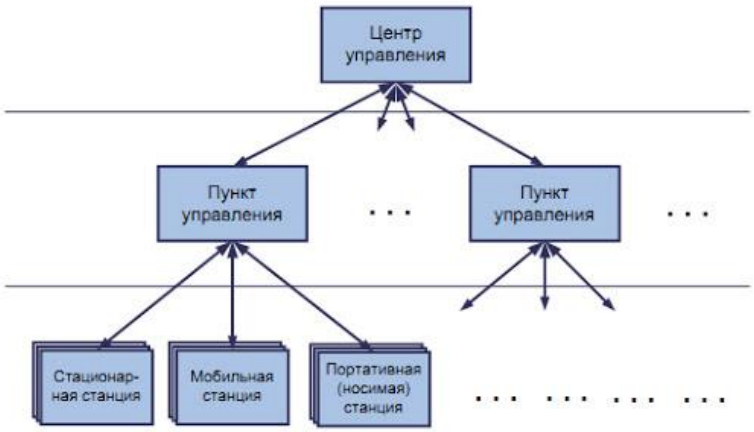


Рис. 1 Структурна схема трьохрівневої системи

© КИРИЛОВА Є.О., 2021

Служба радіомоніторингу може вирішувати три основні групи завдань, а саме:

1) забезпечення перевірки виконання умов ліцензування та частотних присвоєнь;

2) оцінювання реального стану використання РЧР і підготовка даних стосовно зайнятості спектра;

3) підготовка спеціальних звітів в інтересах МСЕ, участю в міжнародних моніторингових заходах.

Вирішення завдання в радіомоніторингу залежить від групи до якої воно належить.

Та зараз переважна більшість завдань радіомоніторингу вирішується за допомогою автоматизованих методів із застосуванням сучасного обладнання та методом спектрального аналізу. Також, ефективність роботи систем радіочастотного моніторингу (СРЧМ) значною мірою визначається рівнем автоматизації процесів радіомоніторингу.

Аналіз методів спектрально-часового аналізу

Головною задачею цієї роботи є розпізнавання сигналів. Вирішенням цього буде метод спектрально-часового аналізу. Спектрально-часовий аналіз (СЧАН) – це метод дослідження нестаціонарних випадкових сигналів, що полягає в оцінці "поточного" спектра частини сигналу, відповідного ковзному часовому вікну деякої малої довжини. Раніше він використовувався на основі перетворення Фур’є, та останній час широко використовується вейвлетне перетворення через свої властивості. Розглянемо обидва методи та зробимо висновки про їх переваги та недоліки.

Класичне перетворення Фур'є, виявляється малоефективним для роботи з нестаціонарними сигналами, для функцій з локальними особливостями, зокрема для імпульсних та цифрових сигналів і зображень. Це пов'язано з тим, що базисна функція рядів Фур'є – синусоїда, визначена на просторі від -∞ до + ∞ і за своєю природою є гладкою і суворо періодичною функцією. Така функція на практиці (в умовах обмеження числа членів ряду або спектра розкладу) принципово не здатна описувати довільні сигнали та функції. Ряд Фур'є використовує як базисну функцію синусоїди. Та ця базисна функція не здатна адаптуватися до локальних змін, тому і виникла необхідність в створенні класу функцій, які не мають даний недолік. Ними стали вейвлети.

Термін "вейвлет" (англ. wavelet) в перекладі з англійської мови означає "коротка або маленька хвиля".

Основа область застосування вейвлет-аналізу є аналіз та обробка нестаціонарних (в часі) або неоднорідних (в просторі) сигналів різних типів. Вейвлет-перетворення полягає в розкладі сигналу по базису, сконструйованому з функції (вейвлета), що володіє певними властивостями, за допомогою масштабних змін та переносів. Базисними функціями вейвлетів можуть бути різні функції, у тому числі ті, що нагадують модульовані синусоїди, функції зі скачками рівня і так далі. Це забезпечує різне уявлення сигналів з локальними скачками і розривами наборами вейвлетів того або іншого типу і відкриває простір в підборі найбільш відповідних вейвлетів, виходячи з умов вирішуваних завдань. Результатом вейвлет-перетворення одновимірного ряду є двовимірний масив амплітуд вейвлет-перетворення.

Перевагою над перетворенням Фур'є це змога виділяти одночасно як частотну, так і часову компоненти мінливості, тобто дає можливість аналізувати часову зміну частотного спектра процесу.

Теперь виявимо переваги та недоліки цих двох методів.

Переваги: фур'є-спектрограми зручні для виявлення періодичної стаціонарної компоненти сигналу. Якщо такою компонентою є гармонійний сигнал, то для його представлення достатня всього одна гармоніка ряду Фур'є.

Вейвлет-перетворення є ефективним математичним засобом локалізації та класифікації особливих точок нестаціонарних сигналів і дозволяє проводити аналіз сигналу одночасно в частотній та часовій областях. Завдяки своїй гнучкості та ефективним обчислювальним реалізаціям вейвлети стали одним з найбільш вживаних методів частотно-часового аналізу сигналів.

Недоліки: основними проблемами при перетворення Фур'є є: збільшення відношення сигнал-шум, яке досягається шляхом усереднення і синхронного накопичення, та мала роздільна

здатність аналізу в високочастотній області, що вимагає застосування процедур детектування (аналізу обвідної).

Однією з проблем є вибір вейвлет-базису при проведенні вейвлет-аналізу, який визначається окремим завданням, але цей вибір нічим не обґрунтований, має випадковий характер і спирається тільки на досвід і думку самого дослідника. Також не існує обґрунтованого підходу для визначення оптимального рівня розкладання для розділення сигналу складної форми на незалежні складові. Крім того, залишається невирішеними ряд завдань, пов'язаних з визначенням параметрів самого вейвлет-перетворення.

Отже, кожний з методів має свої недоліки та переваги. Та, якщо ж покращити деякі характеристики вейвлет-перетворвення, наприклад: створити метод визначення ефективного вейвлет-базису при виконанні вейвлет-перетворення, то можна отримати високоефективний метод спектрально-часового аналізу.

ЛІТЕРАТУРА

1. В. Слободняк, В.Г. Благодарний, В.С. Ступак Довідник з радіомоніторингу
2. Слободянюк П. В. Радиомониторинг: вчера, сегодня, завтра (Теория и практика построения системы радиомониторинга) / П. В. Слободянюк, В. Г. Благодарный. – Прилуки: ООО «Изд-во «Аір-поліграф», 2010. – 296 cGoogle Your Home: Setting Up a Network of Nest Devices In Your Home, December 12, 2019.
3. Смоленцев Н.К. Основы теории вейвле-тов. Вейвлеты в MATLAB / Н.К. Смоленцев. – М.: ДМК, 2008. – 448 с.
4. Chui C.K. Wavelets: A Mathematical Tool for Signal Analysis. / Society for industrialand applied mathematics. Philadelphia, 1997. 210 p.

КИРИЛОВА Єлизавета Олегівна – кафедра безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: xa11867682@student.karazin.ua; ORCID 0000-0002-9001-8989

Наукові інтереси:

- *аналіз методів автоматичного радіомоніторингу (АР);*

УДК 65.0(075.8) **СЕКЦІЯ 2**

КОВАЛЕНКО А. В., БЕРДНІКОВ А. Г.

ПОРІВНЯЛЬНИЙ АНАЛІЗ ЕФЕКТИВНОСТІ ОСНОВНИХ ІНСТРУМЕНТІВ КОНТРОЛЮ ЯКОСТІ ІТ-ПРОЕКТУ

В умовах сучасної економіки питання якості є принципово важливими з точки зору досягнення цілей проекту і його успіху. Підсистема управління якістю поряд з такими підсистемами, як управління вартістю і тривалістю, повинні розглядатися як ключова.

Управління якістю проекту включає в себе процес та дії виконуючої організації, які визначають політики, цілі та сфери відповідальності в області якості таким чином, щоб проект задовольнив ті потреби, заради яких він був зроблений.

Управління якістю проекту направлено як на управління проектом, так і на поставлені результати проекту. Управління якістю проекту поширюється на всі проекти, незалежно від характеру необхідних результатів. Конкретні заходи і методи забезпечення якості залежать від конкретного типу потрібних результатів, вироблених в рамках проекту. Наприклад, для управління якістю необхідних результатів в області програмного забезпечення потрібні інші підходи і заходи, ніж при будівництві атомної електростанції. У будь-якому випадку, невиконання вимог до якості може призвести до серйозних негативних наслідків для окремих або всіх зацікавлених сторін проекту.

Управління якістю проекту направлено на забезпечення відповідних вимог до проекту, включаючи вимоги до продукту, і підтвердження такої відповідності.

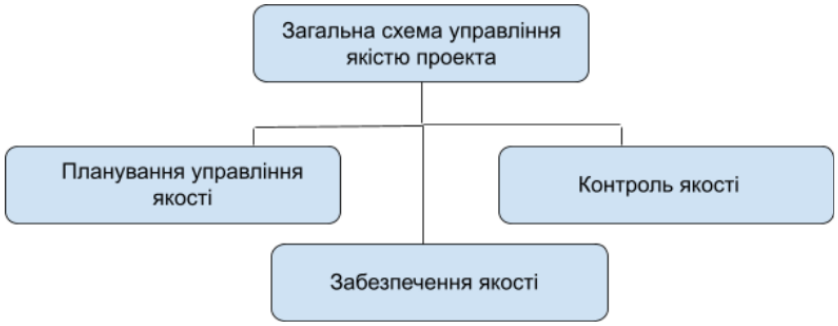


Рис. 1 Загальна схема процесів управління якістю проекту

Дана схема включає в себе :

1) Планування управління якістю - процес визначення вимог і / або стандартів якості для проекту і його потрібних результатів, а також документування того, яким чином проект буде демонструвати відповідність вимогам і / або стандартам якості.

2) Забезпечення якості - процес перевірки дотримання вимог до якості і результатів вимірювань в контролі якості, для забезпечення використання відповідних стандартів якості та операційних визначень.

3) Контроль якості - процес моніторингу та документування результатів дій в області якості для оцінки виконання і винесення рекомендацій щодо необхідних змін.

У процесі контролю якості використовується набір операційних методів і завдань для перевірки відповідності необхідних результатів вимог. Забезпечення якості має використовуватися під час фаз планування і виконання проекту з метою забезпечення впевненості в тому, що вимоги зацікавлених сторін будуть виконані, а контроль якості повинен використовуватися під час фаз виконання і закриття проекту з метою формальної демонстрації, з використанням надійних даних, відповідності критеріям приймання спонсора та / або замовника.

Ключові вигоди даного процесу включають в себе: (1) визначення причин поганої якості процесу або продукту і вироблення рекомендацій та / або вжиття заходів з метою їх усунення; і (2) підтвердження того, що поставляються результати і роботи проекту відповідають вимогам,

© КОВАЛЕНКО А. В., БЕРДНІКОВ А. Г., 2021

визначеним ключовими зацікавленими сторонами в якості необхідних для остаточного затвердження.

У процесі забезпечення якості виконується ряд запланованих систематичних дій і процесів, визначених у плані управління якістю проекту. Метою забезпечення якості є впевненість в тому, що майбутній вихід або незавершений вихід, також відомий як роботи, що знаходяться в процесі виконання, будуть завершені таким чином, яким відповідають зазначеним вимогам і очікуванням. Забезпечення якості сприяє впевненості шляху запобігання дефектам під час процесів планування або інспекції дефектів на стадії виконання робіт. Саме тому кожен проект має критерії якості, що базуються на певних стандартах. Якщо розглядати IT-проект, то основними критеріями оцінки якості будуть для програмного забезпечення(ПЗ) його функціональність, надійність, ефективність, захищеність, портативність та інші. Ці критерії визначають здатність ПЗ виконувати покладені на нього функції. А для визначення контролю якості використовують наступні сім інструментів якості, також відомі як інструменти 7QC(сім основних інструментів якості). Вони використовуються для вирішення проблем, пов’язаних з якістю.

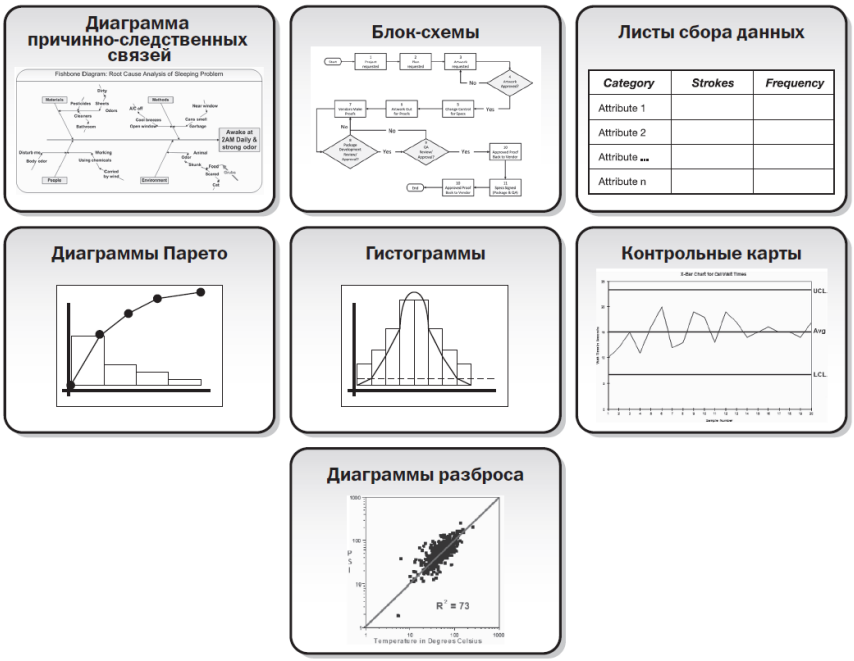


Рис. 2 Приклади основних інструментів якості

Слід розглянути кожний з інструментів більш детально.

1) Діаграми причинно-наслідкових зв'язків, також звані діаграмами «риб'ячий скелет» або діаграмами Ісікава. Опис проблеми, як правило, являє собою виклад проблеми як недоробки, яку необхідно усунути, або мети, яку необхідно досягти. Пошук причин здійснюється шляхом вивчення опису проблеми і пошуку відповідей на питання «чому» до тих пір, поки не буде ідентифікована першопричина, яка потребує прийняття заходів, або до тих пір, поки не будуть вичерпані всі обґрунтовані можливості на кожній частині риб'ячого скелета. Діаграми «риб'ячий скелет» часто виявляються корисними під час пошуку зв'язку небажаних ефектів, що розглядаються як особлива варіація, з встановленої причиною, по відношенню до якої команди проекту повинні виконати коригувальні дії для усунення даної особливої варіації.

2) Блок-схеми, також звані картами процесів, так як вони відображають послідовність кроків і можливості розгалуження процесу, що трансформує один або більше входів в один або більше виходів. Вони можуть виявитися корисними для розуміння і оцінки вартості якості в рамках процесу. Це досягається шляхом використання логіки розгалуження потоку робіт і пов'язаних з нею відносних частот для оцінки очікуваної грошової вартості роботи над відповідністю та роботи над невідповідністю вимогам, необхідної для надання виходу, що відповідає вимогам.

3) Діаграми Парето є вертикальними стовпчасті діаграми особливої форми і використовуються для визначення декількох найбільш важливих джерел, що викликають більшість ефектів проблеми. Категорії, показані на горизонтальній осі, є існуючим розподілом ймовірностей, що враховує 100% можливих спостережень. Значення відповідної частоти виникнення кожної позначеної причини, показаної на горизонтальній осі, зменшується аж до досягнення джерела по замовчуванням, званого «інше», який відповідає за не встановлені причини. Як правило, діаграма Парето організована за категоріями, що вимірює або частоту виникнення, або наслідки.

4) Діаграми розкиду - це нанесення на графік впорядковані пари (X, Y), іноді звані графіками кореляцій, оскільки вони використовуються для пояснення зміни з залежною змінною, Y, щодо зміни, що спостерігається в незалежній змінній, X. Напрямок кореляції може бути пропорційним (позитивна кореляція), зворотним (негативна кореляція), або кореляційної моделі може не існувати (нульова кореляція). Якщо кореляція може бути встановлена, можна визначити лінію регресії і використовувати її для оцінки того, яким чином зміна незалежної змінної може змінити значення залежної змінної.

5) Гістограми - це особливий вид стовпчастий діаграми, який використовується для опису центру розподілу, дисперсії і форми статистичного розподілу. На відміну від контрольної карти гістограма не враховує вплив часу на варіацію, яка існує в межах розподілу.

6) Контрольні карти використовуються для визначення того, чи є процес стабільним чи ні і характеризується він передбачуваним виконанням. Вони можуть бути використані для контролю різних типів вихідних змінних. Хоча найбільш часто контрольні карти використовуються для відстеження повторюваних операцій, необхідних для виробництва промислових виробів, вони також можуть використовуватися для контролю відхилень по вартості і розкладом, обсягу і частоти змін змісту чи інших управлінських результатів, що допомагає визначити, чи знаходяться процеси управління проектом під контролем.

7) Листи збору даних, також відомі як листи для підрахунку, можуть бути використані як контрольні списки при зборі даних. Листи збору даних використовуються для організації фактів таким чином, яке буде сприяти ефективному збору корисних даних про потенційну проблему з якістю. Вони особливо корисні для збору даних про параметри під час виконання інспекцій з метою виявлення дефектів.

ЛІТЕРАТУРА

1. Разу М. Л. Управление проектом. Основы проектного управления: учебник. КНОРУС, 2006. 5-е изд., 768 с.
2. Горбовцов Г.Я. Управление проектом: Учебно-практическое пособие. – М.: Изд. центр ЕАОИ, 2007. – 279 с.
3. Милошевич Д. М60 Набор инструментов для управления проектами / Драган З. Милошевич; Пер. с англ. Мамонтова Е.В.; Под ред. Неизвестного С.И. — М.: Компания АйТи; ДМК Пресс, 2008. – 729 с.: ил.
4. Управление инновационными проектами и программами : учебное пособие / В.В. Быковский, Е.С. Мищенко, Е.В. Быковская и др. – Тамбов : Изд-во ГОУ ВПО ТГТУ, 2011. – 104 с.
5. Клиффорд Ф. Грей, Эрик У. Ларсон. Управление проектами: Практическое руководство/ Пер. с англ. – М.: Издательство «Дело и Сервис», 2003. – 528 с.

КОВАЛЕНКО Анастасія Василівна – кафедра прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: xa11867840@student.karazin.ua; ORCID 0000-0002-5814-5371

БЕРДНІКОВ Анатолій Георгійович – к. т. н., доцент; доцент кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені. В.Н. Каразіна; м. Харків, майдан. Свободи, 6, 61022; e-mail:a.berdnikov@karazin.ua , ORCID 0000-0001-7923-0754

Наукові інтереси:

– *застосування методів теорії ухвалення рішень при управлінні складними телекомунікаційними системами, управління проектами.*

УДК 004.458 **СЕКЦІЯ 7**

КОЗЛОВ В.О., ЛАЗУРИК В.М.

ДОСТУП ДО БАЗ ДАНИХ З ВИКОРИСТАННЯМ ІНСТРУМЕНТІВ PRISMA

Вступ

Двадцять перше століття − час глобальних змін у сфері інформаційних технологій (IT), він характеризується масовою громадською комп'ютеризацією та інформатизацією. Стрімко зростає розробка програмного забезпечення, спрямованого на різні аспекти сфери IT. Особлива увага приділяється веб-розробці, вона включає в себе безліч аспектів сучасної діяльності людини. Це сайти, реклама, електронна комерція, мобільні додатки, ігри та багато іншого. Серед усього цього різноманіття зараз важко знайти веб-додаток, який би не використовував бази даних (БД). Тому питання взаємодії клієнта з базою даних є актуальним. Для їх вирішення використовуються «старі», перевірені часом, методи взаємодії, які постійно удосконалюються і розвиваються. Поряд з ними знаходять застосування нові методи, покликані прискорити розробку і захистити творця від рутинного кодування. Слід зазначити, що через величезну кількість засобів розробки часто буває досить важко оцінити доцільність застосування конкретних програмних рішень та інструментів. Тим більше, що в області IT багато що визначає мода. Наприклад, порівняно недавно, в 2015 році компанією Facebook було опубліковано для масового використання мову запитів і маніпулювання даними GraphQL, що раніше застосовувалась для внутрішніх завдань компанії [1]. На початок 2021 року ця мова вже використовується при створенні багатьох проектів. В [2] автори спробували оцінити доцільність застосування непростого GraphQL, який по суті є додатковим шаром між клієнтом і джерелами даних, при реалізації не високонавантаженого проекту. На прикладі вирішення конкретного завдання оцінювалися можливі проблеми і накладні витрати, пов'язані із застосуванням GraphQL. Такий підхід використаний і в даній роботі.

Робота присвячена розгляду Prisma – нового програмного засобу для роботи з деякими реляційними базами даних (РБД), який позиціонує себе як ORM (object relation mapping) нового покоління для Node.js і TypeScript. У роботі на прикладі рішення задачі проведення онлайн іспитів для студентів робиться спроба оцінити доцільність використання цього програмного забезпечення для ненавантаженого проекту, підтвердити або поставити під сумнів обіцянку «допомогти розробникам будувати їх додатки швидше і робити менше помилок при роботі з PostgreSQL, MySQL і SQLite» [3].

Prisma для роботи з базами даних

Prisma є прошарком між "сухими" (raw) запитами до бази за допомогою javascript і розробником (рис.1). В документації визначається, що Prisma не тільки відображає сутності бази даних у вигляді об'єктів, але й має безліч готових рішень для роботи з даними. Її слоган «служить для абстрагування вашої бази даних» означає, що розробнику не важливо знати, з якою системою управління базою даних (СУБД) він працює. Для нього запити до PostgreSQL, MySQL або SQLite будуть виглядати однаково. Єдине, що він повинен зробити правильно, це створити схему даних і налаштувати підключення Prisma до бази. Автори Prisma підкреслюють, що схема є центральним її елементом, що це − декларативний спосіб визначення моделей даних та їх відносин стосовно додатку, «зрозумілий людині».

Prisma складається з таких інструментів:

1. Prisma Client. Автоматично згенерований та типобезпечний конструктор запитів для Node.js і TypeScript.
2. Prisma Migrate. Декларативне моделювання даних та система міграції.
3. Prisma Studio. Графічний інтерфейс для перегляду та редагування даних у БД.

© КОЗЛОВ В.О., ЛАЗУРИК В.М., 2020

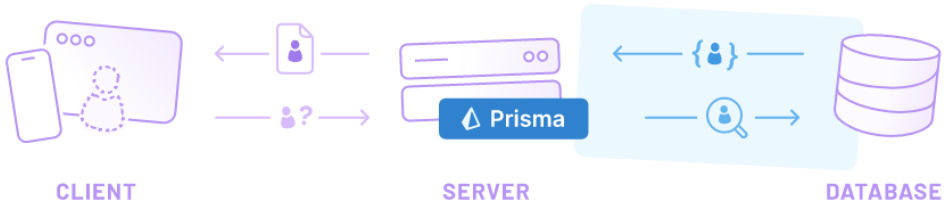


Рис.1 Технологічний стек з Prisma

Автори Prisma вважають, що основною проблемою всіх інструментів взаємодії з базою (SQL, ORM, SQL Query builders) є або недолік контролю над БД і непогана продуктивність при використанні, або повний контроль і погана продуктивність, а саме:

- RAW SQL дає повний контроль над запитами, але малу продуктивність тому, що підключення і відправка запитів до БД, як рядків, дуже громіздка робота. Після виконання запиту повертаються не типобезпечні результати. Також проблема виникає при необхідності переписування коду при зміні моделі даних.
- SQL Query builders залишає високий контроль над запитами і середню продуктивність. Основним мінусом є те, що розробнику все ще доводиться думати про SQL.
- ORM дає менше контролю над БД, але кращу продуктивність. Дозволяє визначати моделі програми у вигляді класів, а не відносин. Мінусом є брак контролю (гнучкості).

На думку авторів Prisma, розробники повинні піклуватися про дані, а не про SQL. Також вони заявляють, що Prisma дозволяє мати високу гнучкість і контроль над БД при найкращій продуктивності.

Задача проведення онлайн іспитів для студентів

Схема БД. Для online проведення контролю знань студентів існують програмні рішення, починаючи з комп'ютерного тестування та закінчуючи співбесідами. Було вирішено для формування тестової задачі розглянути саме цю тему, як таку, яка найбільш затребувана (рис. 2).

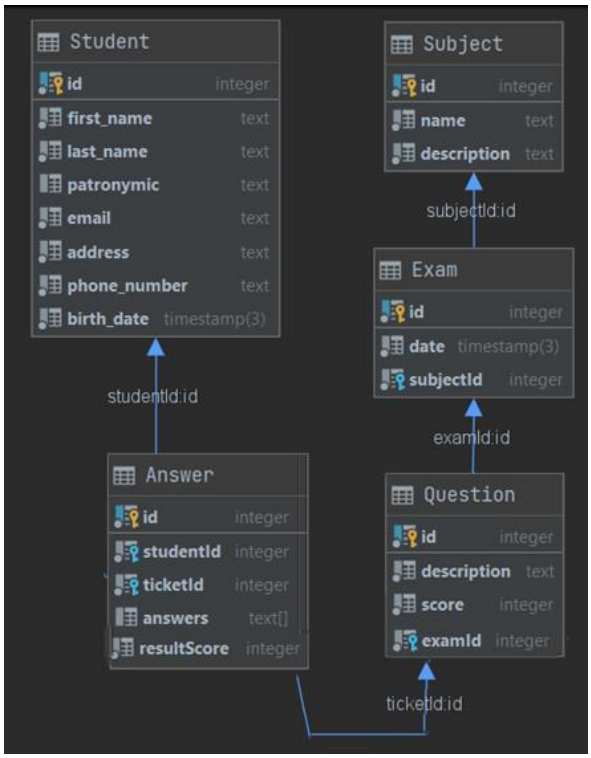


Рис.2 Спрощена схема бази даних щодо проведення онлайн іспитів студентів

Всі таблиці БД мають первинний ключ (PK) та містять дані щодо таких сутностей:

4. Student (студент) – {ім'я, прізвище, по батькові (може не бути), адреса, електронна пошта, номер телефону, дата народження};
5. Subject (предмет) – {назва, опис};
6. Exam (іспит) – {дата проведення, дисципліна, з якої проводиться іспит, викладач};
7. Question (питання) – {текст, максимальна кількість балів, до якого іспиту належить};
8. Answer (відповідь) – {якому студенту належить відповідь, до якого питання відноситься, текст відповіді, кількість балів за відповідь на питання}.

Вибір програмних засобів для розробки. Програмною платформою для серверної розробки було обрано Node.js [4]. Це було обґрунтовано власним досвідом роботи з Node.js, а також тим, що в Інтернеті існують матеріали по використанню зв'язки Prisma + Node.js. В якості середовища розробки був застосований Jetbrains Webstorm [5]. Спочатку була спроба використати в якості СУБД хмарну PostgreSQL за допомогою сервісу ElephantSQL. Але потім при роботі з Prisma від цього прийшлося відмовитися тому, що не було достатньо прав щодо створення бази для міграцій.

Налаштування. Було встановлено локальну PostgreSQL. Налаштування СУБД з Prisma виконувалось згідно документації [6]. Код для підключення БД до Prisma, в якому встановлюється `url` поле `datasource` блоку як змінна середовища, наведений у перших трьох рядках на рис.3. Модель даних необхідно визначити в файлі схеми Prisma. Ця модель потім буде перенесена в БД та створені базові таблиці. Prisma використовує свою мову моделювання даних, код Prisma scheme наведений на рис.3.

```
datasource db {
 provider = "postgresql"
 url      = env("DATABASE_URL")  }

generator client {
 provider = "prisma-client-js"
 previewFeatures = [“createMany”]
}

model Student {
 id    Int     @id
@default(autoincrement())
 first_name String
 last_name String
 patronymic String?
 email String @unique
 address String
 phone_number String @unique
 birth_date DateTime }

model Subject {
 id    Int     @id
@default(autoincrement())
 name String
 description String  }

model Exam {
 id    Int     @id
@default(autoincrement())
 date DateTime
 subject Subject
 teacher String  }

model Question {
 id    Int     @id
@default(autoincrement())
 description String
 score Int
 exam Exam   }

model Answer {
 id    Int     @id
@default(autoincrement())
 answers String[]
 student Student
 ticket Question  }
```

Рис.3 Prisma scheme бази даних онлайн іспитів для студентів

Для синхронізації БД і схеми необхідно виконати команду `prisma db push --preview-feature`. Якщо перевірити створені таблиці в pgAdmin, можна побачити, що Prisma коректно створила `foreign key` в таблицях.

Формування даних. В файлі `app.js` налаштуємо обмін REST запитами для кінцевої точки `students`, для GET запиту будемо повертати всіх студентів, для POST запиту – створювати студента на підставі надісланих даних. Всю логіку для `Student` винесемо в окремий файл `studentService.js` і визначимо дві функції – `findAll` для пошуку усіх

студентів та `createStudent` для створення. В `createStudent` достаємо усе, що одержали в запиті, та викликаємо `prisma.student.create(...)`. Послідовно відправляємо запити на створення окремих записів для студентів, потім відправив `GET /students` запит на пошук усіх студентів, перевіряємо правильність вводу даних. Результат запиту передається як масив JSON об'єктів. Таким же чином працюємо з іншими кінцевими точками та заповнюємо таблиці даними. Перевірити дані можна також і в pgAdmin.

Дослідження можливості формування запитів

Щоб дослідити можливості Prisma було створено та виконано багато запитів. Наприклад:

1. Знайти, які були питання на іспиті, що проводив конкретний викладач. Можливий код запиту наведений на рис.4. Запит виконується та дає коректні результати.

```
exports.findAllByTestTeacher = async (body) => {
    console.log(body)
    try {
        let questions = await prisma.question.findMany({
            include: {
                Session: true
            },
            where: {
                Session: {
                    every: {
                        test: {
                            teacher: {
                                first_name: body.firstName,
                                last_name: body.lastName
                            }
                        }
                    }
                }
            }
        })
        return {isError: false, questions}
    }
    catch (e) {
        return {isError: true, message: "Failed to fetch question for teacher " + e}
    }
}
```

Рис.4 Приклад коду для формування запиту

2. Знайти тих студентів, які на всіх іспитах мають бал 30 і вище. Запит виконується. Але Prisma не дозволяє використовувати вибірку з агрегаціями, якщо необхідне об'єднання кількох таблиць. Це виключає безліч можливих запитів, таких як, наприклад, визначення середнього балу студента по всім іспитам, чи визначення кількості проведених викладачем іспитів.

3. Немає можливості оперувати з підзапитами, алгоритм у цьому випадку – працювати з первинним ключем (PK).

4. Проведено багато запитів з Create, Update, Delete. Все працює коректно.

5. Були створені досить прості запити, але зі складною умовою на вибірку.

Комп'ютерні експерименти по оцінюванню продуктивності запитів

Були проведені експерименти по оцінюванню продуктивності запитів. В середовищі Webstorm було розроблено кілька простих «сухих» (RAW) запитів:

а) на вибірку з використанням внутрішнього об'єднання таблиць (Join) та з умовами різної складності;

б) запити на розрахунок статистичних показників, що застосовують агрегатні функції. Такі ж самі запити були реалізовані з використанням Prisma. Оцінювання проводилось за рахунок фіксації часу початку та закінчення виконання запитів. Такий механізм не дає змоги реально оцінити час виконання запитів тому, що значення цього часу залежить від багатьох показників. Але в цьому експерименті оцінювання реального часу виконання запиту було не до

потреби. При усіх однакових умовах порівнювались одержані показники часу у випадку «сухих» запитів та при використанні Prisma.

Експерименти проводились на різній потужності таблиць, невеликій (~3000 записів),− коли моделюється проведення іспитів на одному курсі, та середній (~75000 записів) при проведенні іспитів на факультеті. Було помічено, що перший по порядку запит з використанням Prisma завжди виконується набагато довше (в 12-13 разів) ніж такий же самий RAW запит. Було зроблене припущення, що для виконання низки запитів Prisma повинна якось «підготуватися», тому перед виконанням першого запиту в усіх експериментах було вставлено «фальшивий» запит, для якого час не оцінювався.

Одержані показники для невеликої потужності таблиць порівняні у двох випадках, за винятком дуже простих запитів, в яких не застосовується об'єднання таблиць і які мають прості умови пошуку. Для таких запитів час виконання з Prisma вдвічі менший, а для запитів з об'єднанням таблиць майже в два рази більший.

При оцінюванні запитів для середньої потужності таблиць зберігалося таке ж саме співвідношення часу виконання як і для невеликої потужності таблиць за виключенням тих запитів, що використовували кілька об'єднань за простими умовами. Для них час виконання в обох випадках майже однаковий, навіть з Prisma десь на 10% - 11% швидше.

Висновки

В роботі на прикладі додатку для дистанційного проведення іспитів студентів були розглянуті можливості та особливості використання Prisma. В результаті проведених досліджень можна зробити наступні висновки.

1. Комп'ютерні експерименти по оцінюванню продуктивності «сухих» запитів в порівнянні з запитами, розробленими під Prisma, показали, що швидкість виконання майже однакова, за виключенням простих запитів під Prisma, які виконуються спритніше. Але виникає питання, якщо додаток використовує прості запити і йому цього вистачає, навіщо застосовувати такий важкий програмний засіб, як СУБД PostgreSQL? А застосування PostgreSQL з додатковим шаром Prisma взагалі не доцільне. Автори вважають, якщо до потреби розробникові прості запити, зручніше та продуктивніше використовувати NoSQL СУБД. Застосування реляційних баз даних для розробки веб-додатків повинно бути визначено за умови використання усіх можливостей, які вони надають.

2. Мова структурованих запитів SQL, яка використовується в усіх реляційних базах даних, стандартизована. Вона має реляційну алгебру та реляційне числення в своїй основі. Для того, щоб будувати запити для PostgreSQL, MySQL чи SQLite не треба вивчати мову запитів кожної з цих СУБД. Твердження Prisma про те, що розробники мають думати про дані, а не про SQL, виглядає привабливо, але побудова запитів в Prisma з використанням багатьох `Include` замість `Join` та багатоповерхових конструкцій для побудови умов замість звичних математичних виразів SQL оператора вибірки, не можна визначити як легше, більш наочне та читане.

3. Prisma має обмеження. Неможливість застосування агрегатних функцій в запитах з кількома об'єднаними таблицями відрізають можливість використання цілого класу корисних запитів.

4. В цій роботі не всі можливі експерименти щодо освоєння Prisma були проведені і не до всього вдалося доступитися. Наприклад, не вдалося застосувати Prisma Migrate до PostgreSQL бази даних. В цьому випадку Prisma вимагає прав адміністратора серверу чи суперкористувача. Але коли сервер PostgreSQL – це корпоративний сервер, ніхто не дасть окремому розробникові бази даних права суперкористувача, а прав адміністратора своєї бази даних недостатньо.

5. На цей час Prisma дозволяє працювати з обмеженою кількістю реляційних баз даних. Реляційна модель розвивається вже багато років і людство багато чого має в своєму арсеналі для роботи з даними цієї моделі. В той же час таке програмне рішення як GraphQL дає можливість працювати в єдиному стилі як з реляційними, так і з NoSQL базами даних. Тому Prisma не може бути рекомендована як така, що краща ніж GraphQL.

Крім того існує емоційна частина, яка складається при вивченні Prisma. Погане враження залишається від перегляду офіційного сайту Prisma, який є лише рекламою. В

документації немає жодного доказу, чи яких-небудь кількісних показників, які дозволи би підтвердити, що запропоноване програмне рішення має високу гнучкість та контроль за найкращою продуктивністю в порівнянні з тими засобами, що застосовуються.

Резюмуючи усі пункти висновків, можна сказати, що Prisma – це ще сире програмне рішення і застосування його для класу ненавантажених проектів не доцільне. Також для додатків, що використовують для розробки ORM, не критична зміна однієї РБД на іншу. А постійна переробка моделі даних, з якою працює додаток, свідчить про погане проектування та відмову від досконального прототипування на користь скорішого кодування. Водночас для високонавантажених проектів, що застосовують кілька різноманітних СУБД, автори вважають доцільним застосування GraphQL і не вбачають необхідності ще одного прошарку в технологічному стеку, такого як Prisma.

ЛІТЕРАТУРА

1. Lee Byron. GraphQL: A data query language. [Електронний ресурс] Режим доступу: https://engineering.fb.com/core-data/graphql-a-data-query-language/

2. Черняєв І.М., Лазурик В.М. Использование GRAPHQL для работы с базами данных // Праці міжнародної науково-технічної конференції «Комп'ютерне моделювання в наукоємних технологіях», Харків: ХНУ імені В. Н. Каразіна, 2020 — С. 282—287, ISBN 978-966-285-639-2

3. Офіційний сайт Prisma. [Електронний ресурс] Режим доступу: https://www.prisma.io/

4. Node.js. [Електронний ресурс] Режим доступу: https://nodejs.org/ru/download

5. WebStorm. The smartest JavaScript IDE. [Електронний ресурс] Режим доступу: https://www.jetbrains.com/webstorm/

6. Getting started /Set up Prisma. [Електронний ресурс] Режим доступу: https://www.prisma.io/docs/getting-started/setup-prisma/start-from-scratch-node-postgres

КОЗЛОВ Віталій Олександрович – студент факультету комп'ютерних наук Харківського національного університету імені В. Н. Каразина. Наукові напрямки:

– *розробка програмного забезпечення.*

ЛАЗУРИК Валентина Михайлівна – старший викладач кафедри штучного інтелекту та програмного забезпечення факультету комп'ютерних наук Харківського національного університету імені В. Н. Каразіна.

Наукові інтереси:

– *розробка комп'ютерних систем для моделювання процесів в радіаційних технологіях; організація баз даних.*

УДК 539.3 **СЕКЦІЯ І**

КОНОНЕНКО Є.С.

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ КОЛИВАНЬ КРУГЛИХ ПЛАСТИН ЗА РІЗНІ УМОВИ ЗАКРІПЛЕННЯ МЕТОДОМ СКІНЧЕННИХ ЕЛЕМЕНТІВ

Вступ

Інтерес до проблем коливань елементів конструкцій, особливо при взаємодії з робочим середовищем: рідиною, газом, зрідженим газом і т. ін. не вщухає вже протягом багатьох десятиліть. Це пов'язано з надзвичайною актуальністю цих питань в практиці енергомашинобудування, нафтохімічної промисловості, авіаційної та ракетної техніки, наземного та підводного будівництва [1-3]. Слід зауважити, що для визначення динамічних характеристик елементів конструкцій за умови гідропружної взаємодії загальноприйнятним є метод заданих форм [4]. При цьому вважають, що форми коливань елементу конструкції, який повністю або частково занурений в рідину, визначаються як лінійна комбінація форм коливань цього ж елементу, але без врахування приєднаних мас рідини. Тобто вказані форми коливань створюють базисну систему для визначення частот та форм коливань елементу конструкції, що взаємодіє з рідиною, або здійснює вимушені коливання під дією зовнішніх нестаціонарних впливів. Зазначимо, що при визначенні вказаних частот та форм велике значення мають граничні умови або умови закріплення пружного елементу. Але в науковій літературі цьому питанню приділено недостатньо уваги [5]. В даній роботі вивчаються коливання пружної пластинки за різні умови закріплення.

Формулювання задачі

Розглянуто кругову пружну пластинку радіуса R. Однім з простіших рівнянь для опису коливань такої пластинки є диференціальне рівняння, що засновано на гіпотезах Кірхгофа -Лава і має вигляд

$$D\Delta\Delta w+\rho_p h\frac{\partial^2 w}{\partial t^2}=q(x,y.t) \tag{1}$$

де $w(x,y,t)$ - прогин пластини, ρ_p - густина матеріалу пластини, $D=\frac{Eh^3}{12(1-\nu^2)}$ - циліндрична жорсткість, $q(x,y,t)$ - сила, що діє на пластину.

Якщо функція $q(x,y,t)=0$, то пластина здійснює вільні коливання, і розв'язок рівняння (1) шукаємо у вигляді

$$w(x,y,t)=\exp(i\Omega t)w(x,y).$$

Існують інші підходи до вивчення вільних коливань пластин, засновані на співвідношеннях теорії пружності [5].

В цьому дослідженні розглянуті вільні коливання круглої пластинки в двох випадках граничних умов, а саме: жорсткого закріплення та шарнірного обпирання.

Умови жорсткого закріплення мають вигляд [6]

$$w\big|_{\rho=R}=0,\quad \left.\frac{dw}{dr}\right|_{\rho=R}. \tag{2}$$

Умови шарнірного обпирання є такими:

© КОНОНЕНКО Є.С., 2021

$$w\big|_{\rho=R}=0, \quad M\big|_{\rho=R}, \tag{3}$$

де M – момент згину [6].

В роботі досліджені вільні коливання круглої пластинки з умовами закріплення у вигляді (2) та (3) шляхом використання співвідношень теорії пружності з використанням методу скінченних елементів [7]. Основне співвідношення цього підходу має вигляд

$$[\mathbf{M}]\ddot{\mathbf{u}} + [\mathbf{C}]\dot{\mathbf{u}} + [\mathbf{K}]\mathbf{u} = \{\mathbf{f}\},$$

де $[\mathbf{M}],[\mathbf{C}],[\mathbf{K}]$ - матриці мас, демпфування та жорсткості, відповідно, $\{\mathbf{f}\}$ - вектор заданих зусиль, що діють на пружне тіло, $\mathbf{u}$ - переміщення пружного тіла.

Зроблено порівняння даних, отриманих з використанням тривимірної теорії пружності, теорії Кірхгофа - Лава [8] та аналітичного розв'язку [6]. Слід зауважити, що в [8] запропонований редукований метод скінченних елементів, що заснований на припущенні аксіальної симетрії елементу конструкції, що розглядається.

Аналіз числових результатів

Розглянуто кругову пластинку за такими параметрами: модуль пружності E= $2.06\cdot10^8$ кПа, коефіцієнт Пуассона ν = 0.3, густина матеріалу ρ = 7850 кг/м^3, товщина пластинки h = 0.01м, радіус пластинки R = 0.5 м. Пластинка розглядалась за різні умови закріплення (2) та (3).

Розрахунки довели, що результати, отримані запропонованим методом, методом [7] гарно корелюють з аналітичним розв'язком [6].

В таблиці 1 наведені частоти коливань жорстко закріпленої та шарнірно-обпертої пластин.

Табл.1. Частоти коливань круглих пластин за різні умови закріплення, Гц

номер частоти m	*жорстке закріплення*		*шарнірне обпирання*	
	аналітичний розв'язок	*числовий розв'язок*	*аналітичний розв'язок*	*числовий розв'язок*
1	48.7043628	48.697	100.8186992	100.74
2,3	137.159236	137.02	209.8161805	209.52
4,5	252.774391	252.19	344.1970592	343.37
6	293.302974	293.04	392.4964410	391.67

З отриманих даних бачимо, що аналітичні та числові розв'язки добре узгоджуються, частоти коливань суттєво залежать від типу закріплення.

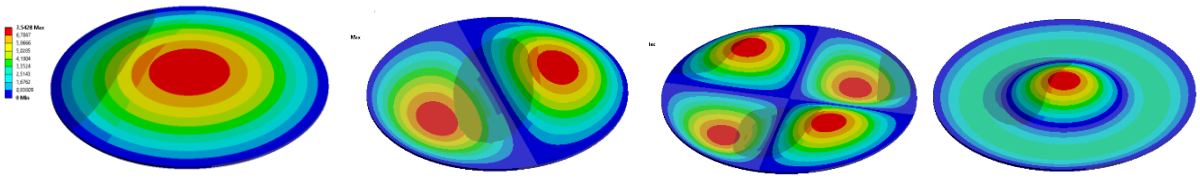


Рис. 1 Форми коливань круглої пластинки

На рис. 1 зображені форми коливань круглої пластини, що відповідають першим 6 частотам. Зауважимо, що умови закріплення впливають біль суттєво на частоти коливань, форми коливань внаслідок першої з умов як в (2), так і в (3).

Висновки

Проаналізовані частоти та форми коливань круглої пластини за різні умови закріплення. Встановлено, що нижчі частоти відповідають випадку шарнірного обпирання. З'ясовано, що метод скінченних елементів при розв'язанні задачі в тривимірному формулюванні дає прийнятну

точність. Хоча цей метод є досить громіздким та потребує великих затрат часу в порівнянні з одновимірним методом скінченних елементів, але саме використання такого підходу дозволить в подальшому перейти до дослідження як вільних так і вимушених коливань елементів реальних просторових конструкцій.

ЛІТЕРАТУРА

1. Еселева Е.В. Собственные колебания сосудов высокого давления при взаимодействии с жидкостью / Е.В. Еселева, В.И. Гнитько, Е.А. Стрельникова // *Пробл. машиностроения*. –2006. Т. 9. – №1, – С.105 - 118.
2. Medvedovskaya T. Free Hydroelastic Vibrations of Hydroturbine Head Covers / T. Medvedovskaya, E. Strelnikova, K. Medvedyeva // *Intern. J. Eng. and Advanced Research Technology (IJEART).* – 2015. – Vol. 1, No 1. – P. 45 - 50. – DOI 10.13140/RG.2.1.3527.4961
3. Місюра С. Ю., Сметанкіна Н. В., Місюра Є. Ю. Раціональне моделювання кришки гідротурбіни для аналізу міцності. *Вісн. Нац. техн. ун-ту «ХПІ». Сер. Динаміка і міцність машин*. 2019. № 1. С. 34–39.
4. Strelnikova E., Gnitko V., Krutchenko D., Naumemko Y. Free and forced vibrations of liquid storage tanks with baffles J. Modern Technology & Engineering Vol.3, No.1, 2018, pp.15-52
5. Сегерлинд Л. Применение метода конечных элементов – М.: Мир, 1979. – 392 С.
6. M. K. Kwak. Axisymmetric vibration of circular plates in contact with fluid. Journal ofSound and Vibration (1991) 146(3), 381-389.
7. Розин Л. А. Задачи теории упругости и численные методы их решения / Л. А. Розин ; Санкт.-Петерб. гос. техн. ун-т. – СПб. : СПбГТУ, 1998. – 532 с.
8. Кочуров Р.Е, Белов СА, Аврамов КВ, Гнитько ВИ. Прочность вертикального кожухотрубного теплообменного аппарата при сейсмическом воздействии. Інтегровані технології та енергозбереження, №1, 2011, с. 35-40.

КОНОНЕНКО Єгор Сергійович – аспірант, Інститут проблем машинобудування ім. А.М. Підгорного НАН України, вул. Пожарського, 2/10, м. Харків, Україна, 61046;, yehorkononenko@gmail.com, https://orcid.org/0000-0001-6221-3608.

- комп'ютерне моделювання методом скінченних елементів

УДК 531.8 **СЕКЦІЯ 2**

КУРГАН А.Р., КОТВИЦЬКИЙ А.Т.

ФІЗИКО-МАТЕМАТИЧНІ МЕТОДИ АНАЛІЗУ РУХОМИХ ПЛАТФОРМ

Вступ

На даний момент існує велика кількість розробок різних алгоритмів управління мобільними роботами, зокрема рухомими платформами, які забезпечують можливість проведення нетривіальних операцій: уточнення або створення карти місцевості, планування оптимальних способів обходу перешкод, проникнення в небезпечні або важкодоступні для людини зони, забезпечення руху по складних траєкторіях зі збереженням при цьому заданої орієнтації конструкцій робота, доставка необхідних вантажів по зараженій або несприятливій місцевості і тому подібні завдання. Створення нових типів мобільних платформ диктується в першу чергу завданнями, які виникають в тих чи інших областях діяльності людського розуму: автоматизація транспортних потоків і засобів, попередження терористичних актів, дистанційне розмінування небезпечних предметів, гасіння пожеж, проведення операцій і т.п. Для ефективного виконання поставлених завдань інтелектуальні роботи повинні бути не тільки забезпечені різними системами і датчиками для сприйняття навколишнього середовища, мати засоби для швидкого аналізу можливих ситуацій і знаходити способи вирішення конфліктів. В першу чергу, вони повинні мати конструкційну можливість зробити, ту чи іншу дію, спрямовану на виконання конкретного завдання.

Постановка задачі

Розробити мобільну квазіголономну систему для переміщення по твердій поверхні.

Конструкція мобільної платформи

Розроблена нами рухома платформа являє собою каркас, зварений з профільної квадратної труби. По краях платформи встановлені чотири незалежних мотор-колеса діаметром 25 см і потужністю 350 ват. Для повороту всієї платформи кожне колесо було зроблено поворотним, причому поворот коліс здійснюється незалежними кроковими двигунами NEMA23, які можуть розвинути на валу максимальний момент 1,8 Нм. Для збільшення моменту двигуна був застосований черв'ячний редуктор NMRV25 з передавальним числом рівним двадцяти. Цей редуктор дозволив не тільки збільшити момент сили, що передається при повороті колеса, але і зафіксувати колесо під потрібним кутом при здійсненні заданого маневру, а також знизити струм, споживаний кроковим двигуном при здійсненні повороту.

Аналіз механізму повороту мобільної платформи

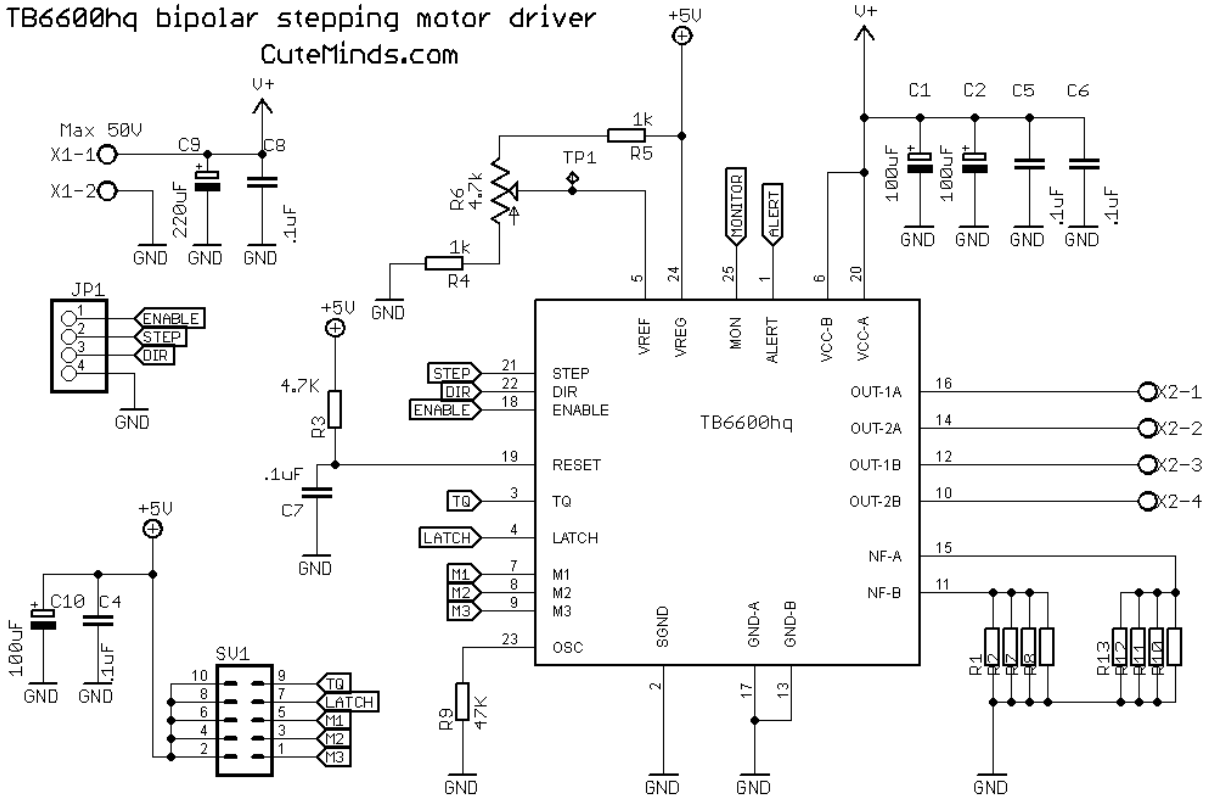
Однією з основних завдань на етапі розробки було сконструювати механізм повороту колеса навколо вертикальної осі і забезпечити його оптимальну роботу з урахуванням енергоспоживання поворотного вузла. Мінімальний момент сили, який необхідно прикласти до колеса на сухому асфальтовому покритті щоб його повернути навколо вертикальної осі становить $M_{min} = 2{,}1$ Нм, при $d = 5$ см та $F = 42$ Н, де d – це плече сили F, яка потрібна для повороту колеса.

Максимальний момент сили було виміряно в припущенні того що мобільна платформа перевозить корисний вантаж масою 150 кг по сухому бетону. При таких умовах $M_{max} = 15{,}3$ Нм. Ці умови були обрані з того що згідно [1] коефіцієнт тертя між гумою і сухим бетоном зазвичай приймається вище ніж коефіцієнт тертя між гумою і сухим асфальтовим покриттям.

Таким чином видно, що крокoвий двигун, який ми використовуємо, не зможе без додаткових пристосувань здійснити поворот колеса навколо вертикальної осі навіть в найсприятливішому випадку. Для збільшення моменту сили, що прикладається до колеса було вирішено використовувати черв'ячний редуктор типорозміру NMRV25. Згідно [2] черв'ячний редуктор з передавальним числом рівним 20 має ККД 0,75. Так як на валу крокового двигуна

© КУРГАН А.Р., КОТВИЦЬКИЙ А.Т., 2021

23HS8430 максимальний створюваний момент становить 1,8 Нм (див. [3]), то на вихідному валу редуктора ми зможемо отримати максимальний момент рівний $M_{редуктор} = 0{,}75 \cdot 20 \cdot 1{,}8 = 27$ Нм. Цей момент сили практично в два рази перевищує необхідний максимальний момент для повороту колеса. Більш того, цей момент досягається при постійному струмі на одну фазу рівним 3,0 А. Звідси видно, що виставляння робочого струму 3 А є, хоч і допустимим для крокового двигуна, але не є оптимальним з точки зору енергоефективності, оскільки призведе до перевитрати електроенергії. Це загрожує несвоєчасним розрядом акумулятора і як наслідок втратою мобільності робота. Завдання, яке ставилося в даній роботі це експериментальне визначення оптимального режиму роботи. Тобто по суті справи виставлення оптимального робочого струму. Як драйвера для управління кроковим двигуном була застосована схема на основі TB6600hq. Принципова електрична схема використаного драйвера приведена на малюнку 1.



Мал.1. Принципова електрична схема драйверу TB6600hq.

Цей драйвер був обраний по ряду причин, по-перше, підвищена максимальна напруга на вході до 50 вольт, що дозволяє підключати стандартний велосипедний акумулятор із середньою напругою 36 вольт і максимальною 42 вольта. По-друге, даний драйвер має плавне регулювання робочого струму, що є в нашому випадку істотним плюсом. Згідно Мал.1 видно, що для виставлення робочого струму використовується змінний резистор R6 номіналом 4,7 кОм. Проблема полягає в тому, що пряме вимірювання струму в процесі роботи крокового двигуна є досить складним завданням, тому що струм змінює свій напрямок кожні 400 мікросекунд. Тому нами була розроблена спеціальна методика, що дозволяє вимірювати виставлений струм.

Згідно з технічним описом на даний драйвер при підключенні піна TQ мікросхеми TB6600hq (третьої ніжки) на шину GND струм утримання складе 50% від максимального. Для управління драйвером був обраний мікроконтролер серії AVR MEGA2560 [4,5]. У режимі тестування пін STEP (21 ніжка) був підключений на 23 ніжку мікроконтролера і керувався четвертим бітом порту B. Пін DIR (22 ніжка) був підключений на 17 ніжку мікроконтролера і керувався п'ятим бітом порту H. Програма управління пінами була написана на мові CI [6], між обертанням вала двигуна за годинниковою і проти годинникової стрілки ми виставили паузу в 10 секунд, саме в цей момент драйвер переключався в режим утримання, в якому струм падав до 50% і тестер в режимі амперметра включений в розрив однієї з фаз мотора встигав виміряти силу

струму. За допомогою пінів M1, M2, M3 (ніжки 7, 8, 9) мікросхеми TB6600hq було встановлено полукроковий режим роботи типу А, для цього на M1, M3 було подано логічний нуль, а на M2 – логічна одиниця. Полукроковий режим дозволив збільшити точність позиціонування кута повороту валу без втрати механічного моменту. В результаті, для здійснення повного обороту валу двигуна потрібно було зробити 400 кроків, а для здійснення повного оберту вихідного валу редуктора 8000 кроків. При цьому при обертанні валу редуктора на нього встановлювався вантаж різної маси для виміру реального моменту сили. Після серії проведених випробувань нами було отримано, що при струмі 2,48 А момент сили становить 15,3 Нм.

ЛІТЕРАТУРА

1. Таблиця коефіцієнтів тертя спокою URL: https://alexfl.pro/inform/inform_stali5.html
2. Основні технічні характеристики черв'ячних редукторів NMRV25 URL: https://www.digchip.com/datasheets/parts/datasheet/000/NMRV25-7_5-9_8056B4-pdf.php
3. Технічні характеристики крокових моторів NEMA23 URL: http://www.motionking.com/Products/Hybrid_Stepper_Motors/2-phase/23H2A_Stepper_Motor_57mm_1.8degree.htm
4. Белов А. В. Разработка устройств на микроконтроллерах AVR: шагаем от «чайника» до профи. Наука и Техника, Санкт-Петербург, 2013, с.530
5. Takashi Kenjo Stepping motors and their microprocessor controls. Clarendon press. Oxford. 1984 p. 252.
   Кенио Т. Шаговые двигатели и их микропроцессорные системы управления. Москва. Энергоатомиздат, 1987 с. 201.
6. Шпак Ю.А. Программирование на языке С для AVR и PIC микроконтроллеров. Киев, МК-Пресс, 2011, с.546.

КОТВИЦЬКИЙ Альберт Тадеушевич – к.ф.-м. н., доцент; доцент кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: kotvytskiy@karazin.ua; ORCID: 0000-0001-8283-505X.

Наукові інтереси:

– *теорія гравітації, гравітаційні лінзи, робототехніка*.

КУРГАН Анастасія Русланівна – студентка факультету комп'ютерних наук, групи КУ-41; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: xa11867688@student.karazin.ua; ORCID 0000-0003-4093-5559.

Наукові інтереси:

– *Робототехніка, Веб-програмування.*

УДК 629.7.036.3 **СЕКЦІЯ 1**

ЛИСИЦЯ О.Ю., КОВАЛЬОВ А.В., МИХАЙЛЕНКО Т.П., ПЕТУХОВ І.І.

МОДЕЛЮВАННЯ ПРОЦЕСІВ УТВОРЕННЯ МАСЛЯНОЇ ПЛІВКИ НА СТІНКАХ КАМЕРИ ПІДШИПНИКА ГТД

Вступ

Зростаючі термодинамічні параметри сучасних газотурбінних двигунів, а також зменшення їх загальних розмірів пред'являють особливі вимоги до процесу проектування камер підшипників у зв'язку з ускладненням теплонапруженого стану. З одного боку, збільшення частоти обертання вала, температури, тиску і швидкості потоку газу потребує більшої витрати масла для забезпечення охолодження і змащення, а з іншого боку, постає необхідність економії паливо-мастильних ресурсів, мінімізації енергоспоживання маслосистеми та витрати повітря для наддуву камер. Тільки глибоке розуміння процесів у камері підшипника може забезпечити вирішення цієї проблеми.

Параметри масляної плівки на стінках камери відіграють ключову роль у процесах переносу теплоти. Тому достовірне моделювання течії плівки важливе як при проектуванні камер підшипника, так і для підбору раціональних витрат повітря і масла. Параметри плівки є сукупним результатом вкладу ефектів міжфразної взаємодії в умовах впливу відцентрових сил та тяжіння. Наявний математичний апарат описує окремі ефекти, однак недостатньо розвинений для їхнього урахування у сукупності. Узагальнюючих залежностей для розрахунку мало, а представлені критерії подібності не враховують наслідків міжфазної взаємодії при утворенні плівки та діаметр крапель. Тому їх придатність навіть для геометрично подібних камер потрібно перевіряти. Крім того, невизначеності, пов'язані з описом поведінки масляної плівки, параметрів потоку крапель та умов міжфазної взаємодії в пристінній області ускладнюють використання існуючих математичних моделей [1-4].

Фізична та математична моделі процесу

Нагріте масло через роликовий підшипник надходить в камеру, яка утворена рухомим валом і корпусом. Через лабіринтове ущільнення в камеру подається повітря. Суміш масла та повітря виходить з камери через трубопроводи суфлювання і відкачування. При формуванні математичної моделі у центрі камери (ядрі) розглядається турбулентний потік, що складається з двох взаємно нерозчинних фаз, де несучою фазою є повітря, а дисперсною – краплі масла.

В даній роботі процеси формування масляної плівки на внутрішній стінці камери підшипника моделюються на базі Єйлерової моделі плівки (Eulerian Wall Film), що реалізована у програмному пакеті ANSYS Fluent. Ця модель описує тонку плівку рідини, що утворюється на поверхні стінки за рахунок крапель, які потрапили на тверду поверхню з пристінної області та ядра потоку. Розглядається кілька варіантів взаємодії основного потоку і тонкої плівки: прилипання, коли крапля потрапляє на стінку з невеликою енергією і залишається майже сферичною; відскок, коли крапля залишає поверхню плівки зі зміненою швидкістю; поглинання, коли крапля потрапляє на стінку з помірною енергією і поглинається плівкою; роздроблення, коли частина падаючої краплі приєднується до плівки, а інша частина повертається в потік меншими краплями.

Плівка виступає в даному випадку як третя фаза, для якої окремо записуються закони збереження маси, енергії та імпульсу. Для тривимірної моделі розрахунку закон збереження маси виглядає як:

$$\frac{\partial \rho_l h}{\partial t} + \nabla_s \cdot \left(\rho_l h \vec{V}_l \right) = \dot{m}_s , \qquad (1)$$

де ρ_l – густина масла; h – товщина масляної плівки; t – час; ∇_s – оператор поверхневого градієнта; $\vec{v}_l$ – середня швидкість руху плівки; $\dot{m}_s$ – джерело маси на одиницю площі стінки

© ЛИСИЦЯ О.Ю., КОВАЛЬОВ А.В., МИХАЙЛЕНКО Т.П., ПЕТУХОВ І.І., 2021

внаслідок подрібнення крапель, відокремлення їх з плівки, зменшення плівки і фазового переходу.

Закон збереження імпульсу для плівки записується як:

$$\frac{\partial \rho_l h\vec{V}_l}{\partial t}+\nabla_s\cdot\left(\rho_l h\vec{V}_l\vec{V}_l+\vec{D}_V\right)=-h\nabla_s P_L+\rho_l h\vec{g}_\tau+\frac{3}{2}\vec{\tau}_{fs}-\frac{3\mu_l}{h}\vec{V}_l+\dot{\vec{q}}_s+\vec{\tau}_{\theta w}, \quad (2)$$

де $\vec{D}_V$ – тензор, розрахований на основі квадратичного профілю швидкості плівки;

$P_L=P_{gas}+P_h+P_\sigma$; $P_h=-\rho h(\vec{n}\cdot\vec{g})$; $P_\sigma=-\sigma\nabla_s\cdot(\nabla_s h)$;

P_{gas} – сила тиску зі сторони газового потоку; $\vec{\tau}_{fs}$ – тензор деформації плівки; μ_l – динамічна в'язкість масла; $\dot{\vec{q}}_s$ – густина теплового потоку; $\vec{\tau}_{\theta w}$ – сила поверхневого натягу рідини плівки.

Результати і їх обговорення

Моделювання потокорозподілу в камері підшипника проводилося за допомогою пакета ANSYS Fluent з використанням гомогенної моделі багатофазного потоку Volume of Fluid, "k- ω SST" моделі турбулентності для нестаціонарного процесу в тривимірній постановці. Після досліджень сіткової збіжності була вибрана сітка, що складається з тетра- і гексагональних елементів загальною кількість 0,7 млн. елементів. Результати CFD-розрахунку товщини масляної плівки та коефіцієнта тепловіддачі по розробленій моделі порівнювалися з експериментальними даними.

Згідно з результатами розрахунку краплі масла під впливом відцентрових сил і безпосередньо потоку повітря випадають на стінку камери, утворюючи плівку. Товщина масляної плівки по колу камери є змінною (рис. 1). Основний діапазон змін становить 0,3-0,6 мм. Швидкість руху плівки визначається імпульсом крапель, що осідають на стінку, силою тяжіння та зсувною силою в міжфазній області, а також поверхневим тертям. Внаслідок зміни співвідношення внеску цих факторів товщина і швидкість масляної плівки, а отже і її термічний опір, неоднорідні. Значення швидкості руху плівки головним чином знаходяться в межах 0,4-0,8 м/с.

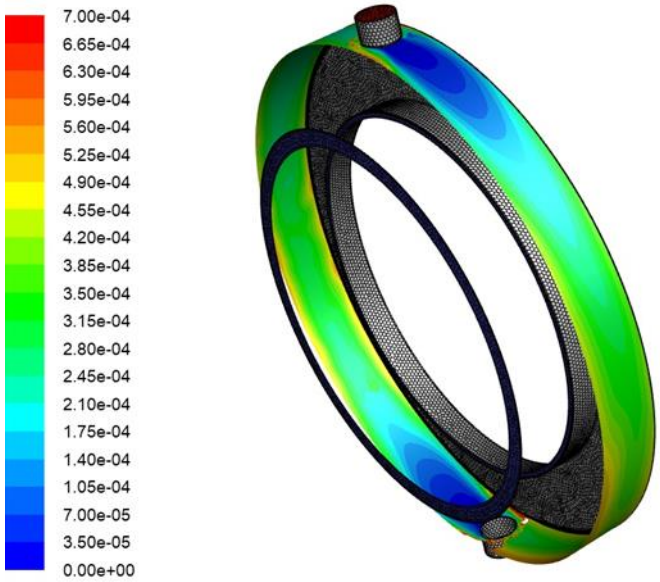


Рис. 1 Розподіл товщини масляної плівки (м) на внутрішній поверхні камери

Гравітація завжди направлена вниз. У той же час вал при своєму обертанні залучає повітря, що надходить у камеру через лабіринтові ущільнення, а сила зсуву спрямована вздовж поверхні плівки. Коли зсувна сила від потоку повітря і сили тяжіння діють в одному напрямку, швидкість плівки збільшується, а товщина зменшується. У протилежній частині камери ці напрямки не збігаються, і плівка сповільнюється зі збільшенням її товщини.

Перенесення теплоти з ядра потоку до масляної плівки здійснюється за рахунок повітря, що примикає до плівки, та за рахунок крапель, котрі на неї потрапляють. Виконаний аналіз показав, що визначальну роль у цьому процесі відіграє осідання крапель. Такий механізм суттєво інтенсифікує перенесення теплоти і тому основний внесок в термічний опір внутрішньої тепловіддачі вносить плівка. Розглянуто питання моделювання її параметрів з урахуванням факторів, що визначаються режимом роботи ГТД.

Висновки

Аналіз CFD-розрахунків показав, що, незважаючи на змінну товщину плівки, вся стінка покрита маслом. В основному товщина плівки становить 0,3-0,6 мм, а значення швидкості 0,4-0,8 м/с. В результаті цього дослідження була сформована стратегія CFD-моделювання процесів у камері підшипника, що формує основу для вдосконалення моделі робочого процесу під час подальших експериментальних досліджень.

ЛІТЕРАТУРА

1. Kakimpa B., Morvan H., Hibberd S. The numerical simulation of multi-scale oil films using coupled VOF and Eulerian thin-film model. ASME Turbo Expo 2016: Turbomachinery Techn. Conf. and Expos., Seoul, South Korea. 2016. Vol. 1: Aircraft Engine. Fans and Blowers. Marine. P. 10. DOI: 10.1115/GT2016-56747.
2. Prabhakar A., Ambrose S., Morvan H. Numerical investigation of two phase flow in a dual drive booster. ASME Turbo Expo 2019: Turbomachinery Techn. Conf. and Expo., Phoenix, Arizona, USA. 2019. Vol. 1: Aircraft Engine; Fans and Blowers; Marine. P. 8. DOI: 10.1115/GT2019-90347.
3. M. Berthold et al. Multiphase CFD modeling of external oil flow from a journal bearing. ASME Turbo Expo 2018: Turbomachinery Techn. Conf. and Expo., Oslo, Norway. 2018. Vol. 1: Aircraft Engine. Fans and Blowers. Marine. P. 12. DOI: 10.1115/GT2018-77130.
4. Busam S., Glahn A., Wittig S. Internal bearing chamber wall heat transfer as a function of operating conditions and chamber geometry. J. Eng. Gas Turbines Power. 2000. Vol. 122, Issue 2. P. 7. DOI: 10.1115/1.483209.

ЛИСИЦЯ Олексій Юрійович – к. т. н., доцент; доцент кафедри аерокосмічної теплотехніки; Національний аерокосмічний університет ім. М.Є. Жуковського "ХАІ", вул. Чкалова 17, м. Харків, Україна, 61070. E-mail: a.lisitca@khai.edu. Номер ORCID: https://orcid.org/0000-0002-5679-8459

Наукові інтереси:

- CFD-моделювання теплогідравлічних процесів в двигунах та енергетичних установках.

КОВАЛЬОВ Артем Вікторович – аспірант; Національний аерокосмічний університет ім. М.Є. Жуковського "ХАІ", вул. Чкалова 17, м. Харків, Україна, 61070. E-mail: kovalev.205khai@gmail.com. Номер ORCID: https://orcid.org/0000-0002-9493-9769.

Наукові інтереси:

- CFD-моделювання термогідравлічних процесів.

МИХАЙЛЕНКО Тарас Петрович – к. т. н., доцент, доцент кафедри аерокосмічної теплотехніки; Національний аерокосмічний університет ім. М.Є. Жуковського "ХАІ", вул. Чкалова 17, м. Харків, Україна, 61070. E-mail: t.mykhailenko@khai.edu. Номер ORCID: https://orcid.org/0000-0003-4708-673X.

Наукові інтереси:

- теплогідравлічні процеси в двигунах та енергетичних установках.

ПЕТУХОВ Ілля Іванович – к. т. н., доцент, доцент кафедри аерокосмічної теплотехніки; Національний аерокосмічний університет ім. М.Є. Жуковського "ХАІ", вул. Чкалова 17, м. Харків, Україна, 61070. E-mail: ilya2950@gmail.com. Номер ORCID: http:// orcid.org/ 0000-0002-0645-7912.

Наукові інтереси:

- термогідравлічні процеси у багатофазних потоках.

УДК 37.091.279.7:004 **СЕКЦІЯ 2**

МАЗОРЧУК М.С. ЛІСОВА Т.В.

МОДЕЛІ АНАЛІЗУ ЛАТЕНТНИХ ФАКТОРІВ У МІЖНАРОДНИХ ДОСЛІДЖЕННЯХ ОЦІНЮВАННЯ ЯКОСТІ ОСВІТИ (НА ПРИКЛАДІ PISA)

Міжнародні дослідження якості освіти відіграють важливу роль у розвитку суспільства. Вони надають не лише інформацію про стан освіти і досягнення учнів різних країн у порівнянні із іншими країнами світу, але й висвітлюють головні проблеми, які потребують вирішення в освіті. Метою таких міжнародних досліджень як PISA, TIMSS, PIRLS, PIAAC, NAEP та ін. є не тільки оцінити рівень грамотності чи компетентності учнів за різними напрямками, але й визначити фактори, які впливають на успішність учнів. Інформацію щодо цих факторів збирають, як правило, за допомогою анкет, які заповнюють учні, вчителі та керівники закладів освіти під час тестування. Більшість факторів пов'язані із різними аспектами життя дитини, які, часом, важко оцінити, оскільки вони є латентними (прихованими). Так, неможливо визначити рівень соціально-економічного статусу учнів, використовуючи лише одне питання в анкетах учнів щодо матеріальних статків батьків та обчислюючи частоту чи загальні міри центральної тенденції розподілу відповідей учасників тестування. Для отримання узагальненої інформації щодо таких факторів, які можуть бути пов'язані із успішністю учнів, використовують спеціальні показники та індекси, які формуються за низкою питань анкет і обчислюють за допомогою спеціальних моделей та методів. Це дозволяє порівнювати ці показники та індекси на рівні різних груп учасників тестування та на рівні країн. Метою даної роботи є огляд деяких моделей аналізу латентних факторів у міжнародних дослідженнях та особливостей їх використання на прикладі міжнародного дослідження PISA.

Україна у 2018 році вперше взяла участь в одному з найбільших міжнародних порівняльних досліджень – у міжнародному дослідженні якості освіти PISA (Programme for International Students Assessment). На сьогодні левову частку результатів України за підсумками PISA-2018 проаналізовано в національному[1] та міжнародному[2] звітах. Також багато матеріалів за окремими питаннями можна знайти на порталі Українського центру оцінювання якості освіти[3]. Учні проходили тестування за трьома основними доменами: читання, математика та науково-природничі дисципліни, а також відповідали на питання анкети[4].

Анкети для учнів у PISA-2018 базуються на структурі, яку описано у фреймворку [1]. Багато питань анкет було розроблено та об'єднано таким чином, щоб можна було виміряти латентні (приховані) фактори, які неможливо спостерігати безпосередньо (наприклад, вмотивованість учнів до навчання, економічне, соціальне та культурне становище учнів та ін.). До цих питань застосовувалися методи та алгоритми перетворення, які дозволили сконструювати індекси [2]. Ці індекси називають «похідними» змінними. Можна виділити три основні види похідних змінних:

– прості індекси, які побудовано за допомогою арифметичного перетворення або перекодування одного чи декількох питань анкет;

– індекси, які отримано за допомогою моделей IRT (Item Response Theory);

– складні оцінки, які обчислено за допомогою шкали IRT або інших методів та об'єднано за спеціальними алгоритмами в узагальнюючі індекси.

Прості індекси використовуються для отримання інформації про контекстуальні характеристики учнів, які легко інтерпретувати. Наприклад, дані щодо віку учнів, терміну навчання у закладах дошкільної освіти, тривалості занять у закладах освіти, наявності

[1] https://testportal.gov.ua/wp-content/uploads/2019/12/PISA_2018_Report_UKR.pdf.

[2]http://pisa.testportal.gov.ua/wp-content/uploads/2020/02/PISA2018_Mizhnarodnyj-zvit_ukr.pdf, https://www.oecd.org/pisa/publications/

[3] http://pisa.testportal.gov.ua/pisa-2018-ukrayina-v-czentri-uvagy/

[4] https://www.oecd.org/pisa/publications/pisa-2018-national-questionnaires.htm

© МАЗОРЧУК М.С. ЛІСОВА Т.В., 2021

комп'ютерної техніки вдома та ін. було отримано за допомогою простих питань і шляхом перекодування значень змінних використано у подальшому аналізі. Для аналізу латентних факторів у більшості випадків застосовувалися індекси, побудовані на основі моделей IRT.

Для простих дихотомічних питань кожна відповідь на *i*-е питання анкети *j*-го учня представляє собою функцію, яка описує ймовірність того, що учень *j* вибере відповідь 1 замість 0. Ця функція є однопараметричною моделлю (моделлю Раша [3]) для дихотомічних змінних, яка має вигляд:

$$P(X_{ij}=1|\theta_j,\beta_i)=\frac{\exp(\theta_j-\beta_i)}{1+\exp(\theta_j-\beta_i)} \quad (1)$$

де $P(X_{ij}=1)$ – ймовірність того, що учень *j* вибере відповідь 1 на питання *i*, θ_j – оцінена прихована ознака учня *j* та β_i – передбачуване розташування питання анкети *i* у цьому вимірі.

У випадку коли учень відповідає на питання з більш ніж двома категоріями *m* (наприклад, упорядковані відповіді за шкалою Лайкерта), цю модель можна узагальнити до часткової кредитної моделі [4] або моделі узагальненого часткового кредиту (GPCM) [5], яка є більш доречною в контексті PISA, оскільки вона дозволяє в межах однієї шкали розподіляти частоту відповідей на різні питання. Ця модель має вигляд:

$$P(X_{ij}=k|\theta_j,\beta_i,\alpha_i,d_i)=\frac{\exp(\sum_{r=0}^{k}D\alpha_i(\theta_j-(\beta_i+d_{ir}))}{\sum_{u=0}^{m_j}(\exp(\sum_{r=0}^{u}D\alpha_i(\theta_j-(\beta_i+d_{ir})))} \quad (2)$$

де $P(X_{ij}=k)$ позначає ймовірність того, що учень *j* набере *k* балів за відповіді на питання *i* з усіх можливих оцінок за цим питанням. Тобто, якщо питання контекстуальної анкети з кількома пунктами (категоріями відповідей), то визначається ймовірність того, що учень відповість позитивно на *k* пунктів з максимально можливих відповідей. θ_j , як і у дихотомічній моделі (1) позначає приховану вимірювану характеристику учня, параметр β_i дає загальне місце розташування на прихованому континуумі. Параметри d_{ir} та α_i є додатковими, які дозволяють отримати більш валідні моделі. D - коефіцієнт, який дорівнює 1.7.

Для отримання оцінок за кожним індексом у шкалі IRT аналізують разом усі відповіді учнів з урахуванням ваг, які відображають вибірку учнів за країнами. Потім дані трансформуються у розподіл із середнім значенням 0 та стандартним відхиленням 1, де середнє 0 та стандартне відхилення 1 відповідає країнам ОЕСР[5], які мають рівні ваги. Такий підхід дозволяє отримати індекси у стандартизованій шкалі, яка є метричною. Значення індексів за усіма країнами розподілено за нормальним законом. Індекси найчастіше приймають значення у межах від -3 до +3, де середнє значення 0 відповідає середньому рівню за країнами ОЕСР. Більше значення індексів відповідає більшій мірі прояву тієї чи іншої характеристики, а менше значення – меншій. У такий спосіб вдається не тільки отримати міру прихованих характеристик учнів, але й використовувати її для порівняння між різними групами учнів як на рівні країни, а також і між країнами.

Складні індекси обчислюються на основі простих індексів, які найчастіше розраховано в шкалі IRT з використанням додаткових методів агрегації даних. Так, наприклад, соціально-економічне становище учнів є широким поняттям, яке об'єднує багато різних напрямків. В PISA, соціально-економічний статус учня звичайно вимірюється на основі індексу, який відображає

[5] ОЕСР (Організація економічного співробітництва та розвитку) – це міжнародна організація, членами якої на сьогодні є 37 країн, які вважаються розвинені країни із високим доходом громадян та високим індексом людського розвитку.

економічний, соціальний і культурний статус учня (economic, social and cultural status (ESCS)). ESCS – це комплексна оцінка, яка будується на основі індикаторів трьох компонент, які розраховані за методом головних компонент [2]:

- найвищий рівень освіти батьків (PARED індекс);
- найвищий професійний статус батьків (HISEI індекс);
- домашні володіння (HOMEPOS індекс), який є приблизною мірою вимірювання родинного добробуту, що включає в собі наявність чи доступність важливих для домогосподарств країни речей, таких як передплата підписки на газету, наявність MP3 плеєра, підключення до високошвидкісного інтернету та інше (типи питань залежать від країн); кількість книжок у домі; інші освітні ресурси, які є вдома, наприклад, комп'ютер, який може бути використаний для виконання домашніх завдань чи специфічне освітнє програмне забезпечення.

Використання трьох компонент обґрунтовано тим, що соціально-економічний статус звичайно базується на трьох складових: освіті, професійному статусі і доходах. Пряме визначення рівня доходів не є можливим в PISA, тому питання стосовно існування предметів домашнього господарства використовуються як оцінка для визначення прихованої змінної добробуту родини, яку можна використовувати для аналізу і порівняння за різними групами учнів як в межах однієї країни, так і між країнами.

Таким чином, для вимірювання латентних (прихованих) змінних використовують підходи, які базуються на перетворенні розподілів відповідей учнів у спеціальні індекси на основі моделей IRT. Це дозволяє отримати не тільки стандартизовані метричні оцінки, які узагальнюють відповіді на учнів і дозволяють робити висновки про ступень прояву тієї чи іншої контекстуальної латентної характеристики учня, але й надає можливість порівнювати ці індекси і використовувати у подальшому аналізу вивчення впливу факторів на успішність учнів.

ЛІТЕРАТУРА

1. OECD (2017). PISA 2015 assessment and analytical framework: Science, reading, mathematic, financial literacy and collaborative problem solving. OECD Publishing, Paris. http://dx.doi.org/10.1787/9789264281820-en
2. OECD (2021). PISA 2018 Technical Report. Chapter 16. https://www.oecd.org/pisa/data/pisa2018technicalreport/PISA2018_Technical-Report-Chapter-16-Background-Questionnaires.pdf.
3. Rasch, G. (1960). Probabilistic models for some intelligence and attainment tests, Nielsen and Lydiche, Copenhagen, Denmark.
4. Masters, G. N. & Wright, B. D. (1997). The partial credit model. In W. J. van der Linden & R. K. Hambleton (eds.), Handbook of modern item response theory, Springer, New York.
5. Muraki, E. (1992). A generalized partial credit model: Application of an EM algorithm. ETS Research Report Series, Vol. 1992/1, i-30.

МАЗОРЧУК Марія Сергіївна – к. т. н., доцент; доцент кафедри кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; Український центр оцінювання якості освіти, вул. В. Винниченка, 5, м. Київ, 04053; e-mail: mariia.mazorchuk@karazin.ua; mazorchuk.mary@gmail.com; ORCID: 0000-0002-4416-8361.

Наукові інтереси:

– *аналіз даних, освітні вимірювання*.

ЛІСОВА Тетяна Володимирівна – к. ф.-м. н., доцент; доцент інформаційних технологій та аналізу даних; Ніжинський державний університет імені Миколи Гоголя, вул. Графська, 2, Ніжин, Україна, 16600; Український центр оцінювання якості освіти, вул. В. Винниченка, 5, м. Київ, 04053; e-mail: tan-lisova@ndu.edu.ua; lisotana@gmail.com; ORCID: 0000-0001-9962-9922.

Наукові інтереси:

– *аналіз даних, освітні вимірювання*.

УДК 004.94 **СЕКЦІЯ 2**

МАКСИМЕНКО В.Ю., ТОЛСТОЛУЗЬКА О.Г.

ДОСЛІДЖЕННЯ КОМП'ЮТЕРНОЇ МОДЕЛІ ВЕБ-СЕРВІСУ ПЕРЕВІРКИ ЗНАНЬ

Вступ

В даний час у світі накопичено значний досвід реалізації систем дистанційного навчання. Які розрізняються не тільки методологією, а й залежать від особливостей тієї чи іншої країни, в якій практикуються методи і підходи до організації дистанційної перевірки знань. Однак, в цілому, всі визнають, що майбутнє за дистанційним навчанням, не тільки в силу розвитку інформаційно-комунікаційних технологій, а й як наступний, закономірний етап розвитку освітніх систем.

Контроль, або перевірка результатів навчання, є обов'язковим компонентом процесу навчання. Вона має місце на всіх стадіях процесу навчання, але особливого значення набуває після вивчення будь-якого розділу програми і завершення ступеня навчання. Суть перевірки результатів навчання полягає у виявленні рівня освоєння знань учнями, який повинен відповідати освітньому стандарту або кваліфікації за даною програмою, предмету.

На даний момент, у зв'язку з епідеміологічною ситуацією частка дистанційного навчання в процесі освіти постійно збільшується. Технічна реалізація системи дистанційного навчання являє собою досить складний програмно-апаратний комплекс.

Методи контролю знань

Найчастіше виділяють такі основні принципи контролювання учнів, яких необхідно дотримуватися в дистанційному контролі знань:

- принцип об'єктивності - пізнавальна діяльність повинна оцінюватися при мінімальному впливі суб'єктивного фактору;
- принцип демократичності - повинні створюватися рівні умови для всіх учнів, що проходять контроль;
- принцип масовості і короткочасності - контроль повинен бути організований так, щоб за якомога менший час здійснити перевірку знань у великої кількості випробовуваних.
- При проектуванні дистанційного контролю знань вживають спеціальні заходи для забезпечення достовірності даних здійснюваного контролю:
- доступ до ресурсів контролю знань здійснюється за допомогою індивідуальних логінів та паролів;
- використовуються різні шифри и кодування, щоб захистити тести від несанкціонованого доступу;
- усі контрольні заходи проводяться на базі сертифікованих центрів контролю знань, що мають доступ до мережі Інтернет;
- використовуються додаткові периферійні пристрої, наприклад, відеокамери;
- час на відповідь обмежується, а варіанти відповідей випадково перемішуються.

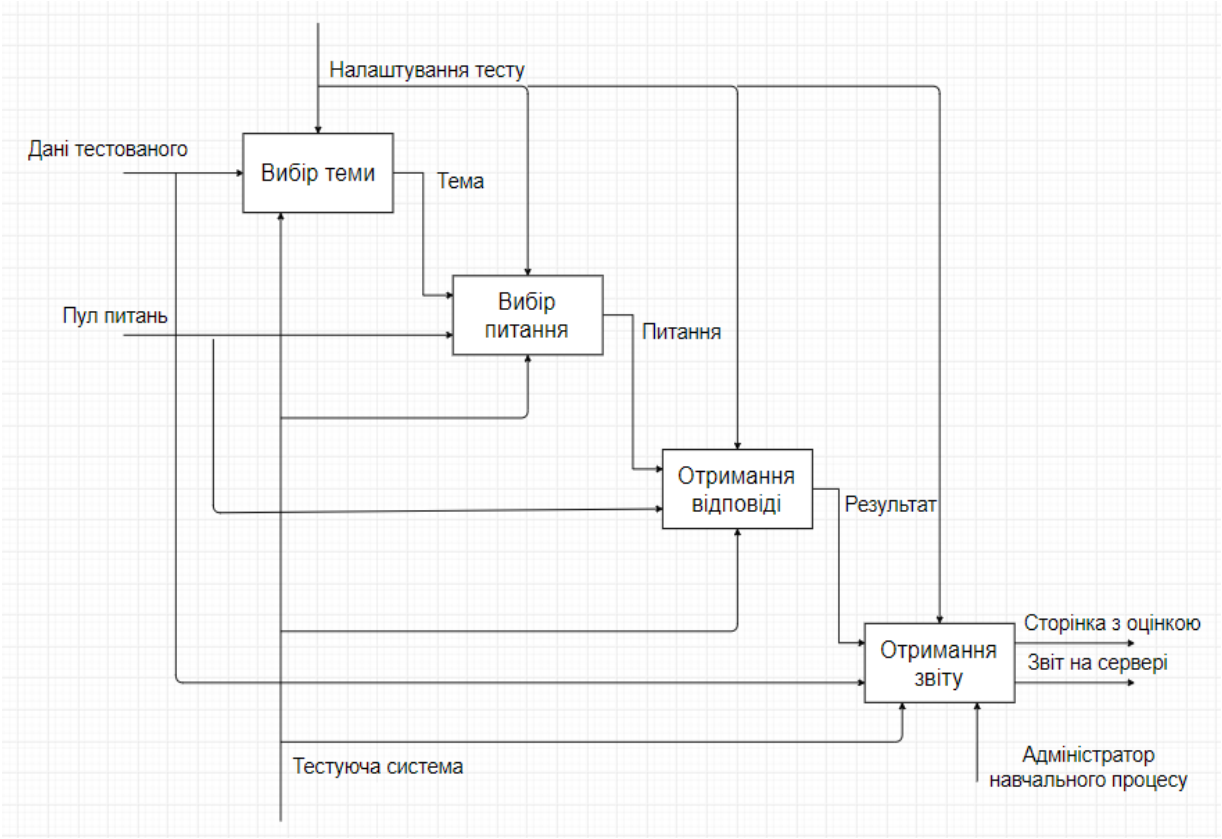
Однозначні і відтворювані оцінки здатні дати лише об'єктивні методи контролю якості знань учнів, що спираються на спеціально створені для цього матеріали - тести. Вони повинні бути розроблені по кожному рівню засвоєння досвіду. Тест - це засіб, який дозволяє виявити рівень і якість засвоєння. Висока похибка вимірювання тестових результатів не дозволяє говорити про високу надійність результатів вимірювання.

Найчастіше на освітніх сайтах можна побачити тести, що працюють в режимі реального часу. В цьому випадку тестований відповідає на питання тесту в режимі прямого діалогу з комп'ютерною програмою віддаленого сервера. Тести включають питання та варіанти відповідей (один з яких, як правило, вірний, а інші - помилкові). Учневі не потрібно записувати

© МАКСИМЕНКО В.Ю., ТОЛСТОЛУЗЬКА О.Г., 2021

відповідь, досить тільки клацнути курсором миші по потрібному рядку. У більшості тестів на перевірку знань дається 3-5 різних відповідей на одне завдання. Після виконання тестів на екрані з'являється результат - коментарі, оцінка, рекомендації щодо подальшої роботи і т. д.

На мал. 1 можна побачити IDEF0 діаграму такого тесту:



Мал. 1 IDEF0 діаграма тесту в режимі реального часу.

Найкращим можна вважати тест, в якому закладено широкий зміст і воно охоплює більш глибокі рівні знань. Розробники комп'ютерних тестів повинні дотримуватися наступних принципів:

- тест повинен відповідати цілям тестування;
- потрібно визначити значимість знань в загальній системі перевіряються знань;
- повинна бути забезпечена взаємозв'язок змісту і форми тесту;
- тестові завдання повинні бути правильними, з точки зору змісту;
- повинна дотримуватися репрезентативність змісту навчальної дисципліни в змісті тесту;
- тест повинен відповідати рівню сучасного стану науки;
- зміст тесту має бути комплексним і збалансованим;
- зміст тесту має бути системним, але разом з тим варіативним.

У процесі тестування необхідно реалізувати функції початку, призупинення, продовження і завершення тестування. Перед тим як почати роботу з тестом, учень повинен заповнити реєстраційну форму на екрані комп'ютера, отримати логін та пароль, який служить сполучною ланкою між реєстраційною інформацією та даними про сеанс тестування.

ЛІТЕРАТУРА

1. Ковальов В. М. Сучасні методології опису бізнес-процесів. Методологія IDEF0: Журнал" Консультант директора", № 12, Червень, 2004 р.
2. Ломовцева Н. В. Контроль навчальної діяльності в дистанційному навчанні.
3. Полат Є. С. Теорія і практика дистанційного навчання: Навчальний посібник для студ. вищ. пед. навч. закладів / Є. С. Полат, М. Ю. Бухаркина, М. В. Моїсеєва. - М. « "Академія", 2004. - 416 с.
4. Гаврилова Л. А. Дистанційна освіта. Електронні курси: Навчально-методичний посібник для викладачів. - Єкатеринбург: УГГУ, 2006. - 74 с.
5. Канаєв В. І. Дистанційне навчання: технологічні аспекти. - М.: Сучасний гуманітарний університет, 2004. - 192 с.
6. Раззаков, Ш. І. Контроль знань в системі дистанційного навчання / Ш. І. Раззаков, У. З. Нарзієв, Р. Б. Рахімов. - Текст: безпосередній / / Молодий вчений. — 2014. — № 7 (66). - С. 70-73.

МАКСИМЕНКО Владислав Юрійович – студент; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: xa11867733@student.karazin.ua; ORCID: 0000-0002-1052-2027.

Наукові інтереси:

– *комп'ютерні моделі веб-сервісів.*

ТОЛСТОЛУЗЬКА Олена Геннадіївна – професор кафедри теоретичної та прикладної системотехніки, д. т. н., с. н. с.; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: elena.tolstoluzka@karazin.ua; ORCID: 0000-0003-1241-7906.

Наукові інтереси:

– *технології автоматичного проектування паралельних програм.*

УДК 519.223 **СЕКЦІЯ 2**

МАКСИМУК А.Р., БАКУМЕНКО Н.С.

МОДЕЛЬ ІНФОРМАЦІЙНОЇ СИСТЕМИ КЛАСИФІКАЦІЇ ПАЦІЄНТІВ ЗА ДОПОМОГОЮ ЙМОВІРНІСНИХ ШТУЧНИХ МЕРЕЖ

Вступ

Процес ефективного аналізу одержуваної інформації в умовах наявності великого обсягу даних є досить об'ємним та складним, і тому є непідсильним для людини. Для вирішення цієї проблеми вимагаються нові методи. У даних умовах, автоматизація різних напрямків людської життєдіяльності шляхом впровадження обчислювальних алгоритмів торкнулася і процесу класифікації. У базу даного процесу лягла ідея про застосування математичних методів для збору, сортування та поділ на групи за певною ознакою об'єктів.

Класифікація - один з розділів машинного навчання, присвячений вирішення наступного завдання: є безліч об'єктів (ситуацій), розділених деяким чином на класи [1]. Основне завдання класифікації полягає в розбитті безлічі елементів даних на категорії або класи так, щоб всі елементи всередині кожного класу мали достатню кількість загальних ознак, що дозволяє знехтувати їх індивідуальними відмінностями [2].

Головним результатом роботи є класифікація станів медико-біологічної системи, за допомогою якої можна буде визначити, до якого класу (в нашому випадку «Пацієнт сталий» і «Пацієнт у критичному стані»), відноситься хворий, виходячи із зібраних даних лабораторних досліджень.

Мета

Метою даної роботи є розробка комп'ютерної моделі класифікації станів медико-біологічної системи за допомогою ймовірнісних штучних мереж.

Завдання

Щоб реалізувати поставлену мету необхідно вирішити наступні завдання: сформулювати постановку задачі класифікації ознак в медико-біологічних системах, провести поширений аналіз існуючих методів класифікації об'єктів, зробити детальний огляд обраного методу для вирішення задачі, скласти математичну модель вирішення задачі класифікації за допомогою ймовірнісних штучних мереж, розробити програмно-алгоритмічну модель класифікації системи, виконати тестування моделі та зробити аналіз отриманих результатів.

Об'єкт

Об'єктом дослідження є застосування та використання моделі класифікації за допомогою ймовірнісних штучних мереж в медичній діагностиці.

Предмет

Предметом дослідження є математичні методи класифікації.

Актуальність

У сучасному світі дуже багато завдань в медичних і біологічних системах є невирішеними. Однією з важливих задач, що зустрічаються в медичній практиці, є надійна діагностика захворювання. Деякі захворювання досить важко виявляються та діагностуються. Тому застосування сучасних інтелектуальних методів аналізу даних для підтримки прийняття рішень в медицині є актуальним [3].

Основна частина

Вирішення задачі класифікації є одним з найважливіших застосувань нейронних мереж [4].

Штучні нейронні мережі — це програмний прототип нейронних структур мозку людини. Мережі можуть змінювати тип переданих сигналів в залежності від електричних або хімічних

© МАКСИМУК А.Р., БАКУМЕНКО Н.С., 2021

сигналів, які в них передаються. Нейронна мережа у людському мозку - величезна взаємопов'язана система нейронів. В ній сигнал, який передається одним нейроном, може передаватися у тисячі інших нейронів. Нейромережний підхід сполучає у собі здатність комп'ютера до обробки чисел і здатність мозку до узагальнення і розпізнавання. Саме тому цей метод ефективний у задачах експертної оцінки. Нейронна мережа дозволяє обробляти велику кількість факторів, незалежно від їхньої наочності [5].

Штучні нейронні мережі імітують поведінку мозку у простішому вигляді [6]. При навчанні мережі пропонуються різні зразки образів із указівкою того, до якого класу вони відносяться. Зразок, як правило, представляється як вектор із його ознак. При цьому сукупність всіх ознак повинна однозначно визначати клас, до якого відноситься зразок. У випадку, якщо ознак недостатньо, мережа може співвіднести той самий зразок з декількома класами, що невірно. По закінченні навчання мережі можна пред'являти невідомі їй раніше образи й одержувати від неї відповідь про приналежність до визначеного класу. Топологія такої мережі характеризується тим, що кількість нейронів у вихідному шарі, як правило, дорівнює кількості обумовлених класів. При цьому установлюється відповідність між виходом нейроної мережі і класом, що він представляє. Коли мережі пред'являється якийсь образ, на одному з її виходів повинна з'явитися ознака того, що образ належить цьому класові. У той же час на інших виходах повинна бути ознака того, що образ даному класові не належить. Якщо на двох або більш виходах є ознака приналежності до класу, вважається що мережа «не упевнена» у своїй відповіді.

Якщо розглядати приклад розпізнавання букв, то вірна відповідь буде очевидною заздалегідь. В більш складних практичних задачах навчена нейромережа буде виступати як експерт, що володіє великим досвідом і здатен дати відповідь на важке запитання [7].

Яскравим прикладом такої задачі, якраз, і є медична діагностика, де мережа повинна враховувати велику кількість числових параметрів (аналіз крові, енцефалограма, результат МРТ, тиск і т.д.). Але ж, звичайно, "думку" мережі в цьому випадку не можна вважати остаточною.

Приведемо переваги та недоліки методу[8]. Переваги:

Зберігання інформації по всій мережі: інформація, така як у традиційному програмуванні, зберігається у всій мережі, а не в базі даних. Зникнення кількох відомостей в одному місці не заважає мережі функціонувати.

Здатність працювати з неповними знаннями: після навчання з штучними нейронними мережами дані можуть давати результат навіть із неповною інформацією. Втрата продуктивності тут залежить від важливості відсутньої інформації.

Наявність стійкості до вад: пошкодження однієї або декількох комірок нейронної мережі не заважає їй генерувати вихід. Ця функція робить мережі стійкими до вад.

Наявність розподіленої пам’яті: для того, щоб ШНМ могла навчатись, необхідно визначити приклади та **навчити** мережу відповідно до бажаного результату, показуючи ці приклади мережі. Успіх мережі прямо пропорційний вибраним екземплярам, і якщо подія не може бути показана мережі у всіх її аспектах, мережа може видавати помилкові результати

Можливість паралельної обробки: штучні нейронні мережі мають чисельну потужність, яка може **виконувати** більше одного завдання одночасно.

Недоліки штучних нейронних мереж:

Незрозуміла поведінка мережі: це найважливіша проблема ШНМ. Коли мережа виробляє прогнозуюче рішення, вона не дає поняття, чому і як. Це зменшує довіру до мережі.

Визначення належної структури мережі: не існує конкретного правила для визначення структури штучних нейронних мереж. Відповідна структура мережі досягається завдяки досвіду та методом спроб та помилок.

Складність подання проблеми в мережі: ШНМ можуть працювати з числовою інформацією. Проблеми повинні бути переведені в числові значення перед введенням в ШНМ. Механізм відображення, який буде визначено, безпосередньо впливатиме на продуктивність мережі.

Тривалість часу навчання мережі невідома: система навчається до досягнення певного значення помилки на вибірці.

Висновки

Розробка та впровадження інтелектуальних систем для діагностики та прогнозування захворювань людини є одним з актуальних напрямків досліджень в інтелектуальному аналізі даних. В даній статті стисло наведено результати та висновки вирішення задачі класифікації за допомогою методу логістичної регресії. Розглянуто ново-обраний метод класифікації. Поставлено мету та завдання на кваліфікаційну роботу магістра.

ЛІТЕРАТУРА

1. statistica.ru - Основы линейной регрессии [Електронний ресурс] Режим доступа: http://statistica.ru/theory/osnovy-lineynoy-regressii/
2. О.І. Черняк, П.В. Захарченко Інтелектуальний аналіз даних: підручник. К. : Знання, 2014. 599 с.
3. Артамонова Н. Сучасні тенденції розвитку наукової медичної інформації. Вісник Книжкової палати. 2009. Вип. 8. С. 27-30. URL: http://nbuv.gov.ua/UJRN/vkp_2009_8_12
4. machinelearning.ru – Классификация [Електронний ресурс] Режим доступа: http://www.machinelearning.ru/wiki/index.php?title=%D0%9A%D0%BB%D0%B0%D1%81%D1%81%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F
5. fs.onu.edu.ua - Нейронні мережі та генетичні алгоритми [Електронний ресурс] Режим доступа: http://fs.onu.edu.ua/clients/client11/web11/metod/imem/neyron.pdf.
6. basegroup.ru - Применение нейронных сетей для задач классификации [Електронний ресурс] Режим доступа: https://basegroup.ru/community/articles/classification
7. codeguida.com - Нейронні мережі - шлях до глибинного навчання [Електронний ресурс] Режим доступа: https://codeguida.com/post/739
8. Maad M. Mijwil. Artificial Neural Networks Advantages and Disadvantages [Електронний ресурс] / Maad M. Mijwil Maad. – 2018. – Режим доступу до ресурсу: https://www.linkedin.com/pulse/artificial-neural-networks-advantages-disadvantages-maad-m-mijwel/

МАКСИМУК Анастасія Родіонівна – студентка 5 курсу факультету комп'ютерних наук спеціальності «Автоматизація та комп'ютерно-інтегровані технології»; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: maksymuk202ku51@student.karazin.ua; ORCID: 0000-0001-8185-6297.

Наукові інтереси:

– впровадження методів машинного навчання в системну інженерію.

БАКУМЕНКО Ніна Станіславівна – к. т. н., доцент; доцент кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: n.bakumenko@karazin.ua; ORCID: 0000-0002-5802-8183.

Наукові інтереси:

– інтелектуальний аналіз даних.

УДК 539.534.9:523.23 **СЕКЦІЯ 4**

МАРЧЕНКО І.Г., АКСЕНОВА В.Ю., ЖИГЛО А.В., ТКАЧЕНКО В.І.

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ ПРИСКОРЕННЯ ДИФУЗІЇ ЧАСТИНОК В ПРОСТОРОВО-ПЕРІОДИЧНИХ СТРУКТУРАХ ЗОВНІШНІМИ ПОЛЯМИ

Вступ

Процес управління мікро та наноструктурою матеріалів багато в чому зводиться до можливості управління процесами дифузії [1]. Однак до теперішнього часу головним параметром такого управління була температура. З підвищенням температури дифузія зростає. При цьому активуються інші термічні процеси. З технологічної точки зору важливо мати можливість керувати дифузією певних типів дефектів або атомів без зміни температури середовища. Застосування зовнішніх полів різної природи безумовно може вирішити цю задачу.

Раніше було встановлено, що дифузійна рухливість частинок може істотно зростати під дією постійної сили як для недодемпфованих [2], так і передемпфованих систем [3]. У роботах [4] було показано, що під дією постійної зовнішньої сили коефіцієнт дифузії частинок у періодичній решітці може зростати на багато порядків у вузькому діапазоні сил, який визначається дисипативними властивостями середовища. У статті [5] були представлені аналітичні вирази для розрахунку посилення коефіцієнта дифузії в залежності від коефіцієнта тертя для недодемпфованих систем. У дослідженні [6] була побудована діаграма існування областей, на яких можна істотно контролювати дифузію. Було показано, що в таких областях реалізується явище так званої температурно-аномальної дифузії (ТАД). При ТАД коефіцієнт дифузії зростає зі зниженням температури. На перший погляд це суперечить інтуїтивним уявленням про процеси дифузії. Однак, варто зазначити, що подібне явище має місце тільки в системах, далеких від рівноваги.

Умови впливу постійної сили, як правило, складно реалізувати технічно. На практиці більш перспективно використовувати змінні у часі поля. В якості таких полів можуть виступати акустичні або електромагнітні поля.

Головна мета цього дослідження полягає в демонстрації реальної можливості посилення дифузії в кристалічних твердих тілах на багато порядків без зміни температури шляхом використання зовнішнього періодичного поля відповідної амплітуди та частоти. Для цього в якості модельної системи використовується одновимірна решітка, досліджуються залежності коефіцієнтів дифузії від амплітуди зовнішньої періодичної сили для різних частот коливань.

Методика моделювання

Рух броунівських частинок на одновимірній решітці під дією періодичної у часі сили $F(t)$ описувався рівнянням Ланжевена:

$$m\ddot{x} = -\frac{\partial}{\partial x}U(x) - \gamma\dot{x} + F_t(t) + \sqrt{2\gamma Q}\xi(t), \qquad (1)$$

де t – час; x – координата частинки; m – її маса; γ – коефіцієнт тертя; $\xi(t)$ – Гаусів білий шум із інтенсивністю, що дорівнює одиниці. Крапка зверху означає диференціювання за часом. Теплова енергія $Q = kT$, де k – стала Больцмана, T – температура.

Задля з'ясування фізичних причин посилення дифузії у випадку впливу періодичної сили, як і в роботах [7-8] розглянемо рух частинок під дією прямокутного періодичного поля:

$$F_t(t) = F sign(\sin(\omega t)), \qquad (2)$$

де ω – кутова частота зовнішньої сили, а F – її амплітуда.

Такий вигляд зовнішнього поля має перевагу перед синусоїдальним з точки зору простоти інтерпретації фізичних результатів, тому що при такому впливі ансамбль частинок

© МАРЧЕНКО І.Г., АКСЕНОВА В.Ю., ЖИГЛО А.В., ТКАЧЕНКО В.І., 2021

рухається в фіксованому потенціалі на кожному напівперіоді. Це значно спрощує аналіз результатів. Використання синусоїдальної залежності, як було показано у [8], не змінює основних фізичних результатів, а тільки варіює отримані чисельні значення.

Потенційна енергія частинки U в одномірній решітці дорівнювала:

$$U(x) = -\frac{U_0}{2}\cos\left(\frac{2\pi}{a}x\right), \tag{3}$$

де a – період одномірної решітки, а U_0 – висота потенційного бар'єра.

На частинку, що рухається, діє періодична сила з боку решітки F_{lat}:

$$F_{lat} = -\frac{\partial U}{\partial x} = F_0 \sin\left(\frac{2\pi}{a}x\right), \tag{4}$$

де $F_0 = \frac{\pi}{a}U_0$.

Параметри просторово-періодичного потенціалу були тими ж, що й у роботах [4-8]: $U_0 = 0.08$ еВ, $a = 2.0$ Å. Маса частинок відповідала масі водню та дорівнювала 1 атомній одиниці маси.

Стохастичні рівняння (1) для кожної частинки розв'язувалися чисельно з кроком по часу, який був менше 0.01 періоду власних коливань $\tau_0 = a\left(2m/U_0\right)^{1/2}$. Статистичне усереднення проводилося за ансамблем із кількістю частинок не менше $N = 5\cdot 10^4$. Початкові умови були тотожні з [4-8].

Під час аналізу результатів зручно перейти до використання безрозмірнісних величин, тоді рівняння (1) приймає більш простий вигляд:

$$\ddot{x}' = -\sin x' - \gamma' \dot{x}' + F'\left(t'\right) + \sqrt{2\gamma' Q'}\,\zeta(t'), \tag{5}$$

Безрозмірні одиниці пов'язані з силою, коефіцієнтом тертя та температурою таким чином:

$$\gamma' = \gamma\frac{\tau_0}{2\pi m}, \quad F' = \frac{F}{F_0}, \quad Q' = \frac{2kT}{U_0}. \tag{6}$$

У проведених дослідженнях використовувалось значення $\gamma' \approx 0.032$, таке ж, як у попередніх роботах [4-8].

Коефіцієнт дифузії обчислювався наступним чином. При кожному розрахунку визначався час t_{lin} досягнення лінійної залежності дисперсії від часу. Коефіцієнт дифузії на одномірній решітці визначався як ½ нахилу кривої $\sigma^2(t)$. Лінійна залежність отримувалася підгонкою даних моделювання за методом найменших квадратів при часі $t > 100\,t_{lin}$. Достовірність підгонки результатів у всіх випадках склала не менше 0.999.

Результати та обговорення

Розглянемо, як змінюється дифузійна рухливість частинок при зовнішньому періодичному впливі. Для з'ясування ступеня посилення коефіцієнта дифузії спочатку були розраховані коефіцієнти дифузії частинок $D_{lat}\left(T'\right)$ за відсутності зовнішнього впливу. На рис. 1 залежність $D_{lat}\left(T'\right)$ зображена порожніми маркерами. Вона має характерний для кристалічних матеріалів вигляд кривої Ареніуса $D_{lat}\left(T'\right) = D_0 e^{-\varepsilon/kT'}$, де ε – величина енергетичного бар'єра під час переходу частинки з однієї комірки одномірної решітки до іншої. Відповідно коефіцієнт дифузії вільних частинок експонентно зменшується зі зростанням зворотної температури.

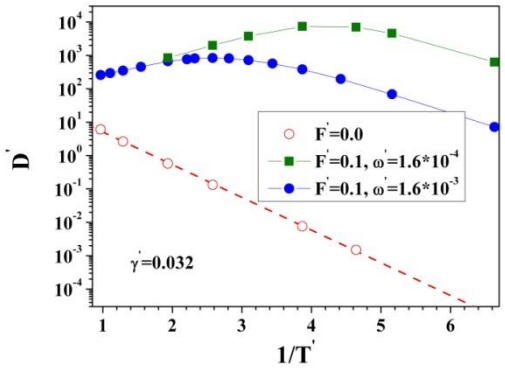


Рис.1. Залежності коефіцієнтів дифузії від зворотної температури. Порожнисті маркери – стала зовнішня сила; заповнені маркери – зовнішня періодична сила $F' = 0.1$ *(відповідає максимальному коефіцієнту дифузії за сталої сили), частота* $\omega' = 10^{-3}$. *Коефіцієнт тертя* $\gamma' = 3 \cdot 10^{-2}$.

Використання зовнішнього постійного поля ($\omega' = 0$) призводить до зростання дифузійної рухливості частинок. Ступінь цього посилення істотно залежить від величини зовнішньої сили. На рис. 2 заповненими маркерами наведена залежність коефіцієнта дифузії від величини прикладеної сили при постійній температурі термостата. Пунктирною лінією відзначений рівень коефіцієнта дифузії D_{lat} частинок у одномірній решітці за відсутності зовнішнього впливу. Штриховою лінією на цьому рисунку показано значення коефіцієнта дифузії у в'язкому середовищі $D_{vis} = k'T'/\gamma'$ за відсутності зовнішніх сил. Видно, що дифузія різко зростає у вузькому діапазоні прикладених сил. Максимальне значення D' досягається при $F' \approx 0.1$. На рис. 1 наведені температурні залежності коефіцієнта дифузії частинок при даному значенні амплітуди сили для двох різних частот. Із рисунка зрозуміло, що використання зовнішнього періодичного поля також збільшує дифузійну рухливість частинок на декілька порядків у широкому діапазоні температур. Більше того, у визначеному температурному інтервалі коефіцієнт дифузії зростає зі зменшенням температури. Це інтервал так званої температурно-аномальної дифузії. При зменшенні частоти зовнішнього впливу даний температурний інтервал розширюється. Тобто змінюючи частоту зовнішнього впливу, можна ефективно керувати коефіцієнтом дифузії при заданій температурі.

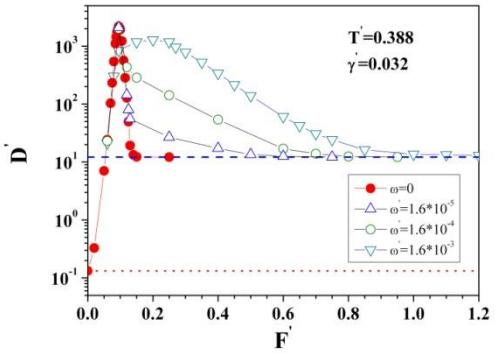


Рис. 2. Залежності коефіцієнтів дифузії від амплітуди коливань зовнішньої періодичної сили F' *для різних частот* ω. *Пунктирною лінією показано значення коефіцієнта дифузії у періодичній решітці за відсутності зовнішньої сили. Штриховою лінією показано значення коефіцієнта дифузії у в'язкому безструктурному середовищі.* $T' = 0.388$, $\gamma' = 3 \cdot 10^{-2}$.

Висновки

У роботі методами компьютерного моделювання вивчено посилення дифузії частинок у просторовій решітці під дією зовнішніх періодичних у часі полів. З'ясовано, що коефіцієнт

дифузії може бути збільшений на порядки при використанні поля відповідної амплітуди та частоти. Максимальне збільшення дифузії спостерігається при $\omega \to 0$. При великих амплітудах впливу в дослідженому діапазоні часто коефіцієнт дифузії прагне до коефіцієнта дифузії у в'язкому середовищі. Отримані результати відкривають перспективи створення нових технологій управління процесами дифузії.

ЛІТЕРАТУРА

1. H.J. Frost, M. F. Ashby, Deformation-mechanism maps : the plasticity and creep of metals and ceramics. Oxford: Pergamon press. 1982. 165 p.
2. Costantini G., Marchesoni F. Threshold diffusion in a tilted washboard potential. 1999. Europhys. Lett. Vol. 48. P. 491.
3. Reimann, P., Van den Broeck C., et. al. Diffusion in tilted periodic potentials: Enhancement, universality, and scaling, Phys. Rev. 2002. Vol. E65. p. 031104.
4. Marchenko I.G. , Marchenko I.I. Anomalous Temperature De-pendence of Diffusion in Crystals in Time-Periodic Ex-ternal Fields. 2012. Europhys. Lett. Vol. 100. p. 5005.
5. Marchenko I.G. , Marchenko I.I., Tkachenko V.I. Temperature-Abnormal Diffusivity in Underdamped Spatially Periodic Systems. 2017. JETP Letters. Vol. 106. No. 4. p. 242.
6. Marchenko I.G. , Marchenko I.I., Tkachenko V.I. Temperature-Abnormal Diffusivity in Tilted Periodic Potentials JETP Letters. 2019. Vol. 109. No. 10, p. 671.
7. Marchenko I.G. , Marchenko I.I. Anomalous Temperature Dependence of Diffusion in Crystals in Time-Periodic External Fields. 2012. JETP Letters. Vol. 95. No. 3. p. 137.
8. Marchenko I.G. , Marchenko I.I., Zhiglo A.V. Enhanced diffusion with ab-normal temperature dependence in underdamped space-periodic systems subject to time-periodic driving. 2018. Phys. Rev. 2018. Vol. E 97. p. 012121.

МАРЧЕНКО Іван Григорович – д.ф.-м.н., завідувач відділом "Фізики радіаційних технологій" Національного наукового центру «Харківський фізико-технічний інститут» (ННЦ ХФТІ), вул. Академічна, 1, Харків-108, Україна, 61108; професор кафедри фізики нетрадиційних енерготехнологій та екології Харківського національного університету імені В.Н. Каразіна (ХНУ), майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: march@kipt.kharkov.ua; ORCID: 0000-0003-1341-4950.

Наукові інтереси:

– комп'ютерне моделювання процесів у твердих тілах, стохастичні процеси, фізика поверхні, фізика радіаційних пошкоджень.

АКСЕНОВА Вікторія Юріївна – м.н.с. ННЦ ХФТІ, вул. Академічна, 1, Харків-108, Україна, 61108; аспірантка кафедри фізики нетрадиційних енерготехнологій та екології ХНУ ім. В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: aksenova@kipt.kharkov.ua; ORCID: 0000-0003-4231-7463.

Наукові інтереси:

– комп'ютерне моделювання процесів у твердих тілах, стохастичні процеси.

ЖИГЛО Андрій Валентинович– к.ф.-м.н., с.н.с. ННЦ ХФТІ, Академічна, 1, Харків-108, Україна, 61108; e-mail: azhiglo@gmail.com; ORCID: 0000-0003-0416-5796.

Наукові інтереси:

– астрофізика, комп'ютерне моделювання складних процесів.

ТКАЧЕНКО Віктор Іванович – д.ф.-м.н., директор НВК ВДЕРТ ННЦ ХФТІ, професор кафедри фізики нетрадиційних енерготехнологій та екології ХНУ ім. В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: tkachenko@kipt.kharkov.ua; ORCID: 0000-0002-1108-5842.

Наукові інтереси:

– фізика плазми, фізика радіаційних пошкоджень.

УДК 539.3 **СЕКЦІЯ 4**

МИРОНЕНКО М.Л.

ДОСЛІДЖЕННЯ ВІЛЬНИХ КОЛИВАНЬ РІДИНИ В ОБОЛОНКАХ, ЧАСТКОВО ЗАПОВНЕНИХ РІДИНОЮ, ПРИ РІЗНИХ ПАРАМЕТРАХ ПЕРЕВАНТАЖЕННЯ

Загальні відомості

Програма польоту ракет-носіїв включає кілька активних, незначних за тривалістю, та пасивних ділянок. Пульсації тяги двигуна, який працює на активній ділянці польоту, чинять дію на ракету, як замкнуту систему, та викликають коливання корпусу ракети та рідини в паливних магістралях [1]. Це може призвести до втрати стійкості та серйозних наслідків.

Політ і управління ракетою-носієм на активній ділянці траєкторії в реальних умовах проходять при впливі зовнішніх сил, зміні температур і щільності атмосфери, вітрових збурень, а також коливань рідини в паливних баках, що виникають в результаті всього спектра зовнішніх впливів [2].

Зміна швидкості ракети-носія до величини, необхідної для виведення космічного кораблю на орбіту (більше 8000 м/с, що перевищує першу космічну швидкість), за незначний проміжок часу активної ділянки польоту викликає суттєві прискорення. Таким чином, поздовжні перевантаження, що діють на ракету-носій на даній ділянці, можуть сягати кількох g [3] та впливати на стійкість.

При наявності вантажу всередині ракети визначено, що зменшення маси ракети-носія за рахунок вигорання палива найбільше впливає на коефіцієнт осьового перенавантаження, який змінюється нерівномірно на активній ділянці траєкторії та набуває максимального значення в кінці роботи двигуна ступені [4].

Проведення досліджень стійкості рідинних ракет-носіїв щодо поздовжніх коливань на активній ділянці польоту викликає труднощі в наземних умовах при стендових випробуваннях. Натурні випробовування потребують значних матеріальних та фінансових затрат, тому застосування методів математичного моделювання дозволяють уникнути подібних витрат та забезпечити вклад у вирішення задач поздовжньої стійкості ракет-носіїв [5].

Мета дослідження

Метою дослідження є створення методики для дослідження вільних коливань рідини в резервуарах при різних значеннях параметру перевантаження.

Виклад основного матеріалу дослідження

У роботі розглянуто пружну оболонку обертання, частково заповнену рідиною. Для моделювання руху рідини була розроблена математична модель, що базується на таких гіпотезах: рідина є ідеальною та нестисливою, рух рідини безвихровий. Досліджуємо лише малі коливання (лінійна теорія). Існує потенціал швидкості $\Phi(x,y,z,t)$, градієнт якого є швидкістю рідини.

Рівняння руху оболонкової системи запишемо в операторній формі

$$\boldsymbol{LU} + \boldsymbol{M}\ddot{\boldsymbol{U}} = p\boldsymbol{n} + \boldsymbol{Q}, \qquad (1)$$

де $\boldsymbol{L},\boldsymbol{M}$ – оператори пружних та масових сил;

p – тиск рідини на змоченій поверхні оболонкової конструкції, Н/м^2;

$\boldsymbol{n}$ – зовнішня одинична нормаль;

$\boldsymbol{Q}$ – сила збудження, Н.

Використовуючи рівняння руху і умови потенційності потоку, приходимо до інтеграла Коші-Лагранжа в такій формі, як в роботі [6].

У припущенні, що потік є безвихровим, рух ідеальної нестисливої рідини описується рівнянням Лапласа для потенціалу швидкості

$$\nabla^2\Phi = 0. \qquad (2)$$

© МИРОНЕНКО М.Л., 2021

Для визначення тиску обчислюємо потенціал швидкостей. Для цього сформулюємо граничні умови для потенціалу швидкостей на границях розрахункової області. На бічних поверхнях та на днищі виконуються умови непротікання. На вільній поверхні виконуються кінематична та динамічна умови:

$$\left.\frac{\partial \Phi}{\partial t} + a_x\left(t\right)x + a_z\left(t\right)\zeta + g\zeta\right|_{S_0} = 0, \quad \left.\frac{\partial \Phi}{\partial \boldsymbol{n}}\right|_{S_0} = \frac{\partial \zeta}{\partial t}, \quad \left.\frac{\partial \Phi}{\partial \boldsymbol{n}}\right|_{\sigma} = \frac{\partial w}{\partial t}, \tag{3}$$

де $w = \left(\boldsymbol{U}, \boldsymbol{n}\right)$ – нормальна складова переміщення оболонкової конструкції;

$\zeta = \zeta\left(x, y, t\right)$ – функція, що описує рух вільної поверхні.

Таким чином, розв'язуємо за виконання граничних умов систему диференційних рівнянь для визначення п'яти невідомих функцій $U_1, U_2, U_3, \Phi, \zeta$:

$$\begin{cases} \boldsymbol{L}\boldsymbol{U} + \boldsymbol{M}\ddot{\boldsymbol{U}} = -\rho_l\left(\dfrac{\partial \Phi}{\partial t} + a_x\left(t\right)x + a_z\left(t\right)z + gz\right) + \boldsymbol{Q}\left(t\right), \\ \\ \nabla^2 \Phi = 0. \end{cases} \tag{4}$$

Для отримання однозначного розв'язку системи рівнянь (4) з граничними умовами (3) додаємо умову Неймана. Систему диференціальних рівнянь (4) слід також доповнити умовами закріплення оболонкової конструкції, тобто умовами відносно вектор-функції U.

При дослідженні коливань рідини за умови низької гравітації виконаємо врахування поверхневого натягу. Динамічна гранична умова на вільній поверхні дозволяє враховувати поверхневий натяг, який стає визначальним фактором при дослідженні коливань оболонкової конструкції в умовах низької гравітації та приймає такий вигляд:

$$\left.\frac{\partial \Phi}{\partial t} + a_x\left(t\right)x + a_z\left(t\right)z + g\zeta - \frac{\sigma_0}{\rho_l}\Delta_S\zeta\right|_{S_0} = 0. \tag{5}$$

Для розв'язання задач про власні та вимушені коливання пружних оболонкових конструкцій з відсіками, що містять рідину, розроблено метод заданих форм. Отримано зв'язану систему диференціальних рівнянь відносно пружних переміщень конструкції та діючого тиску рідини. Для зображення розв'язків цієї системи використовуються три набори базисних функцій. Перший з них складається з власних форм коливань конструкції за відсутністю заповнювача і використовується для побудови гідро-пружних переміщень. Другий та третій набори базисних функцій отримані при побудові потенціалу швидкостей і функції, що описує зміни за часом вільної поверхні конструкції рідини. Потенціал швидкостей зображується сумою двох часткових потенціалів. Один з них відповідає власним коливанням рідини в жорсткому резервуарі при урахуванні сил гравітації; другий – відноситься до власних коливань пружної оболонки з рідиною без урахування гравітаційних сил.

Розглянуто метод зведення задачі визначення частот та форм коливань оболонкових систем при частковому заповненні рідиною до системи сингулярних рівнянь для визначення потенціалу швидкостей в задачі коливань рідини в жорсткій оболонці з урахуванням гравітаційної складової.

Виконано чисельне моделювання для визначення умов стійкості руху оболонок, частково заповнених рідиною, під дією зовнішніх навантажень, згідно з [7, 8, 9] та визначені на першому етапі частоти та форми коливань системи «оболонка-рідина». Визначаємо коливаннями рідини в жорстких оболонках. Запропонованим вище методом отримані частоти і форми коливань рідини для циліндричної і циліндрично-конічної оболонок обертання (рис. 1) при таких параметрах $H = 2$ м, $H_1 = H_2 = 1$ м, $R = 1$ м, $\beta = 60^o$. Визначено, що форма днища при таких

параметрах несуттєво впливає на частоти і форми коливань, а визначальним фактором є рівень заповнення оболонок *H*.

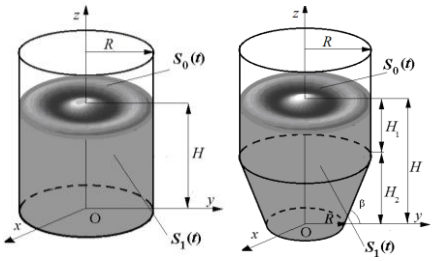


Рис.1 Оболонки, частково заповнені рідиною

Виконано врахування поверхневого натягу для визначення його впливу для циліндричної оболонки. Отримані результати чисельного моделювання свідчать про те, що при значних параметрах перевантаження вплив поверхневого натягу стає несуттєвим. Але зі зменшенням цього параметру (при низьких рівнях гравітації) поверхневий натяг стає домінуючим.

ЛІТЕРАТУРА

1. O'Neil, W.J. Project Galileo - The Jupiter Mission. In: Barbieri C., Rahe J.H., Johnson T.V. & Sohus A.M. (eds) The Three Galileos: The Man, the Spacecraft, the Telescope. Astrophysics and Space Science Library, Springer, Dordrecht. 1997, 220. P. 17-23.
2. Silveira, A.M., Stephens, D.G., Leonard, H.W. An Experimental Investigation of Liquid Oscillation in Tanks with Various Baffles, NASA Technical Note D-715, Washington: National Aeronautics And Space Administration, 1961. P. 5-24.
3. Abramson, H.N. Slosh Suppression, NASA Technical Report SP-8031, Washington: National Aeronautics And Space Administration, 1969. P. 1-23.
4. Nayfeh, Ali Hasan. Third-harmonic resonance in the interaction of capillary and gravity weaves, Journal of Fluid Mechanics, 1971, 48 (2). P. 385–395.
5. Goldrick, L. F. Mc. An experiment on second-order capillary gravity resonant wave interaction, Journal of Fluid Mechanics, 1970, 40(2). P. 251–271.
6. Strelnikova E., Kriutchenko D., Gnitko V., Degtyarev K., Boundary element method in nonlinear sloshing analysis for shells of revolution under longitudinal excitations. Engineering Analysis with Boundary Elements. 2020. Vol. 111. P. 78-87. DOI: doi.org/10.1016/j.enganabound.2019.10.008. URL: http://www.sciencedirect.com/science/article/pii/S0955799719306149 (дата звернення: 22.11.2020).
7. Strelnikova A., Choudhary N., Kriutchenko D., Gnitko V., Tonkonozhenko A. Liquid vibrations in circular cylindrical tanks with and without baffles under horizontal and vertical excitations Engineering Analysis with Boundary Elements. 2020. Vol. 120. P. 13-27. DOI: doi.org/10.1016/j.enganabound.2020.07.024. URL: http://www.sciencedirect.com/science/article/pii/S0955799720302009 (дата звернення: 22.11.2020).
8. Butikov E. Analytical expressions for stability regions in the Ince–Strutt diagram of Mathieu equation. American Journal of Physics, 2018, Vol.86. P. 257–267. DOI: https://doi.org/10.1119/1.5021895. URL: https://aapt.scitation.org/doi/pdf/10.1119/1.5021895 (дата звернення: 25.11.2020).
9. Kolukula S. S., Chellapandi P. Dynamic stability of plane free surface of liquid in axisymmetric tanks, Advances in Acoustics and Vibration, 2013. P. 16. DOI: 10.1155 / 2013/298458. URL: https://www.researchgate.net/publication/258391808_Dynamic_Stability_of_Plane_Free_Surfa Su_of_Liquid_in_Axisymmetric_Tanks (дата звернення: 25.11.2020).

МИРОНЕНКО Марія Леонідівна – аспірант, Інститут проблем машинобудування ім. А.М. Підгорного НАН України, м. Харків; тел.: (098) 931-85-19; e-mail: mariamyronenko87@gmail.com; ID ORCID: 0000-0002-0266-4463.

Наукові інтереси:

– чисельні методи моделювання.

УДК 004.056.55 **СЕКЦІЯ 5**

МІГАЛЬ Д.О., БОБУХ В.А.

ВАРІАЦІЇ ПОСТКВАНТОВОГО МЕТОДУ ЕЛЕКТРОННОГО ПІДПИСУ RAINBOW

Вступ

Розвиток технологій неспинно змінює вигляд навколишнього світу та впливає на всі сфери життя людей. Криптографія це одна з ланки областей, в якій недостатньо рухатись у ритмі з часом, необхідно мати фору для забезпечення інформаційної безпеки на декілька десятиліть вперед. Наразі, технології наблизилися до квантової межі що, очевидно, змінить підхід до криптографії, обчислювальних систем та методів захисту інформації. Цей факт був очевидним ще наприкінці минулого століття, а у 2016 Національний інститут стандартів та технологій (NIST) створив конкурс, метою якого є розробка нових методів шифрування та створення електронного підпису, що зможуть забезпечити безпеку у постквантову еру.

Метою цієї доповіді є аналіз одного з трьох фіналістів конкурсу NIST США – Rainbow, а також його варіацій.

Метод постквантового підпису Rainbow

Rainbow – це багатовимірна схема підпису з багаторівневою конструкцією. Вона являє собою розвинену ідею схеми підпису Unbalanced Oil-Vinegar (UOV). Додаткова структура, що накладається рівнями Rainbow, збільшує кількість криптографічних перетворень, що призводить до підвищення ефективності схеми. До переваг Rainbow можна віднести високу швидкість генерування підпису та малу довжину підпису. Натомість, довжина відкритого ключа є основним недоліком цієї схеми.

Зважаючи на довжину ключа, Rainbow хоч і є дуже перспективним методом, проте не підходить як алгоритм підпису загального призначення. Зокрема, великі відкриті ключі роблять ланцюги сертифікатів надзвичайно великими. Однак є варіанти використання, в яких не потрібно занадто часто надсилати ключі. Для таких додатків Rainbow пропонує невеликі та швидкі підписи. Якщо аналізувати єдиний інший підпис-кандидат з подібним профілем продуктивності – GeMSS, то він має значно більші ключі, і має проблеми з реалізацією на маленьких та не дуже потужних пристроях. Дослідники NIST [1] відзначили розрив між продуктивністю та теоретичною складністю для кількох напрямків атаки, що стосуються схеми Rainbow. Під час другого раунду були опубліковані деякі більш строгі теоретичні аналізи (а також нові алгоритми) відомих атак. Зокрема, виявилось, що налаштування параметрів необхідне для всіх наборів параметрів для досягнення заявленого рівня безпеки. Перш, ніж Rainbow буде готовим до стандартизації, її параметри повинні бути скориговані, щоб забезпечити відповідність заявленим цілям безпеки. Крім того, NIST віддає перевагу алгоритмам без ліцензійних платежів, щоб стимулювати їх широке розповсюдження.

Схема підпису Unbalanced Oil-Vinegar

Схема Unbalanced Oil-Vinegar (UOV), що є основою Rainbow, є модифікованою версією схеми Oil та Vinegar, розробленої Дж. Патаріном. Ця схема відноситься до групи багатовимірної криптографії. Назва була отримана через використання двох типів змінних: «oil» і «vinegar», що формують відкритий ключ. При генерації підпису змінні різних типів не змішуються, а лише перемножуються, подібно до «маслу» і «оцту». Безпека підпису даної схеми заснована на рішенні NP-повної задачі [2] (тобто задачі, що належить до класу NP та всі задачі з класу NP можна звести до неї за поліноміальний час). Для створення і перевірки підписів необхідно вирішити систему мінімальних квадратичних рівнянь. Рішення m рівнянь з n змінними є NP-складною задачею, що означає, що проблему майже напевно важко ефективно вирішити у гіршому випадку, навіть при використанні квантового комп'ютера. Задача стає простішою, якщо значення m сильно відрізняється від n, в середньому випадку – коли m і n майже рівні – проблема вважається складною, навіть при використанні квантових обчислень.

© МІГАЛЬ Д.О., БОБУХ В.А., 2021

Істотним недоліком UOV є те, що розмір ключа може бути великим. Зазвичай n – кількість змінних, вибирається як подвійне m – кількість рівнянь. Для кодування коефіцієнтів усіх цих рівнянь у ключі потрібно значний принаймні 200 кілобайт для системи, яка забезпечить безпеку на рівні з алгоритмом електронного підпису або алгоритмом, що заснований на еліптичних кривих.

Варіанти схеми підпису Rainbow Cyclic Rainbow

Схема підпису «Cyclic Rainbow», що була запропонована Петцольдом [3]. У Cyclic Rainbow використовувалась певна циклічна структуру всередині відкритого ключа Rainbow. Однак, уважно розглянувши цю основну ідею, автори зазначили, що процес генерації відкритого ключа з особистого ключа є частково оборотним процесом. Тобто існує вірогідність обрати випадковим чином частини відкритого ключа та частини особистого ключа, а потім обчислити з цього решту відкритого ключа та особистий ключ. Оскільки цей процес є оборотним, можна легко переконатися, що ця зміна у виробництві ключів не впливає на безпеку.

CZ-Rainbow

Крім стандартної схеми Rainbow, на другому етапі випробувань NIST було запропоновано варіант під назвою CZ-Rainbow або «циркуменітальна веселка». У цьому варіанті було змінено процес генерації ключів Rainbow: замість обчислення відкритого ключа із закритого ключа, фіксуються основні частини відкритого ключа, а потім обчислюємо центральне перетворення Rainbow із відкритого ключа.

Схема CZ-Rainbow базується на цій ідеї змінити процес генерації ключів Rainbow. Для того, щоб підкреслити той факт, що CZ-Rainbow змінює спосіб генерації ключів Rainbow і, що він взагалі не використовує циклічні матриці, було вирішено використати нову назву CZ-Rainbow.

Оскільки частину відкритого ключа в будь-якому випадку можна обрати випадковим чином, ключовою ідеєю схеми CZ-Rainbow є використання криптографічного генератора псевдовипадкових чисел для генерації цієї частини відкритого ключа. Щоб створити пару ключів CZ-Rainbow, навмання обираються два лінійних відображення S і T і створюється частина відкритого ключа з невеликого seed-значення S_{pub} за допомогою ГПВЧ. З цього обчислюється центральне відображення F схеми Rainbow, а також відсутні частини відкритого ключа. Тому, замість того, щоб зберігати весь відкритий ключ, ми можемо просто зберегти seed-значення S_{pub}, а також деяку невелику частину P_2 відкритого ключа. Роблячи це, ми можемо зменшити розмір відкритого ключа схеми Rainbow до 70%.

Однак алгоритм генерації ключів циклічної Rainbow є недостатньо ефективним. Тому автори розробили новий алгоритм генерації ключів для схеми CZ-Rainbow, який лише трохи повільніший за стандартний процес генерації ключів. Однак під час процесу верифікації ми повинні розпаковувати відкритий ключ перед його оцінкою, і через це процес верифікації схеми значно сповільнюється.

Rainbow із «стисненим» ключем

Окрім стандартного та CZ варіантів, авторами також було запропоновано варіант Rainbow із «стисненим» ключем. Процес генерації ключів у цьому варіанті працює так само, як і у CZ-Rainbow, тільки обчислене центральне відображення не зберігається. Замість цього особистий ключ складається з двох seed-значень S_{pub} і S_{priv}, які використовувались для створення двох матриць S і T. Тому під час генерації підписів нам доведеться знову запустити процес генерації ключів, щоб сформувати центральне відображення з S_{pub} і S_{priv}. Це сильно уповільнює процес генерації підписів, проте значно зменшує розмір особистого ключа.

Висновки

1. У роботі наведено постквантовий метод електронного підпису Rainbow, що є розвинутою та покращеною схемою Unbalanced Oil та Vinegar (UOV). Rainbow є представником криптографії на багатовимірних перетвореннях. Головною перевагою цього методу є розмір електронного підпису та час його формування, проте через великий розмір ключа цю схему

рекомендують використовувати тільки для специфічних завдань, де розмір ключів не є критичним. Незважаючи на недоліки метод є признаним NIST та є одним із фіналістів.

2. Атаки, направлені на цей метод базуються на атаках на UOV, а тому вже добре вивчені і за останні десять років жодних нових недоліків системи знайдено не було. Багаторівнева структура Rainbow забезпечує більший рівень безпеки [4]. Експериментально було доведено, що оптимальною кількістю рівнів є u=2. Саме вона є ідеальним співвідношенням між складністю обчислювання і приростом рівня безпеки.

3. Зважаючи на обмеження викликане великим розміром ключа, використання методу має обмеження. Rainbow може використовуватися у системах, де не має необхідності часто пересилати відкритий ключ. Це можуть бути, наприклад, протоколи транспортного рівня, Secure та Verified Boot або забезпечення прозорості сертифікатів.

Зважаючи на вищевказані переваги – простоту, ефективність, безпечність, невеликий розмір підпису та скромні вимоги до обчислювальної системи, Rainbow має великі перспективи у майбутньому.

ЛІТЕРАТУРА

1. Jintai Ding Rainbow / Jintai Ding, Ming-Shing Chen, Albrecht Petzoldt, Dieter Schmidt, Bo-Yin Yang //. URL: https://csrc.nist.gov/CSRC/media/Presentations/Rainbow/images-media/Rainbow-April2018.pdf (Last accessed: 20.03.2021).
2. M. R. Garay, D. S. Johnson: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman, 1979.
3. Rainbow Signature. URL: https://www.pqcrainbow.org/ (Last accessed: 19.03.2021).
4. D. Smith-Tone, R. Perlner: Rainbow Band Separation is Better than we Thought. National Institute of Standards and Technology, USA, University of Louisville, USA.

МІГАЛЬ Демид Олексійович – аналітик з систем захисту інформації АТ "Інститут інформаційних технологій", вул. Бакуліна, 12, Харків, Україна, 61022; e-mail: demidmgl@gmail.com.

Наукові інтереси:

– *постквантова криптографія, системи захисту інформації.*

БОБУХ Всеволод Анатолійович – к.т.н., начальник відділу апаратних засобів захисту інформації АТ "Інститут інформаційних технологій", вул. Бакуліна, 12, Харків, Україна, 61022; e-mail: bobukhv@iit.kharkov.ua.

Наукові інтереси:

– *системи апаратного захисту інформації, захист інформації.*

УДК 004.2 **СЕКЦІЯ 2**

МОРОЗ О. Ю., ТОЛСТОЛУЗЬКА О. Г.

АНАЛІЗ ЗАСОБІВ ТЕХНОЛОГІЙ ВЕРИФІКАЦІЇ ПАРАЛЕЛЬНИХ ПРОГРАМ

Актуальність задачі верифікації програмного забезпечення

Технології обчислень стали всюдисущими: від телефонів мобільних до суперкомп'ютерів. І саме такі комп'ютерні системи за рахунок комп'ютерного моделювання і паралельних обчислень дозволяють «заглянути за горизонт» в багатьох сферах досліджень. Програмне забезпечення є в основі майже будь-якої інфраструктури, і задача якісного його забезпечення на сьогодні актуальна як ніколи. Практично всі компанії вже використовують у своїй діяльності штучний інтелект, Інтернет речей, хмари, бізнес-аналітику, соціальні мережі та інше Традиційні інформаційні технології (ІТ) і вбудовані системи поступаються місцем повсюдного програмного забезпечення (ПЗ), а успіх цифрової трансформації підприємств залежить від роботи програмних систем, що відповідають усім галузевим вимогам щодо надійності та доступності сервісів.

Сьогодні для усунення ризиків, що пов'язані з людським фактором пропонується безліч інструментів верифікації коду(Рис. 1). При цьому важливо не тільки впровадити самі інструменти, але й розробити відповідні компетенції та вибудувати стратегію тестування. Потрібні розвинені можливості автоматичного виявлення критичних точок та дефектів. Сучасні організації вкладають близько 30% бюджету, що виділяється на ІТ, у забезпечення якості та тестування, що не дивно – більше половини всіх систем є критично важливими для бізнесу [1]. При цьому компаніям і організаціям необхідно максимально гнучко реагувати на зміни і впроваджувати формалізовані процеси та методи саме контрольованого випуску безпечного ПЗ.

Рис. 1 Інструменти для перевірки основних характеристик якості ПЗ [3]

У противагу засадам програмування послідовних обчислень паралельна програма породжує сукупність процесів обробки інформації, які протікають одночасно та є повністю незалежними та пов'язані між собою статистичними або динамічними просторово-часовими, або причинно-наслідковими відносинами. Паралельне програмування використовує переваги багатоядерних і багатопроцесорних комп'ютерів, тому поділ процесу на кілька потоків, що виконуються паралельно реалізовуються складніше. Саме тому необхідно піддавати його строгому аналізу та перевіряти за допомогою різних методів верифікації. Ці процеси постійно вдосконалюються, а також розробляються нові. Гарантоване обґрунтування правильності паралельних програм може бути отримано тільки за допомогою альтернативного підходу, принципово відмінного від тестування. Даний підхід називається верифікацією [2].

Верифікация проводиться практично завжди та виконується методом перевірки або звірення характеристик продукції з заданими вимогами, а результатом дій є висновок про відповідність чи невідповідность продукції.

© МОРОЗ О. Ю., ТОЛСТОЛУЗЬКА О. Г., 2021

Вибір методів верифікації

Зазвичай, для перевірки правильності проекту більшістю розробників програмних систем практикуються методи імітаційного моделювання і тестування. Але, гідною альтернативою імітаційного моделювання та тестування є методи формальної верифікації, бо формальна верифікація забезпечує вичерпний аналіз всіх можливих варіантів поведінки системи.

Для проектованої системи прийнято будувати її абстрактну модель (наприклад, кінцеву систему переходів), що застосовує інструментальни засоби верифікації моделей програми. Для апаратури та програмного забезпечення, як правило, застосовують динамічні логіки, тимчасові логіки і їх варіанти з нерухомими точками. Результатом таких обчислень алгоритму глобальної перевірки на моделі є безліч станів моделі, де специфікація виконується, а алгоритм локальної перевірки на моделі показує, чому формула не виконується. Такі специфікації особливо важливі для пошуку тонких помилок в складних системах переходів. Порівняно з іншими підходами до формальної верифікації програм, саме метод перевірки на моделі володіє двома чудовими перевагами: він повністю автоматичний, і його застосування не вимагає від користувача особливих знань в математичних дисциплінах. Основний недолік методу перевірки на моделі це «комбінаторний вибух», який виникає, коли в системі переходи в деяких компонентах виконуються паралельно. К. МакМіллан ще у 1987 р. показав, що, використовуючи, символьне уявлення графа переходів, можна верифікувати дуже складні системи. Символьне уявлення було засновано на впорядкованих діаграмах Бріана.

Методи Agile та безперервної розробки ПЗ, які спрямовані на підвищення якості та задоволення вимог користувачів, повинні спиратися на ефективні, зручні в застосуванні інструменти, що можуть увійти в арсенал як розробників ПЗ, так і користувачів. Вибір саме методів верифікації ПЗ залежить від моделі розробки (каскадна, V-модель, спіральна або ін.) та стандарту (ISO / IEC 12207 до: 2017, ISO / IEC 25000 SQUARE) (рис. 2).

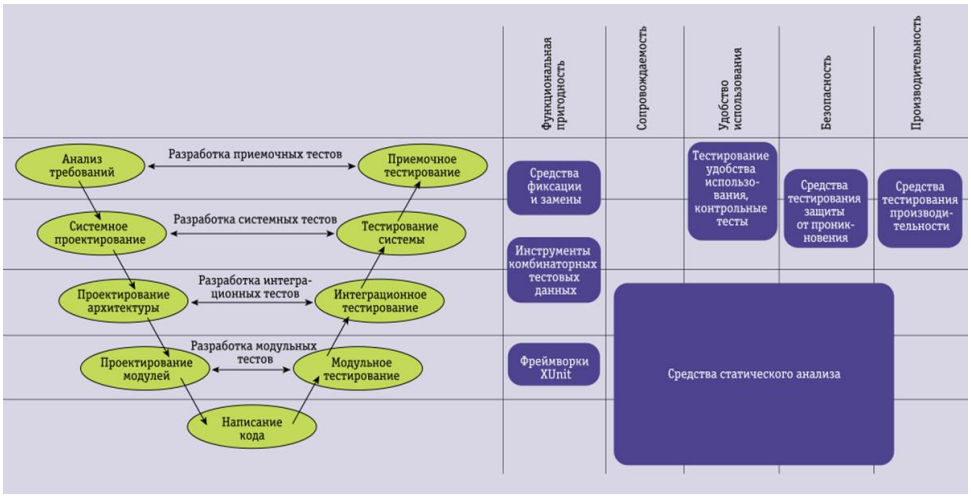


Рис. 2 Інструменти для перевірки основних характеристик якості ПЗ [3]

Технології та інструменти

За якість самого процесу розробки ПЗ відповідають стандарт ISO / IEC 12207, що регламентує процеси верифікації та валідації, а також V-модель, в рамках якої кожне завдання розробки ставиться у відповідність процесу тестування. Наприклад, модульний тест перевіряє відповідність вихідного коду низкорівневої архітектури, інтеграційні тести перевіряють сумісність раніше протестованих компонентів, системні тести дозволяють з'ясувати, чи відповідає повністю інтегрований продукт специфікаціям, а приймальні випробування – чи відповідає продукт очікуванням користувача. [3]

У стандартах серії ISO 25000, що відносяться до якості програмного продукту, виділяють наступні характеристики ПЗ: функціональна придатність; супровідність; відносна продуктивність та зручність використання. На рис. 2 зображені елементи забезпечення якості процесів і продуктів та відповідні інструменти. Дана схема може допомогти у виборі засобів верифікації, які можна задіяти на кожному з етапів V-моделі та для кожного параметра якості ПЗ.

Інструментів тестування так само багато, як і мов програмування.

Засоби XUnit застосовуються для перевірки правильності роботи кожного модуля, що розробляється. При цьому необхідно забезпечити максимальне покриття коду. Фреймворки XUnit отримали найбільш широке поширення серед технологій автоматизації верифікації, дозволяючи на спеціальних мовах описувати тестові ситуації та автоматично їх виконувати.

Засоби комбинаторного тестування генерують тестові дані, що дає можливість перевіряти коректність функціонування різних інтегрованих модулів.

Інструменти фіксації і заміни використовуються для перевірки коректності та повноти функціонування системи, в тому числі під час проведення приймальних тестів. Ці засоби реєструють взаємодії тестувальника з додатком, генеруючи тестові сценарії, які потім можна виконувати автоматично.

Інструменти контролю супроводу. Дозволяють аналізувати вихідний код і перевіряти його на відповідність правилам модульности, читання та ін.

Інструменти контролю зручності використання. Застосовуються для оцінки програмного продукту в процесі роботи з ним реальних користувачів. У той же час такі інструменти дозволяють проводити валідацію призначеного для користувача інтерфейсу без участі самих користувачів.

Засоби контролю безпеки. Дозволяють виявляти уразливості в системі і визначати, чи захищає вона дані при збереженні необхідної функціональності. Зокрема, засоби випробування на захист від проникнення імітують атаки на програмну систему або мережу за допомогою сканування та інших дій, спрямованих на пошук і використання слабких місць. Їх також називають інструментами етичного, або білого хакерства [5].

Таблиця 1. Засоби безперервної веріфікації та валідації [3] .

Непрерывная верификация и валидация	Инструмент	Технология	Описание	Лицензия	Популярность
	Cast Highlight	Поддерживает 18 языков программирования.	Интегрируется с остальными инструментами, участвующими в цикле разработки, позволяя оценивать и отслеживать качество портфеля приложений, обнаруживать уязвимости и создавать тесты.	Коммерческая	3
	Checkmarx Software Exposure Platform	Поддерживает более 20 языков программирования и написания скриптов.	Набор инструментов, позволяющих встраивать в циклы DevOps процессы обеспечения безопасности ПО, в том числе распознавания, приоритизации и устранения брешей.	Коммерческая	2
	Codacy	Поддерживает более 20 языков программирования.	Обеспечивает контроль качества и безопасности кода, интегрирует соответствующие механизмы в процесс разработки. Выполняет мониторинг изменений по мере обновления версий продукта, позволяя задавать стандарты качества и настраивать пороги срабатывания.	Есть бесплатная и коммерческая версии	1
	Coverity Scan	Java, Си/C++, C#, JavaScript, Ruby, Python.	Облачный сервис статического анализа, созданный для сообщества Open Source. Интегрирован с GitHub и Travis CI, обеспечивает непрерывную верификацию и валидацию.	Есть бесплатная и коммерческая версии	2
	Kluwan	Более 30 языков программирования.	Позволяет анализировать сопровождаемость, эффективность, безопасность, переносимость и надежность программного продукта. Встраивается в цикл разработки, обеспечивая непрерывную верификацию и валидацию. Помогает планировать исправления и прогнозировать соответствующие затраты времени.	Коммерческая	3
	SonarQube	Более 20 языков.	Один из самых широко используемых инструментов непрерывной оценки качества ПО. Позволяет оценивать сопровождаемость, безопасность, надежность, распознает проблемные участки кода, уязвимости и ошибки. Встраивается в цикл разработки, обеспечивая оценку качества изменений кода.	Бесплатно для некоторых языков	3
	Squore	Более 20 языков.	Получает данные от внешних инструментов — средств проверки на основе правил, анализа тестового покрытия, запроса изменений, конфигурационного управления и управления требованиями. Анализирует результаты работы таких средств совместно с собственными данными, позволяет создавать информационные панели и отчеты, помогает планировать изменения.	Коммерческая	1

Популярность: 1 — низкая, 2 — средняя, 3 — высокая.

Засоби перевірки продуктивності. Дозволяють з’ясувати, наскільки швидко система працює під навантаженням (табл. 1). Ці засоби не призначені для пошуку дефектів в додатку, а застосовуються для оцінки вимірних характеристик: часу відгуку, пропускної спроможності та ін.

Засоби безперервної перевірки та затвердження. Принцип безперервного забезпечення якості передбачає тісну інтеграцію процесів верифікації та валідації на всіх етапах розробки. Типовий приклад – розробка з орієнтацією на тестування, де планові показники якості задаються ще до розробки ПЗ. Однією з основ DevOps, все більшого значення надається інструментам, що допомагають не тільки автоматизувати перевірки та затвердження, а й інтегрувати ці процеси в цикл розробки. Такі інструменти дозволяють перевіряти характеристики якості в рамках середовищ безперервної інтеграції Jenkins, Travis CI, Bamboo, GoCD, Ansible та ін. Потрібно

вбудувати верифікацію в життєвий цикл ПЗ, забезпечивши автоматичне, прозоре для розробників виконання відповідних процесів. Такі інструменти перевіряють кілька характеристик якості та дозволяють отримати дані для глобальної візуалізації та оцінки. В рамках процесів безперервної перевірки та затвердження також можуть застосовуватися інструменти модульного тестування, фіксації та заміни і статичного аналізу [4].

Для управління діями, що виконуються за допомогою всіх перерахованих видів інструментів, і контролю над процесом верифікації в цілому, застосовуються засоби управління тестовими випадками: Test Link, Test Rail, Microfocus Quality Center, VSTS, IBM Rational Quality Manager, XStudio та ін. [5].

Висновки

У сучасному світі все більше стає ризиків, пов'язаних з ростом кількості загроз кібербезпеки і проблемами, які обумовлені незручністю використання додатків. Це означає, що потрібно прискорити розвиток технологій верифікації. Уникнути виявлення зловмисниками критичних вразливостей і появи нових атак після випуску продукту допоможуть безперервна верифікація та валідація, які повинні стати основою стратегії забезпечення якості ПЗ. Знадобляться механізми, які будуть перешкоджати запуску систем у разі, якщо на них не встановлені найсвіжіші оновлення програмного забезпечення. У число таких систем входять виробничі лінії та обладнання, яке потребує особливо високого рівня безпеки. Вибір систем перевірки визначається багатьма факторами. Залежно від середовища розробки, яке застосовується в різних організаціях, відповідні процеси мають свої особливості та будуються на базі різних сполучень інструментів. При цьому важливо не тільки впровадити самі інструменти, а також розробити відповідні компетенції та вибудовувати стратегію верифікації. Для усунення ризиків, пов'язаних з людським фактором, нам потрібні розвинені можливості автоматичного виявлення критичних точок та дефектів. Зауважимо, що широкий вибір інструментів – це важливо, але головне – створення культури забезпечення якості ПЗ і придбання відповідних навичок.

ЛІТЕРАТУРА

1. Ball T., Bounimova E. , Cook B., Levin V., Lichtenberg J., McGarvey C., Ondrusek B., Rajamani S. K., Ustuner A. Thorough Static Analysis of Device Drivers. In Proc. of EuroSys 2006. ACM SIGOPS OperatingSystems Review, 2006. P. 74.
2. Broy M., Jonsson B., Katoen J.-P., Leucker M. , Pretschner A. Model Based Testing of Reactive Systems. LNCS 3472, Springer, 2005. P. 273.
3. Capgemini. MicroFocus and Sogeti. World quality report. 2018. [Online]. URL: https:// www.capgemini.com/service/world-quality-report-2018-19/(дата обращения: 16.01.2021).
4. Moises Rodriguez, Mario Piattini, Christof Ebert, Software Verification and Validation Technologies and Tools. IEEE Software, March–April 2019 : IEEE Computer Society. All rights reserved. Reprinted with permission.
5. Родригес М., Пьяттини М., Эберт К. Верификация и валидация ПО: технологии и инструменты. Open Systems : DBMS 2020, N. 2 Vol. 28.

МОРОЗ Ольга Юріївна – старший викладач кафедри теоретичної та прикладної системотехніки факультету комп'ютерних наук; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 6, Харків, Україна, 61022; e-mail: o.moroz@karazin.ua; ORCID: 0000-0002-4920-4093.

Наукові інтереси: *технології автоматичного проектування паралельних програм.*

ТОЛСТОЛУЗЬКА Олена Геннадіївна – д.т.н., с.н.с., професор кафедри теоретичної та прикладної системотехніки факультету комп'ютерних наук; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 6, Харків, Україна, 61022; e-mail: elena.tolstoluzka@karazin.ua; ORCID: 0000-0001-2741- 180.

Наукові інтереси: *технології автоматичного проектування паралельних програм.*

УДК 004.021 **СЕКЦІЯ 2**

НЄВЄЖИНА В.Ю., СТРІЛЕЦЬ В.Є.

МОДЕЛІ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ НЕЧІТКОЇ ЛОГІКИ

У сучасному світі інформатизація зачіпає практично всі сфери діяльності людини. Використання сучасних інформаційних технологій стає критичним фактором розвитку більшості галузей знання і областей практичної діяльності, тому розробка та впровадження інформаційних систем є однією з найактуальніших завдань.

Для побудови систем підтримки прийняття рішень для складних нелінійних, погано формалізованих об'єктів часто застосовують методи і алгоритми на основі нечіткої логіки [1, 2]. Ці методи принципово відрізняються від звичайних класичних методів автоматизації «людським» підходом і «людськими» прийомами вирішення завдань управління.

Теорія нечітких множин, основні ідеї якої були запропоновані американським математиком Лотфі Заде [3] більше сорока років тому, дозволяє описувати якісні, неточні поняття і наші знання про навколишній світ, а також оперувати цими знаннями з метою отримання нової інформації. Засновані на цій теорії методи побудови інформаційних моделей істотно розширюють традиційні сфери застосування комп'ютерів і утворюють самостійний напрямок науково-прикладних досліджень, який отримав спеціальну назву – нечітке моделювання.

Нечітка логіка є одним з найбільш перспективних напрямків сучасної теорії управління та прийняття рішень. У світі щорічно виходять сотні книг і десятки спеціалізованих журналів, присвячених, як теорії нечіткої логіки, так і питань її застосування, випускаються спеціальні нечіткі контролери та мікрочіпи. розроблено безліч програмних пакетів, що дозволяють реалізовувати нечіткі алгоритми.

В основі нечіткої логіки лежить теорія нечітких множин, де функція приналежності елемента множини не бінарна, а може приймати будь-яке значення в діапазоні від нуля до одиниці. Це дає можливість визначати поняття, нечіткі за самою своєю природою:

- хороший;
- високий;
- слабкий.

Нечітка логіка дозволяє виконувати над такими величинами весь спектр логічних операцій: об'єднання, перетин, заперечення та ін. Нечітка логіка дає можливість будувати бази знань і експертні системи нового покоління, здатні зберігати і обробляти неточну інформацію.

У порівнянні з традиційними методами аналізу й імовірнісним підходом методи нечіткого управління дозволяють швидко проводити аналіз завдання та отримувати результати з високою точністю. Характерними рисами алгоритмів розв'язання задач методами нечіткої логіки є наявність деякого набору тверджень (правил), кожне правило складається з сукупностей подій (умов) і результатів (висновків).

У 1975 році англійський математик Е. Мамдані розробив алгоритм, який був запропонований як метод для управління паровим двигуном. Запропонований ним алгоритм, заснований на нечіткому логічному висновку, дозволив уникнути надмірно великого обсягу обчислень і був гідно оцінений фахівцями. Цей алгоритм нині набув найбільшого практичне застосування в задачах нечіткого моделювання.

Даний алгоритм описує кілька послідовно виконуються етапів (рис. 1). При цьому кожний наступний етап отримує на вхід значення, отримані на попередньому кроці.

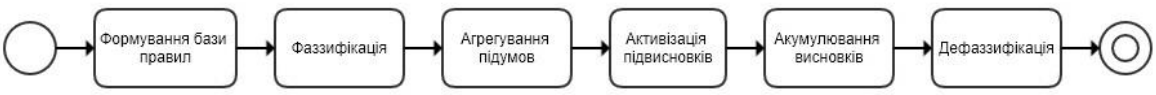


Рис. 1. Схема процесу нечіткого виводу

© НЄВЄЖИНА В.Ю., СТРІЛЕЦЬ В.Є., 2021

Алгоритм нечіткого виведення включає три обов'язкові процедури [4]:

1. Фазифікація – визначення нечітких множин і правил перекладу вихідних даних у нечіткі (включаючи визначення лінгвістичних змінних і функцій приналежності).

2. Рішення завдання в нечітких термінах – створення правил обробки отриманих даних.

3. Дефазифікація – визначення правил перекладу результатів рішення в чіткі величини, які можуть використовуватися в подальших процедурах.

Алгоритм примітний тим, що він працює за принципом «чорного ящика». На вхід надходять кількісні значення, на виході вони ж. На проміжних етапах використовується апарат нечіткої логіки та теорія нечітких множин.

Розглянемо приклад нечіткого виведення [5]. Нехай база нечітких правил прийняття рішень містить визначені експертами залежності шансу працевлаштування молодого спеціаліста від його рівня підготовки (рейтингу) та попиту на спеціалістів такого профілю на ринку праці. Введемо лінгвістичні змінні: рейтинг = (високий, низький); попит = (високий, низький); шанс = (високий, середній, низький). Наведемо декілька із можливих правил:

- R_1 : якщо попит є високим і рейтинг є високим, то шанс є високим;
- R_2 : якщо попит є низьким і рейтинг є високим, то шанс є середнім;
- R_3 : якщо попит є високим і рейтинг є низьким, то шанс є середнім.

Припустимо, що лінгвістичні терми входів описуються такими нечіткими множинами:

- високий попит = {100/0.2; 200/0.4; 300/0.8; 400/1};
- низький попит = {100/1; 200/0.8; 300/0.6; 400/0.4};
- високий рейтинг = {50/0.1; 71/0.8; 88/0.9; 100/1};
- низький рейтинг = {50/1; 71/0.3; 88/0.2; 100/0.1}.

Терми виходу описуються такими множинами:

- високий шанс = {0/0.1; 0.5/0.5; 1/1};
- середній шанс = {0/0.5; 0.5/1; 1/0.5}.

Необхідно визначити шанс працевлаштування при незначному попиті та середньому рейтингу.

Звернемо увагу на те, що вхідні дані не визначають термів незначний попит та середній рейтинг. Вихідну реакцію на ці нечіткі значення необхідно отримати в процесі логічного виведення на основі бази правил.

Нехай на вхід системи надходять нечіткі множини попиту A'_1= {100/0.4; 200/0.3; 300/0.2; 400/0.1} та рейтингу A'_2= {50/0.1; 71/0.8; 88/0.8; 100/0.1}.

Операції визначення мінімуму та максимуму позначимо у вигляді $\wedge$ та $\vee$ відповідно.

Для обчислення виходу виконаємо етапи нечіткого логічного виведення:

1. Обчислення рівнів істинності правил.

a_1 = min[max(0.4^0.2, 0.3^0.4, 0.2^0.8, 0.1^1), max(0.1^0.1, 0.8^0.8, 0.8^0.9, 0.1^1)] =
= min[max(0.2, 0.3, 0.2, 0.1), max(0.1, 0.8, 0.8, 0.1)] = min[0.3, 0.8] = 0.3

a_2 = min[max(0.4^1, 0.3^0.8, 0.2^0.6, 0.1^0.4), max(0.1^0.1, 0.8^0.8, 0.8^0.9, 0.1^1)] =
= min[max(0.4, 0.3, 0.2, 0.1), max(0.1, 0.8, 0.8, 0.1)] = min[0.4, 0.8] = 0.4

a_3 = min[max(0.4^0.2, 0.3^0.4, 0.2^0.8, 0.1^1), max(0.1^1, 0.8^0.3, 0.8^0.2, 0.1^0.1)] =
= min[max(0.2, 0.3, 0.2, 0.1), max(0.1, 0.3, 0.2, 0.1)] = min[0.3, 0.3] = 0.3

2. Обчислення виходів правил.

B_1 = {0/min(0.3, 0.1), 0.5/min(0.3, 0.5), 1/min(0.3, 1)} = {0/0.1, 0.5/0.3, 1/0.3}

B_2 = {0/min(0.4, 0.5), 0.5/min(0.4, 1), 1/min(0.4, 0.5)} = {0/0.4, 0.5/0.4, 1/0.4}

B_3= {0/min(0.3, 0.5), 0.5/min(0.3, 1), 1/min(0.3, 0.5)} = {0/0.3, 0.5/0.3, 1/0.3}

3. Агрегування виходів.

$B'=B'_1 \vee B'_2 \vee B'_3$ = {0/max(0.1, 0.4, 0.3), 0.5/max(0.3, 0.4, 0.3), 1/max(0.3, 0.4, 0.3)} =
= {0/0.4, 0.5/ 0.4, 1/0.4}

4. Дефазифікація виходу.

$$y = \frac{0*0.4+0.5*0.4+1*0.4}{0.4+0.4+0.4} = 0.50.$$

Отже, для заданих нечітких множин, при незначному попиті на спеціалістів та середньому кваліфікаційному рейтингу шанс працевлаштуватися становить 50 %.

Висновки. Методи нечіткої логіки ефективні в задачах прийняття багатокритеріальних рішень, обробки думок групи експертів. Вони знаходять застосування як в управлінні технічними процесами (в тому числі в побутових пристроях), так і в системах управління бізнес-процесами найвищого рівня, включаючи найбільші банківські консорціуми та промислові корпорації.

До недоліків використання нечіткої логіки слід віднести трудомісткість розробки нечітких систем і їх підвищені вимоги до обчислювальних ресурсів. Крім того, фахівців в області нечіткої логіки не так вже й багато, тому обслуговування таких систем і їх модифікація також пов'язані з додатковими витратами.

ЛІТЕРАТУРА

1. Мендель, А.В. Модели принятия решений: учебное пособие / А.В. Мендель. – М.: Юнити, 2015. – 463 с.
2. Принятие решений в условиях нечеткой информации: учеб. пособие / А.Н. Павлов, Б.В. Соколов; ГУАП. – СПб., 2006 – 72 с.
3. Заде Л. Понятие лингвистической переменной и его применение к принятию приближенных решений / Л. Заде. – М.: Мир, 1976. – 165 с.
4. Зак Ю.А. Принятие решений в условиях нечетких и размытых данных: Fuzzy-технологии / Ю.А. Зак. – М. : Либроком, 2013. – 352 с.
5. Кравець П. Системи прийняття рішень з нечіткою логікою / П. Кравець, Р. Киркало // Вісник Національного університету "Львівська політехніка". – 2009. – № 650 : Комп’ютерні науки та інформаційні технології. – С. 115-123.

НЄВЄЖИНА Вероніка Юріївна – студентка кафедри прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: nevezhina511@gmail.com;ORCID: 0000-0003-3915-8959

Наукові інтереси:

– *машинне навчання*.

СТРІЛЕЦЬ Вікторія Євгенівна – к.т.н.; доцент кафедри прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: striletsvictoria@gmail.com; ORCID: 0000-0002-2475-1496.

Наукові інтереси:

– *машинне навчання;*

– *методи штучного інтелекту*.

УДК 004.056.55 **СЕКЦІЯ 5**

ОСТРЯНСЬКА Є.В., ЄСІНА М.В.

ГЕНЕРАЦІЯ ЗАГАЛЬНОСИСТЕМНИХ ПАРАМЕТРІВ ДЛЯ СХЕМИ ЕЛЕКТРОННОГО ПІДПИСУ RAINBOW

Вступ

Однією з основних проблем сучасної криптографії є створення криптографічних схем, які були б безпечними у постквантовий період, тобто будуть захищені як від квантових, так і від класичних атак. На даний момент у світі зусилля багатьох дослідників зосереджені на відкритому конкурсі NIST PQC [1], основною ідеєю якого є знаходження такого алгоритму шифрування, який би був стійким до квантових атак і тому міг бути прийнятим у якості нового стандарту. Фіналістами третього етапу конкурсу NIST PQC стали три схеми ЕП: CRYSTALS-DILITHIUM, FALCON (алгебраїчні решітки) та Rainbow (багатовимірні перетворення). У даній роботі особлива увага була приділена схемі електронного підпису Rainbow.

Rainbow є узагальненням структури UOV [5]. Теоретична безпека Rainbow базується на тому, що вирішення набору випадкових багатовимірних квадратичних систем є NP-складною проблемою [6]. Rainbow заявляє стійкість EUF-CMA, використовуючи геш-конструкцію з випадковою сіллю та пропонує дуже малі підписи лише в кілька сотень бітів.

Для безпечного використання будь-якої криптографічної схеми повинні бути знайдені набори загальносистемних параметрів, за яких забезпечується стійкість до всіх відомих атак. NIST у рамках конкурсу при проектуванні Rainbow, зацікавлені в наборах загальносистемних параметрів до 256 біт проти класичного та до 128 біт включно проти квантового криптоаналізу. Дані обмеження, на нашу думку, пов'язані зі складністю обчислення параметрів, а також з суттєвим впливом їх збільшення на швидкодію електронного підпису. Проте, доцільною є генерація загальносистемних параметрів для 384 і 512 біт безпеки для використання в системах, що потребують підвищеної безпеки, та за яких буде забезпечена стійкість до всіх відомих та потенційних атак.

Метою даної роботи є первинний аналіз відомих атак на схему електронного підпису Rainbow і встановлення обмежень та розробка алгоритмів генерації загальносистемних параметрів для 384 і 512 біт безпеки.

1. Відомі атаки на Rainbow

1.1 Прямі атаки

Найбільш прямолінійною атакою на багатовимірну схему Rainbow, є пряма алгебраїчна атака, в якій загальновідоме рівняння $\mathrm{P}(\mathbf{z})=\mathbf{h}$ розглядається як проблема MQ. Оскільки Rainbow – це невизначена система з $n \approx 1.5 \cdot m$ рівнянь, найефективнішим способом вирішення цієї системи є фіксація $n-m$ змінних для створення детермінованої системи. Можна очікувати, що отримана детермінована система має рівно одне рішення. Складність вирішення такої системи з m квадратних рівнянь у m змінних можна оцінити за допомогою рівності (1)

$$\mathrm{Complexity}_{\mathrm{direct;classical}} = \min_k \left(q^k \cdot 3 \cdot \binom{m-k+d_{reg}}{d_{reg}}^2 \cdot \binom{m-k}{2} \right), \qquad (1)$$

де d_{reg} – це так звана ступінь регулярності системи.

1.2 MinRank атака

Під час атаки MinRank криптоаналітик намагається знайти лінійну комбінацію загальновідомих поліномів мінімального рангу. У випадку з Rainbow така лінійна комбінація рангу υ_2 відповідає лінійній комбінації центральних поліномів першого рівня. Таким чином, знаходячи o_1 цих лінійних комбінацій низького рангу, можна ідентифікувати центральні поліноми першого рівня та відновити еквівалентний секретний ключ Rainbow.

© ОСТРЯНСЬКА Є.В., ЄСІНА М.В., 2021

На сьогодні найбільш ефективний метод вирішення проблеми MinRank був запропонований у [4], в результаті якого отримуємо складність (2)

$$\text{Complexity}_{\text{MinRank}} = 3\cdot\left(\left((o_2+1)\cdot\binom{n'}{r}\right)^2\cdot(r+1)\cdot(o_2+1)\right). \quad (2)$$

1.3 Атака HighRank

Метою атаки HighRank [5] є виявлення (у лінійному представленні) змінних, що з'являються найменшу кількість разів у центральних поліномах (вони відповідають Oil-змінним останнього рівня Rainbow, тобто змінним x_i з $i \in O_u$). Складність цієї атаки можна оцінити за допомогою (3)

$$\text{Complexity}_{\text{HighRank; classical}} = q^{o_u}\cdot\frac{n^3}{6}. \quad (3)$$

1.4 UOV атака

Оскільки Rainbow можна розглядати як продовження добре відомої схеми підпису Oil and Vinegar [2], її можна атакувати, використовуючи всі відомі атаки UOV [6]. Можна розглядати Rainbow як екземпляр UOV з $\upsilon = \upsilon_1 + o_1$ та $o = o_2$. Метою даної атаки є пошук попереднього відображення так званого Oil підпростору O афінного перетворення T, де $\mathrm{O} = \left\{\mathbf{x} \in \mathrm{F}^n : x_1 = \cdots = x_\upsilon = 0\right\}$. Знаходження цього простору дозволяє відокремити Oil від змінних Vinegar та відновити закритий ключ. Складність цієї атаки можна оцінити за допомогою (4) множень у полі

$$\text{Complexity}_{\text{UOV-Attack; classical}} = q^{n-2o_2-1}\cdot o_2^4. \quad (4)$$

1.5 RBS атака

Атака RBS [7] спрямована на пошук лінійних відображень S і T, що перетворюють загальновідомі поліноми в поліноми форми Rainbow. Для цього криптоаналітик повинен вирішити кілька нелінійних мультиваріативних систем. Складність цього кроку визначається складністю вирішення першої (і найбільшої) з цих систем, яка складається з $n+m-1$ квадратних рівнянь з n змінними і можна оцінити як (5), де $\mathrm{M}_{\alpha,\beta}$ позначає кількість одночленів (α,β)

$$Compl_{RBS} = \min_{\alpha,\beta} 3\cdot \mathrm{M}_{\alpha,\beta}(t,s)^2\cdot(n_x+1)\cdot(n_y+1). \quad (5)$$

2. Генерація параметрів для 384, 512 біт стійкості

В цьому розділі наведено вибір (генерацію) параметрів для 384, 512 біт стійкості для схеми ЕП Rainbow над полем GF(256). Під час вибору параметрів керувалися такими умовами:

- Кількість рівнянь, що нам необхідна, залежить від складності прямої атаки та атаки на геш-функцію;
- Кількість змінних залежить від складності атак RBS, UOV та HighRank.

Тож, якщо підсумувати вище сказане, то знайти параметри υ_1, o_1, o_2 при $q = 256$, тобто $GF(q) = GF(256)$ можливо з умов (4)-(14). На основі цього було розроблене програмне забезпечення, з використанням якого були згенеровані параметри υ_1, o_1 та o_2 для схеми ЕП Rainbow для 384 та 512 біт безпеки, що наведені в таблиці 1.

Табл. 1 Основні загальносистемні параметри Rainbow для 384, 512 біт безпеки

Безпека	υ_1	o_1	o_2	$GF(q)$
384	192	48	136	$GF(256)$
512	272	120	128	$GF(256)$

При таких параметрах отримуємо наступні розміри ключів, гешу та підписів для трьох версій Rainbow: класичної (Classic), циклічної (CZ-Rainbow), та стисненої (Compressed), що наведені в таблицях 2-4 відповідно.

Табл. 2 Розміри ключів та підписів Classic Rainbow

Безпека	Набір параметрів $(\mathrm{F}, \upsilon_1, o_1, o_2)$	Розмір відкритого ключа (у байтах)	Розмір закритого ключа (у байтах)	Розмір гешу (у байтах)	Розмір підпису (у байтах)
384	$(GF(256), 192, 48, 136)$	13041184	9752288	64	392
512	$(GF(256), 272, 120, 128)$	33594080	24752480	64	536

Табл. 3 Розміри ключів та підписів CZ-Rainbow

Безпека	Набір параметрів $(\mathrm{F}, \upsilon_1, o_1, o_2)$	Розмір відкритого ключа (у байтах)	Розмір закритого ключа (у байтах)	Розмір гешу (у байтах)	Розмір підпису (у байтах)
384	$(GF(256), 192, 48, 136)$	3337344	9752288	64	392
512	$(GF(256), 272, 120, 128)$	8939840	24752480	64	536

Табл. 4 Розміри ключів та підписів Compressed Rainbow

Безпека	Набір параметрів $(\mathrm{F}, \upsilon_1, o_1, o_2)$	Розмір відкритого ключа (у байтах)	Розмір закритого ключа (у байтах)	Розмір гешу (у байтах)	Розмір підпису (у байтах)
384	$(GF(256), 192, 48, 136)$	3337344	64	64	392
512	$(GF(256), 272, 120, 128)$	8939840	64	64	536

Швидкодія при заданих параметрах для трьох версій Rainbow, що представлена в тактах процесора, наведена в таблицях 5-7 відповідно.

Табл. 5 Швидкодія Classic Rainbow

Набір параметрів	Генерація ключів	Генерація підпису	Верифікація підпису
384	4658727146	16484356	2645458
512	16999329532	43986312	8165628

Табл. 6 Швидкодія CZ-Rainbow

Набір параметрів	Генерація ключів	Генерація підпису	Верифікація підпису
384	4658375168	16799206	2927814
512	16900406374	52017328	10617618

Табл. 7 Швидкодія Compressed Rainbow

Набір параметрів	Генерація ключів	Генерація підпису	Верифікація підпису
384	4631048528	16288184	2398794
512	16694833556	44923458	7611730

Висновки

1. Схема електронного підпису Rainbow значно відрізняється від інших кандидатів конкурсу NIST, оскільки заснована на багатовимірних перетвореннях. Вона є узагальненням структури UOV, що забезпечує ефективну параметризацію за рахунок додаткової алгебраїчної структури. Теоретична безпека Rainbow базується на тому, що вирішення набору випадкових

багатовимірних квадратичних систем є NP-складною проблемою. Rainbow заявляє стійкість EUF-CMA, використовуючи геш-конструкцію з випадковою сіллю.

2. Перевагами Rainbow є те, що процес вироблення підписів складається з простих операцій лінійної алгебри, таких як множення матричних векторів та вирішення лінійних систем над малими скінченними полями. Також Rainbow пропонує дуже малі підписи лише в кілька сотень бітів. Але головним недоліком Rainbow є великий розмір відкритих ключів. Результати досліджень наведено в таблицях 2-4.

3. Як видно з таблиць 5-7, процес верифікації підпису CZ-Rainbow є повільнішим, ніж у стандартній схемі Rainbow. Однак необхідно зазначити, що це спричинене використанням криптографічно захищеного PRNG на базі AES, що постачається OpenSSL (що є таким самим, що і у NIST), для створення «фіксованих» частин відкритого ключа. Використовуючи швидший потоковий шифр або навіть генеруючи відкритий ключ, використовуючи реєстр лінійного зворотного зсуву (LFSR), цього уповільнення можна уникнути майже повністю.

4. Було розглянуто низку атак на схему ЕП Rainbow, а саме пряму атаку, атаку MinRank та HighRank, атаку UOV та атаку «Rainbow Band Separation» (RBS). Хоча пряма атака є атакою підробки підпису, яка повинна виконуватися для кожного повідомлення окремо, атаки MinRank, HighRank, UOV та RBS є ключовими атаками відновлення. Після відновлення секретного ключа Rainbow за допомогою однієї з цих атак криптоаналітик може виробляти підписи так само, як законний користувач.

5. В даній роботі були запропоновані набори загальносистемних параметрів схеми ЕП Rainbow для 384 та 512 біт безпеки: (GF(256),192,48,136) та (GF(256),272,120,128) відповідно.

ЛІТЕРАТУРА

1. PQC Standardization Process: Third Round Candidate Announcement. July 22, 2020. [Electronic resource]. – URL: https://csrc.nist.gov/News/2020/pqc-third-round-candidate-announcement

2. Gentry C., Peikert Ch., and Vaikuntanathan V. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, 40th ACM STOC, pages 197–206. *ACM Press*, May 2008.

3. Stehlé D. Steinfeld R. Making NTRU as secure as worst-case problems over ideal lattices. In Kenneth G. Paterson, editor, *EUROCRYPT 2011*, volume 6632 of LNCS, pages 27–47, Tallinn, Estonia, May 15–19, 2011. Springer, Heidelberg, Germany.

4. Prest Th. Gaussian Sampling in Lattice-Based Cryptography. *Theses, École Normale Supérieure*, December 2015.

5. Kipnis A., Patarin J., Goubin L. Unbalanced Oil and Vinegar schemes. *EUROCRYPT 1999*, LNCS vol. 1592, pp. 206 - 222. Springer, 1999.

6. Ding J., and other. Rainbow Signature. 2020. P. 16-22. [Electronic resource]. – URL: https://www.pqcrainbow.org/

7. Ding J., Schmidt D.: Rainbow, a new multivariable polynomial signature scheme. ACNS 2005, *LNCS* vol. 3531, pp. 164 - 175. Springer, 2005.

ОСТРЯНСЬКА Єлизавета Вадимівна – аналітик з систем захисту інформації; АТ «Інститут інформаційних технологій», вул. Бакуліна, 12, Харків, Україна, 61166; e-mail: antelizza@gmail.com; ORCID: 0000-0003-1412-8470.

Наукові інтереси:

– *постквантова криптографія, багатовимірна криптографія, криптографія на алгебраїчних решітках.*

ЄСІНА Марина Віталіївна – к.т.н, доцент кафедри безпеки інформаційних систем і технологій Харківського національного університету імені В.Н. Каразіна, майдан Свободи, 6, Харків, Україна, 61000; e-mail: m.v.yesina@karazin.ua; ORCID: 0000-0002-1252-7606.

Наукові інтереси:

– *постквантова криптографія, багатовимірна криптографія, криптографія на алгебраїчних решітках.*

УДК 1331 **СЕКЦІЯ 2**

ПАВЛОВ А.М, ХИЖНЯК О.В.

ОПТИМІЗАЦІЯ СТРУКТУРИ СКЛАДНОЇ СИСТЕМИ ЗА КРИТЕРІЄМ НАДІЙНОСТІ

Вступ

В даний час системи управління ставали складніше і складніше. Відповідно до них виявляються більш високі вимоги по надійності. При цьому завжди виникає суперечність між забезпеченням максимальної надійності і вартістю системи. Дозвіл протиріччя можливо при вирішенні оптимізаційної задачі при обмеженні ресурсів і техніко-економічних параметрів елементів підсистем.

Математична модель цієї задачі має наступний вигляд: визначити варіант u0, який доставляє максимум цільової функції

$$P(v) = \prod_{j=1}^{n} P_j(u_{j(lj)}) \quad (1)$$

при наявності обмежень

$$g_p(v) = \sum_{j=1}^{n} g_p\left(u_{j(lj)}\right) \le g_p^* \qquad (p = 1, \dots, q) \quad (2)$$

$$g_p(v) = \sum_{j=1}^{n} g_p\left(u_{j(lj)}\right) \ge g_p^* \quad (p = q+1, \dots, Q),\ \text{v} \in \text{V},\ u_{j(lj)} \in U_j,\ (\text{j}=1,\dots,\text{n}) \quad (3)$$

де U_j сукупність елементів різних типів, які можуть бути використані в j-й підсистемі, кількість елементів у множині U_j равно z_j;

$V = \prod_{j=1}^{n} U_j$ – варіант системи v ϵ V;

$P_j\left(u_{j(lj)}\right)$ – надійність (ймовірність безвідмовної роботи на заданому інтервалі часу) елемента j-й підсистеми l_j-го типа;

$g_p\left(u_{j(lj)}\right)$ – значення p-го обмежуючого фактора для елемента l_j-го типа j-й підсистеми;

$g_p(v)$ – кількість p-го обжежуючого фактора, витраченого на всю систему;

g_p^* - максимально можливу кількість p-го обмежує фактора для всієї системи в цілому.

Задача (1) - (3) еквівалентна наступній завданню:

найти максимум

$$f(v) = \sum_{j=1}^{n} f_j(u_{j(lj)}) \quad (4)$$

при наявності обмежень (2), (3), де $f_j\left(u_{j(lj)}\right) = lgP_j\left(u_{j(lj)}\right)$.

Таким чином, методика оптимізації структури складної системи за критерієм надійності буде полягати в наступному:

1. Побудувати систему, що складається з послідовно з'єднаних підсистем.
2. Визначити можливі елементи, з яких може бути побудована кожна підсистема.
3. Провести відсів елементів за типами, не застосовуючи резервування, визначаючи допуски для техніко-економічних параметрів елементів для кожної підсистеми.
4. Визначити мінімальну кратність резервування для кожного з решти типів елементів.
5. Визначити значення цільової функції за критерієм надійності за умови резервування елементів в підсистемах.
6. Знайти оптимальне рішення при максимізації цільової функції.

Застосування методики та математичної моделі покажемо на прикладі.

Табл.1 Техніко-економічні параметри

№ підсистеми	Тип елемента в підсистемі	Вартість	Вага	Габарити	Надійність
1	1	5	10	4	0,6
	2	17	5	2	0,9
	3	10	5	5	0,85
	4	2	40	20	0,55
	5	5	6	5	0,8
	6	25	2	2	0,95

© ПАВЛОВ А.М, ХИЖНЯК О.В., 2021

	7	7	35	10	0,65
2	1	9	15	15	0,7
	2	30	10	10	0,95
	3	8	10	40	0,75
	4	10	15	10	0,85
3	1	12	12	20	0,75
	2	10	15	15	0,7
	3	25	10	10	0,9
	4	11	15	39	0,7
4	1	5	5	13	0,7
	2	4	10	35	0,8
	3	20	2	5	0,95
	4	5	6	10	0,8
5	1	10	10	20	0,6
	2	12	10	10	0,85
	3	6	40	30	0,55
	4	10	11	10	0,8
	5	27	5	15	0,95
	6	7	20	40	0,6

Необхідно вибрати варіант системи, що володіє максимальною надійністю, застосовуючи резервування елементів підсистем, причому в якості основного і резервного елементів в j-й підсистемі можуть бути обрані елементи одного з z_j типів і при цьому не повинні порушуватися обмеження на систему в цілому: за вартістю $g_1(v) \le g_1^*$=50, по вазі $g_2(v) \le g_2^*$=65, по габаритам $g_3(v) \le g_3^*$=65.

1. Відсів елементів за типами, не застосовуючи резервування.

Рис. 1. Логічна надійністна схема

1.1 Відсів елементів за вартістю.

Для цього в таблиці 1 розглянемо стовпець значень вартостей елементів різних типів всіх підсистем, і, упорядкувавши її в рядках по зростанню отримаємо таблицю 2.

Табл. 2

1 підсистема	2(4)	5(5)	5(1)	7(7)	10(3)	17(2)	25(6)
2 підсистема	8(3)	9(1)	10(4)	30(2)			
3 підсистема	10(2)	11(4)	12(1)	25(3)			
4 підсистема	4(2)	5(1)	5(4)	20(3)			
5 підсистема	6(3)	7(6)	10(1)	10(4)	12(2)	27(5)	

Обчислюємо $\Delta g_1^1 = g_1^* - g_1(\frac{v^1}{U_1})$ = 50-28=22 - «допуск» для елементів першої підсистеми.

Аналогічно обчислюються допуски для інших підсистем (рядків в таблиці 2).

1.2 Відсів елементів по вазі

Аналогічно п.1.1 обчислюємо допуски для елементів кожної підсистеми.

1.3. Відсів елементів за габаритами

Аналогічно п.1.2 обчислюємо допуски для елементів кожної підсистеми.

1.4 Повторення ітерації з урахуванням відсіяних елементів

Потім знову повертаємося до таблиці 2, викресливши елементи.

2. Визначення максимальної кратності резервування

Для кожного з решти типів елементів в кожній підсистемі визначаємо максимальну кратність резервування

$$\lambda^*_{j(lj)} = \min_{p=1\ldots a}\left[\frac{\Delta g^j_p}{g_p(u_{j(lj)})} - 1\right] \quad (5)$$

де [a] – ціла частина числа.

Так, для першого елемента першої підсистеми максимальна кратність резервування визначається:

$$\lambda^*_{11} = \left\{min\left\{\left[\frac{\Delta g^1_1}{5}, \frac{\Delta g^1_2}{10}, \frac{\Delta g^1_3}{4}\right]\right\} - 1\right\} = 1, \quad (6)$$

де 5, 10, 4 - відповідно значення вартості, ваги і габаритів елемента 1 в першій підсистемі, а Δg^1_1, Δg^1_2, Δg^1_3 – обчислені вище допуски за вартістю, вагою та габаритами. В результаті отримаємо таблицю 6.

Табл. 6

№ підсистеми	1	2	3	4	5
Тип элемента	1, 3, 5	1, 4	1, 2	1, 4	1, 2, 4
Максимальна кратність резервування	1, 0, 2	0, 1	0, 0	1, 1	0, 0, 1

3. Максимізація цільової функції за критерієм надійності за умови резервування елементів в підсистемах

Для нашого прикладу

$$f(v) = \sum_{j=1}^{n}\left(c_0 lg P_j + c_1\right) \quad (7)$$

де $P_j = 1 - \left(1 - P_{j(lj)}\right)^{\lambda_{j(lj)}+1}$ – надійність j-й підсистеми;

$P_{j(lj)}$ – надійність элемента l_j-го типа j-й підсистеми;

c_0, c_1 – константи (c_0>0), які вибирають з зручності обчислень.

Значення логарифмів досить брати з точністю 0,01.

4. Пошук оптимального рішення

4.1. Перше обмеження на цільову функцію

Обчислюємо перше обмеження на цільову функцію

$$f_1^* = \frac{f_{max}+f_{min}}{2} \cong 56 \quad (8)$$

4.2. Друге обмеження на цільову функцію

Обчислюємо друге обмеження на цільову функцію

$$f_2^* = \frac{f_{max}+f_1^*}{2} = 74 \quad (9)$$

4.3. Третє обмеження на цільову функцію

Обчислюємо третє обмеження на цільову функцію

$$f_3^* = \frac{f_{max}+f_2^*}{2} = 83 \quad (10)$$

.

4.4. Четверте обмеження на цільову функцію

Обчислюємо четверте обмеження на цільову функцію

$$f_4^* = \frac{f_{max}+f_3^*}{2} = 87 \quad (11)$$

4.5. Розширення обмеження на цільову функцію

Обчислюємо п'ятий обмеження на цільову функцію

$$f_5^* = \frac{f_3^* + f_4^*}{2} = 85 \quad (12)$$

Варіант 1 системи 1.5.5, 2.4.4, 3.2.2, 4.4.4, 5.3.4 (20 + 15 + 7 + 20 + 12 = 74)
Варіант 2 системи 1.3.3, 2.4.4, 3.2.2, 4.4.4, 5.3.4 (15 + 15 + 7 + 20 + 12 = 69)
Варіант 1 є допустимим і має більшу надійність.

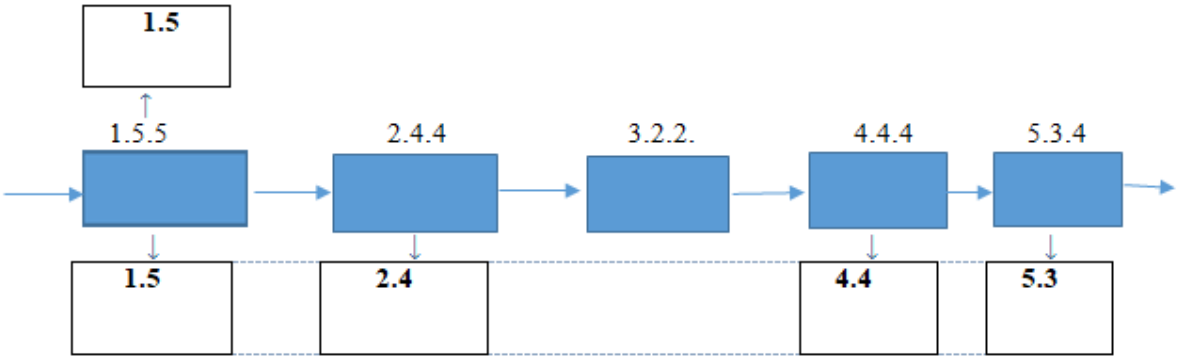


Рис. 2. Результат обчислень

Висновки

Отриманий варіант найкращий по надійності і є допустимим і оптимальним.

ЛІТЕРАТУРА

1. Felix Mikhailovich, Anatoly Georgievich. Methodical recommendations for project reliability assessment.
2. Anatoly Georgievich. Methodical instructions on registration of documentation on software products (products) at performance of diploma works, term papers and practical tasks on disciplines of chair of theoretical and applied system engineering.

ПАВЛОВ Анатолій Миколайович – старший викладач факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: anatolijpavlov1331@gmail.com,; ORCID: 1331-1331-1331-1331
Наукові інтереси:
- *Комп'ютерні мережі, моделювання комп'ютерних мереж.*

ХИЖНЯК Олег Вікторович – студент групи КУ-41 факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: oleghiznyak987@gmail.com; ORCID: 0000-0002-8343-0246.
Наукові інтереси:
- *Комп'ютерні мережі, моделювання комп'ютерних мереж.*

УДК 621.394 **СЕКЦІЯ 7**

ПАНЧЕНКО Н.В.

МАТЕМАТИЧНА МОДЕЛЬ СИСТЕМИ ВЕРИФІКАЦІЇ ВІДБИТКІВ ПАЛЬЦІВ НА ОСНОВІ ЦИЛІНДР-КОДІВ

У сучасному світі стрімкого розвитку та поширення технологій одночасно зростає актуальність питання надійного забезпечення цілісності, доступності та конфіденційності інформації. Необхідним є існування автентифікаці особистості при процесі контролю доступу до заданого ресурсу або системи.

Математична модель системи верифікації відбитків пальців на основі циліндр-кодів базується на біометрії - системі, що використовує фізичні або поведінкові характеристики людини для встановлення особи. Біометричні повинні мати такі властивості: унікальність, продуктивність, універсальність, індивідуальність, стабільність, стандартність та вимірюваність. У даній математичній моделі основою є біометричної характеристики неповторності папілярних ліній узорів на пальцях. Відбитки складається з ліній та мінуцій – ділянок, де окремі лінії зходяться, розходяться або обриваються. Далі воно трансформується у цифровий код для подальшої верифікації з шаблоном, що був введений у систему раніше. Під час процедури розпізнавання відбитків пальців можуть виникнути проблеми: зсув, поворот, масштабування, фрагментація окремих елементів, хибні мінуції або їх зникнення.

Суть даної метрики у тому, що будується локальна структура для кожної з мінуцій m - циліндр, що є стійким по відношенню до таких трансформацій, як зсув і поворот. Основна відмінність від інших локальних уявлень у тому, що вплив сусідніх мінуцій на основну не враховується. Попереднє вірівнювання зображення не потрібне. Використовуються три параметри шаблонів для кожної з мінуцій: координати x, y та кут приходу в мінуцію θ.

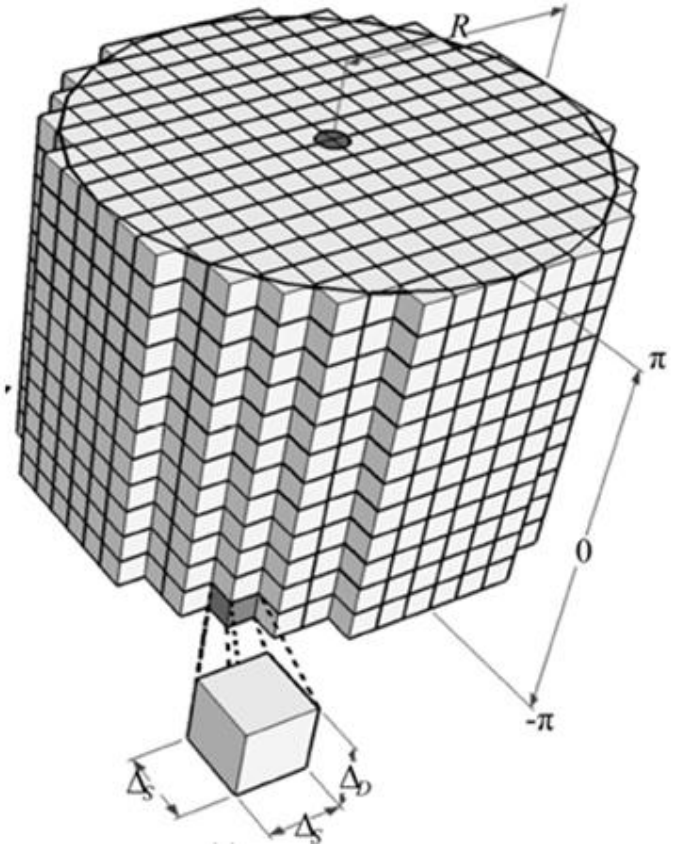


Рис. 1 – Циліндр – локальна структура

© ПАНЧЕНКО Н.В., 2021

Циліндр є вписаним у паралелепіпед, довжина ребер якого співпадає з діаметром основи циліндра. Висота змінюється, а координати центру співпадають з глобальними. Висота - 2π, $N_s \cdot \Delta_S$ - довжини основ паралелепіпеда. Радіус основи:

$$R = \frac{N_S \cdot \Delta_S}{2} \tag{1}$$

Циліндр розбивається на паралелепіпедні осередки розміру $\Delta_S \cdot \Delta_S \cdot \Delta_D$, що ідентифікується системою координат (i, j, k), де напрям основи i вирівнюється згідно з кутом мінуції θ_m. При обчисленнях задіюється лише глобальна система координат (x, y). N_D - частини циліндра вздовж осі k.

При кодуванні кожній ділянці мінуції, що потрапляє у радіус, зіставляється своє числове значення. Кожен шар відповідає куту $(\theta_m - \theta_{m_t}) \in d\varphi_k$, де d - відстань між точками:

$$d\varphi_k = -\pi + (k - \frac{1}{2}) \cdot \Delta_D \tag{2}$$

Координати центру ділянки зіствляються числовими значеннями координатів та кутів напряму найближчих мінуцій – множина сусідства. Евклідова відстань між точками мінуції, що впливає, та центром ділянки, обчислюється за допомогою гауссової функції:

$$G_S(t) = \frac{1}{\sigma_S \sqrt{2\pi}} \exp[-\frac{t^2}{2 \cdot \sigma_s^2}] \tag{3}$$

Вона характеризується просторовою контрибуцією

$$C_m^D(m_t, p_{i,j}^m) = G_s[d_s(m_i, p_{i,j}^m)] \tag{4}$$

та дирекційною контрібуцією:

$$C_m^D(m_t, d\varphi_k) = G_D[d\varphi(d\phi(d\varphi_k, d_\theta(m, m_t))], \tag{5}$$

де $d\phi(\theta_1, \theta_2)$ - різницевий кут.

Числове значення кожної ділянки може бути релевантним або не релевантим, якщо вона не має числового еквіваленту. Для релевантих ділянок розраховується числове значеня, враховуючи вплив сусідніх мінуцій. Далі циліндр, що являє собою числові значення, зіставляється із кожною мінуцією шаблону. Циліндр не є релевантним, якщо число релевантних ділянок у ньому менше мінімума, або число найближчих мінуцій у шаблоні менше мінімального.

Ступінь схожості циліндрів перетворюються у одномірні вектори числових характеристик, для яких розраховується ступінь кореляційної близькості. Масив індексів (i, j, k) перетворюється в одномірний масив q:

$$q = (k-1) \cdot (N_S)^2 + (j-1) \cdot N_S + i \tag{6}$$

Для зіставлення нерелевантні ділянки заповнюються нулями, збільшується кількість елементів, що не несуть в собі інформації. Масиви чисел для порівняння - C_1 та C_2. Описує схожість величина взаємної кореляції $\gamma(C_1, C_2)$:

$$1 - \frac{\|c_1 - c_2\|}{\|c_1\| + \|c_2\|}, \tag{7}$$

якщо циліндри тотожні, 0 - якщо не тотожні.

Наявність множини нульових елементів у складі циліндр-кодів, що винеачається алгоритмом кодування, суттєво знижує об'єктивність розрахунків локального збігу циліндрів мінуцій.

Беручи до уваги усі раніше перераховані переваги, можна зробити висновок, що дана метрика є дійсно актуальною та корисною. Недоліки: великий обсяг обчислень та громіздкість опису та наявність елементів, що не несуть інформації. Але алгоритм є стійким до спотворень відбитка пальця, зсуву та надає змогу отримати вектори числових кодів певного розміру, що дорівнюють кількості осередків у паралелепіпеді, у який вписано циліндр. Шляхом подальшого розвитку методу можна досягнути більш удосконалених результатів, що говорить про його перспективність.

ЛІТЕРАТУРА

1. Jain, A. K.; Ross, Arun & Prabhakar, Salil (January 2004), "«An introduction to biometric recognition»", IEEE Transactions on Circuits and Systems for Video Technology.
2. ISO/IEC 19794-2:2005, Information Technology—Biometric Data Interchange Formats – Part 2: Finger Minutiae Data, 2005.
3. R. Cappelli, M. Ferrara, and D. Maltoni "Minutia Cylinder-Code: A New Representation and Matching Technique for Fingerprint Recognition". IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 32, NO. 12, DECEMBER 2010. pp. 2128-2141.
4. W. U. Baig, U. Munir, W. Ellahi, A. Ejaz, K. Sardar "Minutia Texture Cylinder Codes for fingerprint matching".
5. James W. Yang "A Dab Fingerprint Verification System Based on Classical Methods".
6. Дж. Фэн и А. Цай, «Индексирование отпечатков пальцев с использованием инвариантов хребта», Proc.Междунар. Конф. Распознавание образов, 2006.

ПАНЧЕНКО Наталя Вячеславівна – студент; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків, Україна, 61022; e-mail: xa122084021@student.karazin.ua; nataliapanchenko@ukr.net; ORCID: 0000-0003-0481-0281.

Наукові інтереси:

– *математичні моделі систем верифікації відбитків пальців.*

УДК 621.59:04 **СЕКЦІЯ 1**

ПЕТУХОВ І.І., ЛИСИЦЯ О.Ю.

ОСОБЛИВОСТІ CFD-МОДЕЛЮВАННЯ РОБОЧОГО ПРОЦЕСУ СТРУМИННОГО ОХОЛОДЖУВАЧА РІДИНИ

Вступ

З огляду на низькі температури кріогенних рідин їх застосування пов'язане з необхідністю охолодження. Особливо актуальна ця задача при великих об'ємах кріогенних баків і обмежених по габаритним показникам можливостях вдосконалення теплоізоляції. Такі умови характерні для ракетно-космічної техніки. Причому завдання охолодження може ускладнюватися через перспективність використання кріогенних компонентів палива в переохолодженому або шугоподібному стані [1], коли тиск насичення нижче атмосферного.

Для охолодження широко використовується часткове випаровування рідини в адіабатних умовах. Найбільш просто процес реалізується за рахунок вакуумування парового простору ємності. Однак при великих її об'ємах створення значного розрідження проблематичне з міркувань стійкості і міцності конструкції. Використання поверхневих теплообмінників з розділеними потоками пари і охолоджувальної рідини усуває цей недолік, однак з огляду на малі температурні напори маса конструкції надзвичайно велика.

Недоліки обох способів усуваються при використанні струминного охолоджувача рідини (СОР) [2]. Висока швидкість потоку і розвинена поверхня контакту фаз забезпечують малі габарити пристрою і швидкість охолодження 400...600 К/с. Крім того, тут наявна робота відведеного пару корисно використовується для підвищення повного тиску рідини при їх спільному розширенні. Однак реалізований рівень математичного опису робочого процесу СОР не дозволяє досить надійно проектувати пристрій і визначити його характеристики для конкретних умов експлуатації.

Основна частина

У СОР охолодження рідини забезпечується за рахунок часткового її випаровування при розширенні потоку, що адіабатно скипає в соплі 1, подальшого поділу високошвидкісного парорідинного потоку в сепараторі 2 і відновлення статичного тиску охолодженої рідини в дифузорі 3 (Рис. 1). Точка "S" відповідає початку пароутворення в потоці рідини, що закипає. Далі паровміст зростає і рідина перебуває в соплі у вигляді дрібнодисперсних крапель. Парорідинний потік на зрізі сопла має нижчу, ніж на вході, температуру і кінетичну енергію $v_2^2/2$.

Розвинена міжфазна поверхня обумовлює високу інтенсивність тепло-масообмінних процесів, силової взаємодії між парою і рідиною. Це дозволяє значно зменшити габарити і масу пристрою, дає можливість підвищити повний тиск охолоджувальної рідини за рахунок роботи, яка здійснюється паром при спільному розширенні з рідиною. В ідеалізованому процесі повний тиск охолодженої рідини на виході з дифузора СОР p5 завжди вище тиску на вході (Рис. 1б).

У поверхневому сепараторі 2 парорідинний потік розділяється на фази. Утворений в соплі пар, частка якого становить 4...20% в залежності від величини охолодження і роду рідини, відводиться в вихлопний патрубок (переріз 6). Охолоджена рідина відновлює свій тиск в дифузорі 3. Тиск охолодженої рідини на виході СОР (переріз 5) завжди вище тиску насичення. При зниженні тиску на зрізі сопла до значення нижче потрійної точки можливо отримати потік шугоподібного компонента.

У реальному процесі розширення рідина в горлі сопла закипає при тиску нижче насичення, також має місце зміна структури парорідинного потоку, термічна і швидкісна нерівноважність фаз. Процес сепарації супроводжується тепло-масообміном, винесенням дрібних крапель паром, розбризкуванням рідини і потраплянням пари в відсепарований шар при осадженні крапель.

© ПЕТУХОВ І.І., ЛИСИЦЯ О.Ю., 2021

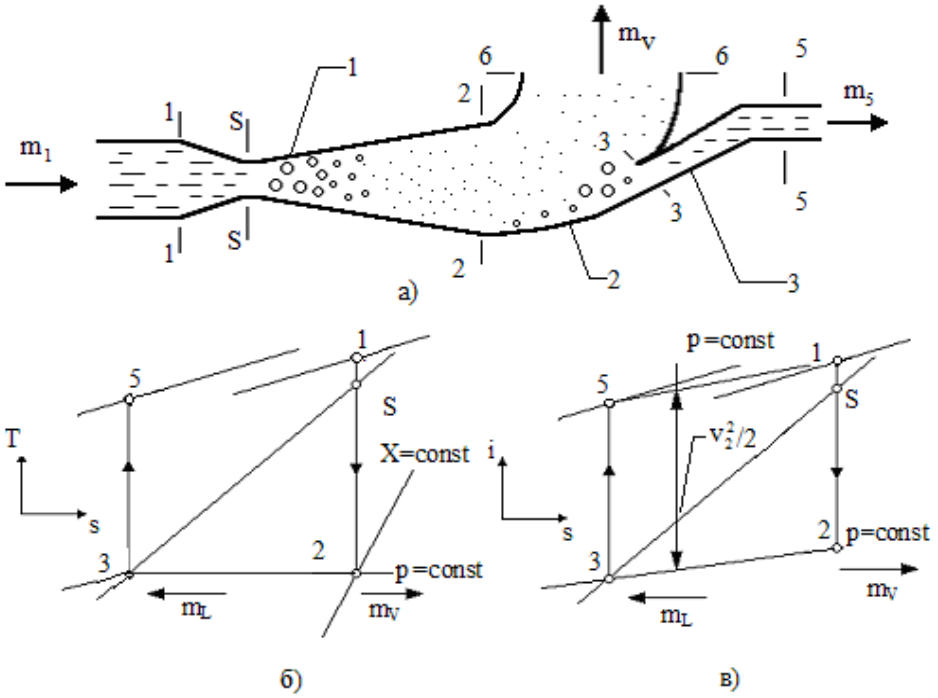


Рис. 1 Ідеалізований робочий процес СОР. а) схема СОР: 1 - двофазне сопло; 2 - поверхневий сепаратор; 3 - дифузор; б) ідеалізований робочий процес СОР в T, s - координатах; в) ідеалізований робочий процес СОР в i, s – координатах

Поблизу криволінійної поверхні формується відсепарований бульбашковий потік. По мірі віддалення від поверхні частка рідини зменшується практично до нуля. Далі потік зі змінним по висоті об'ємним вмістом вологи направляється в дифузор, де відбувається відновлення тиску, що супроводжується стрибком ущільнення – конденсацією. Зі збільшенням висоти перерізу входу в дифузор зростає витрата охолодженої рідини, але знижується досяжний тиск за дифузором.

Сучасні CFD-пакети [3] принципово дозволяють сформувати модель цілісного робочого процесу СОР на основі підходу Ейлера. Однак в цьому випадку навряд чи вдасться достовірно описати процес формування відсепарованої плівки. Представлений в [3] модуль використовує припущення однорідної по товщині тонкої плівки і в такому вигляді не може застосовуватися в моделі СОР. Тому більш прийнятним виглядає використання підходу [2] поелементного математичного моделювання СОР. При цьому для кожного елемента слід використовувати свою CFD-модель, а результати розрахунку попереднього елемента повинні формуватися як вихідні дані для подальшого. В якості базової досить використовувати двовимірну модель, так як СОР являє собою профільований плоский канал з досить великою шириною.

При розрахунку сопла заданої геометрії на вході задаються тиск і температура рідини. Базова система рівнянь для нерівноважного двофазного (n=2) потоку з міжфазним тепло-масообміном в позначеннях [3] має для q-ї фази наступний вигляд:

рівняння нерозривності

$$\frac{\partial}{\partial t}(\alpha_q \rho_q) + \nabla \cdot (\alpha_q \rho_q \vec{v}_q) = \sum_{p=1}^{n} \dot{m}_{pq} + S_q \,,$$

рівняння збереження імпульсу

$$\alpha_q \rho_q \frac{\partial \vec{v}_q}{\partial t} + \alpha_q \rho_q (\overrightarrow{v_q} \cdot \nabla)\vec{v}_q = -\alpha_q \nabla p + \nabla \overline{\overline{\tau_q}} + \alpha_q \rho_q \vec{g} + \sum_{p=1}^{n} (\vec{R}_{pq} + \dot{m}_{pq}\vec{v}_{pq} - \dot{m}_{pq}\vec{v}_q) + (\vec{F}_q + \vec{F}_{lift,q} + \vec{F}_{wl,q} + \vec{F}_{vm,q} + \vec{F}_{td,q}) \,,$$

рівняння збереження енергії

$$\alpha_q \rho_q \frac{\partial h_q}{\partial t} + \alpha_q \rho_q (\vec{v}_q \cdot \nabla) h_q = \alpha_q \frac{dp_q}{dt} + \bar{\bar{\tau}}_q : \nabla \vec{v}_q - \nabla \vec{q}_q + \sum_n^{p=1} (Q_{pq} + \dot{m}_{pq} h_{pq} - \dot{m}_{pq} h_q)$$

Основними результатами є масова витрата, профіль швидкості пара, розподіл крапель і їх швидкості по висоті зрізу сопла. Ці дані використовуються в якості вихідних для розрахунку сепаратора. При цьому система рівнянь при розрахунку парокрапельного потоку не відрізняється від представленої вище. Для розрахунку параметрів в пристінній області може використовуватися умова інверсії структури потоку або сформований додатково призначений для користувача модуль моделі плівки. Неоднорідний по нормалі потік поблизу криволінійної стінки на виході з сепаратора визначає параметри на вході в дифузор. При цьому задається висота його щілини. Параметри на виході з дифузору можуть розраховуватися на основі моделі [3].

Висновки

Значні масогабаритні переваги СОР, висока швидкість охолодження рідини та можливість використання роботи розширення пари є основними чинниками, котрі визначають перспективні схеми його використання. Найбільш прийнятними виглядають системи охолодження кріогенних компонентів палива та високотемпературної термообробки рідких сумішей. Наявність достовірних математичних моделей робочого процесу СОР є важливим елементом для його впровадження. Для удосконалення цих моделей слід комбінувати CFD-модулі з наявними поелементними методами, особливо в частині процесу в дифузорі.

ЛІТЕРАТУРА

1. Шугообразное горючее. Ракетная техника, серия 1, №32, 33, 1989, с. 16. ..18.
2. Петухов И.И. Научные основы рабочего процесса систем быстрого охлаждения и получения шугообразного состояния жидкости. - Наукові праці: Науково-методичний журнал. Т. 111. Вип.. 98. Техногенна безпека. – Миколаїв: Вид-во ЧДУ ім. П. Могили, 2009. – С. 83-89.
3. Help Ansys Fluent. Release 2020 R1.

ПЕТУХОВ Ілля Іванович – к. т. н., доцент, доцент кафедри аерокосмічної теплотехніки; Національний аерокосмічний університет ім. М.Є. Жуковського "ХАІ", вул. Чкалова 17, м. Харків, Україна, 61070. E-mail: ilya2950@gmail.com. Номер ORCID: http:// orcid.org/ 0000-0002-0645-7912

Наукові інтереси:

- *термогідравлічні процеси у багатофазних потоках.*

ЛИСИЦЯ Олексій Юрійович – к. т. н., доцент; доцент кафедри аерокосмічної теплотехніки; Національний аерокосмічний університет ім. М.Є. Жуковського "ХАІ", вул. Чкалова 17, м. Харків, Україна, 61070. E-mail: a.lisitca@khai.edu. Номер ORCID: https://orcid.org/0000-0002-5679-8459

Наукові інтереси:

- *CFD-моделювання теплогідравлічних процесів в двигунах та енергетичних установках.*

УДК 004.9 **СЕКЦІЯ 7**

ПОМОГАЄВ К.О, АРТЮХ О.А.

МОДЕЛЬ 3D-ВІЗУАЛІЗАЦІЇ НАВЧАЛЬНИХ ПРИМІЩЕНЬ КАФЕДРИ

3D-моделювання та його застосування

3D-моделювання - це процес створення тривимірних моделей об'єктів за допомогою комп'ютерних програм. Ці моделі ще називають 3D-моделями. За їх допомогою можна відобразити будь-який предмет або середовище, при цьому зберігаючи відповідні розміри. Ці моделі можна оглянути з усіх ракурсів та усунути різні недоліки, які виникають у процесі створення.

Існує 3 основних види тривимірного моделювання: полігональне моделювання, сплайнове моделювання та скульптинг. У переважній кількості випадків віртуальне середовище і персонажі в іграх, анімаційних роликах і кіно створені з використанням полігонального моделювання. Полігони - це трикутники і чотирикутники, які складають сітку на поверхні об'єкту. За допомогою виконання нескладних маніпуляцій з вершинами і ребрами полігонів змінюється форма моделі. Зрозуміло, що добре деталізована модель потребує великих ресурсів. Сплайнове моделювання отримало таку назву через те, що при його використанні об'єкт формується з кривих ліній, які також називають сплайнами. Разом вони утворюють каркас і можуть бути якими завгодно: дуги, багатокутники, кола і т.д. При цьому в будь-який момент можна змінити форму сплайна, тобто навіть готовий об'єкт можна відредагувати. Цифрове моделювання (скульптинг) - це тип тривимірного моделювання, коли користувач взаємодіє з цифровою моделлю так само, як при роботі з глиною. Користувачі можуть штовхати, тягнути, стискати або скручувати віртуальну глину для створення своєї моделі.

Насамперед 3D-моделювання застосовується для фільмів, комп'ютерних ігор, реконструкцій історичних подій, відтворення зображення втрачених пам'яток культури та предметів тощо. Але крім розважального, художнього і культурного застосування у 3D-візуалізації є ще практичне застосування. Наприклад, підприємці для демонстрації власних проєктів та продуктів часто застосовують 3D-модель. Оскільки демонструвати креслення або схеми є не дуже продуктивним, і не кожен їх зрозуміє, будується 3D-модель.

3D-візуалізація допомагає вирішити завдання, коли необхідно донести тонкощі проекту або продукції потенційному споживачеві або інвестору в зручній для сприйняття формі (у вигляді кольорових зображень, відео). Оскільки 3D-модель є свого роду імітацією реальності, зображення виходять досить достовірними і зрозумілими будь-якій людині, а отже якщо споживачеві зрозумілий продукт або проект, то йому простіше прийняти рішення на користь саме цієї продукції.

Аналіз існуючих програм

На сьогодні існує багато різних середовищ розробки 3D-моделювання. Кожне з них має свої напрями та переваги. Однак, найголовнішими програмами для 3D-моделювання є: 3Ds Max, 3D Maya, Cinema 4D, Blender. Вони містять у собі інструменти для моделювання, ріггінгу, анімації, а також базові інструменти скульптингу та багато іншого. Ріггінг – це процес будування скелету моделі, щоб, прив'язавши його до моделі, її можна було рухати. За допомогою кожної з цих програм можна створити з нуля цілу сцену, персонажів, анімувати їх та зробити рендер, тобто отримати зображення або відео моделі. Вони дуже схожі за своїм функціоналом, але в той самий час сильно відрізняються його реалізацією.

В 3Ds max наявна велика кількість інструментів для моделювання та візуалізації. Програма має сильну систему параметризації об'єктів. Можна з легкістю змінити параметри моделі, і всі ці налаштування знаходяться в одному місці. Найголовніше те, що можливостей та інструментів зміни моделі значно більше ніж в інших програмах. В 3Ds Max є діалоговий інструмент під назвою Material Editor, який дозволяє користувачам створювати і редагувати

© ПОМОГАЄВ К.О, АРТЮХ О.А., 2021

матеріали в своїх сценах, застосовувати креативні текстури, а також імітувати заломлення, відображення та інші ефекти при призначенні матеріалів об'єктам. Він має значно більший функціонал ніж інші програми. Саме тому ця програма займає перше місце у сфері візуалізації.

Maya відрізняється від усіх програм великою кількістю інструментів анімації персонажів. За це у програмі відповідає Graph Editor, тобто графік залежностей відстані від часу. Graph Editor являє анімацію в наочному вигляді, а графіки можна легко масштабувати, переміщати, віддзеркалити. У Graph Editor зручно можна додати, видалити і перемістити анімаційні ключі. Можна вручну визначати форму анімаційних кривих. Невелика зміна кута нахилу такої кривої сильно впливає на характер руху персонажа. Схожі інструменти є в інших програмах, але в Maya вони постійно оновлюється і пропонують більше варіантів анімації. Іншою особливістю є те, що сцена збирається з безлічі різних частин: локація, персонажі, камери - все зберігається в окремих файлах. Також сильною стороною Maya є створення тканин. За допомогою технології Maya Fur на дуже високому рівні створюються волосся, трава, хутро тощо.

В Cinema 4D головною перевагою є візуальні ефекти, а саме система x-particles, яка дозволяє робити найсучаснішу симуляцію маленьких частинок інтуїтивно-зрозумілим способом. Із таких частинок складаються такі елементи як вода, дим, туман, вогонь, і їх програма розробляє краще за інших. Також сильною стороною програми є динамічна симуляція. Програма здатна створювати руйнування найбільш універсальним способом. Ще однією особливістю Cinema 4D є набір інструментів Mograph. Він створений спеціально для управління частинками при створенні візуальних ефектів. В інших програмах подібні інструменти реалізовані більш примітивно, тому Cinema 4D знаходиться на іншому рівні в плані розробки ефектів та анімаційної графіки. Саме це відрізняє її від інших програм.

Blender значно відрізняється від своїх конкурентів. Перевагою Blender є те, що взаємодіяти з будь-яким об’єктом можна за допомогою гарячих клавіш. Це значно прискорює процес розробки і ставить Blender на перше місце в плані швидкості. В усіх інших програмах потрібно управляти об’єктом за допомогою миші, а для того, щоб застосувати інструмент або модифікатор на об’єкт, потрібно провести багато рухів та натискань миші. В Blender це робиться за допомогою сполучення кількох клавіш за декілька секунд. Звичайно, в інших програма є гарячі клавіши, але їх дуже мало і застосовуються вони для іншого. Значною особливістю Blender є 2 інтегровані рендер-движки Cycles та Eevee. Cycles базується на трасуванні променів, і тому дозволяє робити високоякісні зображення. Eevee – це рендер-движок, який працює в режимі реального часу. Він видає зображення нижче якістю, але воно не забирає час, і, таким чином, можна подивитися на готове зображення і оцінити його. Інші програми підтримують багато зовнішніх движків, які можна завантажити. Одним з найсильніших рендер-движків є Arnold. Він видає зображення найвищої якості, і є потужним та швидким рендер-движком. Blender не має підтримки таких движків. З однієї сторони мати в собі вже інтегровані Cycles та Eevee, які не потрібно шукати та завантажувати, це плюс. З іншої сторони, найсильніші зовнішні движки видають зображення вищої якості. Знову ж таки, Blender виграє в часі, однак програє в якості. Ще однією ознакою Blender є режим Compositing. Він дозволяє обробляти зображення або відео, які були створені в Blender. Цей режим схожий на програму Photoshop, однак, звичайно, має менший функціонал. В інших програмах такого режиму немає.

Загалом, ці відмінності дуже умовні. Ви можете моделювати, анімувати, створювати різні ефекти та симуляції у будь-якій з цих програм. В цілому, всі ці програми відрізняються наявністю або відсутністю тих чи інших інструментів для здійснення цих процесів. Все залежить від того, що саме вам потрібно.

Вимоги до 3D-моделі

Розроблювана модель повинна відповідати таким вимогам:

1. Давати повну уяву про об’єкти моделювання, враховуючи реалістичність зовнішнього вигляду, відповідність пропорцій розмірів, гами кольорів.
2. Деталізація повинна досягати рівня, який дозволить дати уяву про реальний вигляд та призначення складових предметів та окремих частин в загальній картині об’єкта моделювання (в приміщенні).
3. Огляд повинен здійснюватися з різних ракурсів, повинні зберігатися пропорції та реалістичність вигляду. Огляд повинен бути організований в усіх трьох площинах з місця

спостереження в модельованому приміщенні.

4. Інтерфейс керування повинен бути зручним та інтуїтивно-зрозумілим для середньо-статистичного користувача.
5. Розроблювана модель повинна мати можливість забезпечення подальшого нарощування та модернізації.
6. Необхідно забезпечити можливість віддаленого доступу до моделі за допомогою засобів комунікації в мережі.

Вибір програми та обґрунтування

Для реалізації була обрана програма Blender. Головною її перевагою є швидкість. Система гарячих клавіш дозволить зекономити багато часу, і зробити більше об'єктів у приміщеннях. Програма включає в себе весь необхідний перелік інструментів для розробки 3D-моделі. В програмі наявні такі інструменті як Bevel tool, Merge tool, Extrude tools, які являються базовими та основними інструментами для моделювання. Bevel tool дозволяє створювати округлі кути на об'єктах. Merge tool дозволяє з'єднувати вершини полігонів для зменшення їх кількості. За допомогою Extrude tool можна створювати нові вершини або полігони об'єкта, які автоматично будуть з'єднані з цим об'єктом. Також Blender має такі модифікатори, як Subdivision surface та Boolean tool, які допоможуть створити все, що необхідно для проекта. Subdivision surface автоматично додає моделі більше полігонів, тим самим згладжуючи її. За допомогою Boolean tool можна робити вирізи в об'єктах будь-якої форми.

Режим "Shading" дозволить нанести всі необхідні кольори та текстури на наші об'єкти. У цьому режимі можна комбінувати шейдери для створення матеріалів, таких як скло, дерево, пластмаса, метал та наносити необхідні текстури.

Рендер-движок Cycles, який наявний у програмі, зможе видати високоякісне зображення. Його буде достатньо для проєкту, щоб дати повну уяву про об'єкти моделювання та зробити деталізацію потрібного рівня.

Процес 3D-моделювання

Робота по створенню моделі включає кілька етапів.

1. Розробка концепції та вимог до розроблюваної моделі.

Спочатку були обрані аудиторії, які потрібно змоделювати, а саме 317 та 320. Після цього відбувся процес обговорення ідеї та концепції 3D-моделі, на основі якого були розроблені вимоги до неї.

2. Збір та обробка ескізних матеріалів.

На цьому етапі були зроблені фото аудиторій з різних ракурсів для того, щоб максимально точно можна було відтворити приміщення. Наявність фото допомагає побачити всі деталі об'єктів і не тримати їх в голові.

3. Створення моделі за допомогою обраного середовища моделювання (розробки).

Рис. 1 Аудиторія 317

Процес 3D-моделювання розпочався з побудови самої кімнати, а саме стін, вікон, дверей, підлоги та стелі. Потім були додані об'єкти інтер'єра, які розроблювалися окремо. Після розробки кожної моделі на них були нанесені кольори та текстури, а потім було виставлене освітлення. Фінальною частиною стало розміщення камери та рендер. На малюнку 1 можна побачити зображення аудиторії 317.

4. Створення інтерфейсу користувача для керування моделлю.

Моделі приміщень мають зручний інтерфейс у форматі віртуального тура. Потрапивши на сторінку зображення, можна керувати переміщенням за допомогою миші або клавіш на клавіатурі. Натиснувши та придержуючи ліву або праву клавішу миші, можна переміщати камеру на 360 градусів на всі сторони. Покрутивши колесо миші вперед можна приблизити зображення, а якщо покрутити назад, можна його віддалити.

Натиснувши на клавіатурі стрілку вліво або вправо, можна відповідно переміщати камеру вліво або вправо на 360 градусів. Також за ці функції відповідають клавіші A і D. Щоб перемістити камеру вгору або вниз, треба натиснути клавіші W і S відповідно. За допомогою клавіші Z можна приблизити зображення, за допомогою клавіші Q можна його віддалити.

Для тих, хто оглядає модель зі свого смартфона, для того, щоб пересувати камеру треба провести пальцем по зображенню у відповідному напрямку. Також переміщати камеру можна, пересуваючи сам телефон. Для того, щоб приблизити зображення, треба натиснути двома пальцями на екран та розвести їх в сторони. Для того, щоб віддалити зображення, треба звести пальці.

5. Вирішення питань забезпечення доступу до моделі широкого загалу користувачів.

Для того, щоб користувачі мали доступ до моделі, її було розміщено на сайті. Таким чином, кожен може переглянути модель з комп'ютера та смартфона. Аудиторії можна оглянути за наступними посиланнями: https://kuula.co/post/7w0gp - аудиторія 317, https://kuula.co/post/73k96 - аудиторія 320.

ЛІТЕРАТУРА

1. King Roger. 3D Animation for the Raw Beginner Using Autodesk Maya. 2nd Edition. — Taylor & Francis Group, LLC, 2019. — 450 p. — ISBN 9780815388791.
2. Danan Thilakanathan. 3D Modeling For Beginners: Learn everything you need to know about 3D Modeling! 1st Edition - ISBN-13: 978-1530799626.
3. John M. Blain. The Complete Guide to Blender Graphics: A Beginner's Guide. 1st Edition - ISBN-13: 978-1466517035
4. Pradeep Mamgain. MAXON Cinema 4D R20: Modeling Essentials. Kindle Edition
5. Brian L. Smith. 3ds Max Design Architectural Visualization: For Intermediate Users. 1st Edition - ISBN-13: 978-0240821078
6. https://docs.blender.org/manual/en/latest/

ПОМОГАЄВ Костянтин Олександрович – студент факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: xa11867765@student.karazin.ua; ORCID: 0000-0002-5010-2369.

Наукові інтереси:

– Розробка 3D-моделей місцевості та предметів, анімація, створення художніх фільмів.

АРТЮХ Олексій Анатолійович – ст. викладач кафедри ТПС факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: oleksiy.artiuh@student.karazin.ua; ORCID: 0000-0002-2038-4131.

Наукові інтереси:

– Мережеві технології, моделювання інформаційних процесів в мережах.

УДК 004.422.8 **СЕКЦІЯ 7**

ПОНОМАРЕНКО В.В.

ПОРІВНЯННЯ І АНАЛІЗ CONTENT MANAGEMENT SYSTEM ТА WEB CONTENT MANAGEMENT SYSTEM

Причини для проведення порівняння

«Система управління контентом» є універсальним терміном для систем, які забезпечують роботу більшості веб-сайтів, управляють, здебільшого, внутрішнім контентом і поширюють його по різним адресатам. Цей загальний термін може привести до великої плутанини - в основному для тих, хто купує таке програмне забезпечення, - оскільки він призводить до того, що самі галасливі платформи привертають найбільшу увагу, навіть якщо вони в певних сценаріях підходять найгіршим чином.

Приклад.Індустрія досить добре переконує багатьох власників веб-сайтів, що WordPress може виконати роботу за них, тобто будь-яку роботу. Однак ця платформа має безліч обмежень і проблем безпеки, які необхідно перевірити перед фіксацією. Кращим варіантом для споживачів цих платформ є пошук більш конкретних термінів, таких як перераховані нижче, щоб знайти рішення, яке більш точно відповідає їх вимогам.

Що таке CMS(Content Management System). Системи управління контентом (CMS) були представлені видавцям контенту ще в 1982 році. Першою була система управління підприємством (EMS) IBM під назвою FileNet. Заснована Тедом Смітом, який раніше працював над Basic 4, на початку 1990-х вона перейшла від пропрієтарної до більш «відкритою» версії.

Переходячи до основ, в найпростіших термінах CMS - це «система», яка «керує» контентом. Його часто визначають як програмну платформу, яка надає набір процедур, які використовуються для управління робочим процесом в середовищі спільної роботи.

Типова система управління вмістом (CMS) використовуватиме скрипт на стороні сервера, такий як PHP, для виклику вашого «контенту» з бази даних, наприклад MySQL (My Structured Query Language), та динамічної подачі його на ваш вебсайт. Цей самий метод він використовуватиме, щоб дозволити публікувати контент, редагувати цей контент та додавати або видаляти функціональність вашого веб-сайту за допомогою так званого адміністративного «back-end».

Його переваги для неспеціаліста або новачка стають очевидними. Фактична функціональність, простота використання та налаштування системи управління контентом (CMS) сильно відрізнятиметься від однієї платформи до іншої, але в більшості випадків вони розроблені таким чином, щоб дозволити непрограмісту, який практично не має навичок в кодуванні, публікувати вміст автоматизовано.

Ця автоматизація надає користувачам додаткові можливості, такі як шаблони, легко редагований вміст, підтримка багатомовності, поширення контенту, су місна робота, управління робочими процесами та ще десятки плагінів, модулів та варіантів оновлення.

Що таке WCMS(Web Content Management System). WCMS - це специфічний вид CMS. Спеціально призначена для веб-сайтів, це підмножина CMS, відома як «Система управління веб-контентом». WCMS - це система веб-управління, яка також обробляє вміст. По суті, це програмне забезпечення, призначене для допомоги в створюванні вмісту та керування ним для поширення в Інтернеті. Це включає в себе управління мовами сайту, створення мікросайтів, управління кількома сайтами в одному інтерфейсі. WCMS також допоможе вам створити та підтримувати веб-сайт, навіть якщо вам не вистачає навичок програмування. Це вигода для малого бізнесу або підприємця, який не має бюджетів або необхідних знань з кодування, щоб самостійно створити та підтримувати складний веб-сайт.

Ключова різниця між WCM та CMS. Не вдаючись в деталі, система управління контентом (CMS) - це комп'ютерна програма цифрового контенту, яке підтримує кілька користувачів в просторі для спільної роботи. На відміну від WCMS, вона керує контентом з чітко визначеними правами власності, що належать до документів, фільмів, фотографій, номерів телефонів і точок наукових даних.

© ПОНОМАРЕНКО В.В., 2021

Важливо відзначити, що існує галузева дилема при обговоренні відмінностей між CMS і WCMS. Вони часто використовуються як взаємозамінні, тому що між ними дуже багато спільного. Справа в тому, що система управління контентом (CMS) часто містить компоненти WCMS, створюючи сірі зони для визначення відмінностей між WCMS і CMS.

Управління веб-сайтом і управління контентом часто виконуються разом, але можуть виконуватися окремо. Маючи це на увазі, думайте про систему управління веб-контентом як про симбіоз між системами веб-управління та системами управління контентом. За допомогою WCMS і CMS ви можете створювати, публікувати і поширювати контент між пристроями. За допомогою WCMS ви також можете керувати сайтом або сайтами, а не тільки контентом.

У загальних рисах, зовнішня чи внутрішня CMS пропонує функції для управління веб-контентом, в той час як WCMS використовує ці функції і розширює їх в різних областях, таких як: сумісна праця; управління цифровими активами; управління робочими процесами / Управління бізнес-процесами; управління веб-контентом; управління знаннями; візуалізація та управління документами; підтримка декількох сайтів одночасно; вбудовані SEO-рішення; ведення документації.

Що таке ECM(Enterprise Content Management). ECM - це ще один термін, про який варто знати, - це управління корпоративним контентом. На відміну від WCMS або CMS, ECM полегшує спільну роботу над контентом на робочому місці. Це не одна частина програмного забезпечення, а скоріше збірний термін для позначення набору процесів, інструментів, програмного забезпечення і стратегії, пов'язаних з корпоративним контентом. Це досягається за рахунок інтеграції управління документами і надання кінцевим користувачам рольового доступу до цифрових активів організації.

Висновок

Всі ці терміни ускладнюють ситуацію, особливо якщо ви хочете знайти платформу або постачальника послуг для допомоги в вашому проекті. Яке рішення підійде саме вам? Це залежить від ваших потреб і вимог. Всі екземпляри завжди різні, і саме тут індустрія CMS надає клієнтам величезну ведмежу послугу, особливо коли ваша CMS служить центром всієї вашої організації. Основні кроки, які ви повинні зробити, починаються з оцінки кожної платформи, як згадувалося вище, по її можливостям, а не за назвою. Потім вам потрібно дійсно проаналізувати, в чому полягає ваш ключовий контент. Відповідь на це питання у всіх різна, але у кожної компанії є свій тип контенту, який їх відрізняє.

Наприклад: якщо ви є дистриб'ютором електроінструментів, список вашої продукції буде вашим контентом. Якщо ви медіа-компанія, що спеціалізується на відео, то відео - це ваш основний контент. Очевидно, що якщо ви ведете блог, повідомлення - це ваш основний контент.

Для деяких з цих прикладів доступні відмінні платформи, які заповнюють цю нішу і відмінно з нею справляються. Я не буду сперечатися - не дивлячись на те, що ви припускали з попередніх повідомлень, - що WordPress відмінно підходить для розміщення простого блогу. Але чи буде це хорошою платформою для розміщення всього процесу виробництва відео, інтеграції декількох каналів поширення і масштабування для обробки великої кількості відвідувачів? Скоріш за все ні.

ЛІТЕРАТУРА

1. Deane Barker Web Content Management: Systems, Features, and Best Practices 1st Edition. O'Reilly Media, 2016. -378c.
2. Web Content Management System (WCMS). URL: https://techsofto.wordpress.com/2020/01/10/web-content-management-system-wcms/
3. WCMS vs CMS (and Everything Else): Explaining The Difference URL: https://www.npgroup.net/blog/wcms-vs-cms-and-everything-else-explaining-the-difference/
4. What is a Content Management System (CMS) | The Complete Guide URL: https://www.zesty.io/mindshare/marketing-technology/what-is-a-content-management-system-cms-the-complete-guide/

ПОНОМАРЕНКО Владислав Вікторович – студент; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, м. Харків, Україна, 61022; e-mail: xa11867702@student.karazin.ua; ORCID: 0000-0001-5874-8677.

Наукові інтереси: – *аналіз різноманітних систем управління.*

УДК 004.415.5 **СЕКЦІЯ 6**

ПУДОВКІНА Л.Ф.

СЦЕНАРІЇ ТЕСТУВАННЯ ВИМОГ І ЯКІСТЬ ПРОГРАМ

Ефективність управління процесом розроблення системи, управління всіма етапами технологічного процесу, безпосередньо залежить від розроблення специфікацій різного призначення, всіляких видів проектування, верифікації та валідації. Треба виконувати не тільки всі види тестування, згідно технологічного циклу процесу створення системи, але прогнозувати на самих ранніх фазах особливості системи, що визначають її якість і конкурентоздатність.

Неоднозначні і нечіткі специфікації і сценарії вимог - це завжди передумова для некоректного виконання тестування, що визначає якість створюваного програмного забезпечення.

Формувати функціональні вимоги потрібно з урахуванням набору характеристик і критеріїв для встановлення і оцінки якості програмного забезпечення, його компонентів та складових частин його архітектури. З надійністю функціонування [1] сучасного програмного забезпечення тісно пов'язаний обгрунтований показник його якості [2,3].

Якість програмного забезпечення [2,3] характеризується шістьма характеристиками першого рівня. До першого рівня відносяться показники якості, які характеризуються найбільшою різноманітністю значень властивостей додатків і наборів даних і охоплюють весь спектр класів, призначень і функцій сучасних систем. Серед показників якості до цієї групи належить функціональна придатність, що є домінуючою характеристикою будь-яких систем. Значення інших показників якості безпосередньо визначаються необхідними функціями програмного забезпечення і, в тій чи іншій мірі, впливають на виконання цих функцій.

Функціональна придатність - це найбільш відповідальна, об'єктивно важко формалізується і оцінюється в проекті характеристика комплексів програм. Дана характеристика пов'язана з тим, які основні і додаткові функції і завдання повинен вирішувати програмний продукт для задоволення потреб користувачів.

Конструктивні характеристики пов'язані з тим, як і за яких умов, задані функції можуть виконуватися з необхідною якістю. Атрибути функціональної придатності можна характеризувати в основному з його властивостями, категоріями і якісним описом функцій, для яких важко визначити чисельні заходи і шкали.

До другого рівня показників якості відносяться достовірно і об'єктивно вимірювані кількісні показники програмного забезпечення. Значення цих конструктивних характеристик найбільшою мірою впливають на функціональну придатність у використанні ПЗ. Тому вибір і обгрунтування їх необхідних значень повинно проводитися ретельно вже на етапі проектування програмного забезпечення. Їх субхарактерістікі можуть бути описані впорядкованими шкалами вимірюваних значень, необхідні чисельні величини яких можуть бути визначені замовниками або користувачами ПЗ. Такими характеристиками є надійність функціонування і ефективність комплексів програм. Ці величини можуть фіксуватися в специфікації вимог, із зазначенням методики їх вимірювань або порівнянням з показниками фізичних випробувань. Інтервал часу, тривалість очікування результатів (відгуку) і деякі інші характеристики функціонування програмного забезпечення можуть бути обрані і кількісно вказані в специфікаціях вимог замовником.

Третій рівень стандартизованих показників якості програмного забезпечення важко повністю описати вимірюваними кількісними значеннями і їх деякі субхарактерістікі і атрибути мають описовий, якісний вид. Залежно від функціонального призначення програмного продукту за погодженням із замовником можна визначати ступінь необхідності (пріоритет) цих властивостей і значення рівня реалізації їх атрибутів в життєвому циклі конкретного ПЗ. Проблема полягає у виявленні факторів, від яких вони залежать, в створенні методів і засобів зменшення їх впливу на функціональність програмного забезпечення, необхідної якості функціонування комплексу програм

Якість програм покращує валідація вимог. Це процес виявлення помилок в розроблюваних сценаріях., Процес ітераційний, який складається з наступних кроків.

© ПУДОВКІНА Л.Ф., 2021

1.Формалізований опис вимог у вигляді сценаріїв. Сценарій, описаний за допомогою прецеденту, визначає необхідну поведінку системи з точки зору певного класу користувачів визначає порядок тестування конкретної функціє, дії, режиму.

Прецеденти є структурованою і розумно формальної нотацією для фіксації дуже важливого підмножини інформації про вимоги: як система взаємодіє з користувачем при наданні своїх функціональних можливостей.

Кожен виявлений прецедент може практично без зміни використовуватися в якості сценарію тестування під час будь-якого виду контролю. Сценарій містить крім умов його виконання та статусу користувача наступну інформацію: опис стану програмного забезпечення на початку сценарію, опис нормального протікання подій і опис інформації і дій, які можна здійснити під час виконання сценарію.

2. Наступний крок це додавання опису всіляких виняткових ситуацій і способів їх обробки, детальний перелік всіляких альтернативних умов. Це дозволяє обгрунтовано вибрати критерії аналізу результатів виконання всіх видів тестування, визначати особливості та набори тестових даних для тестування. На цьому етапі потрібно ефективно організувати, оптимізувати і структурувати набори даних для тестів, які залежать від ретельно обраної стратегії виконання тестування, застосовуваних методів.

3.Потім на основі цих сценаріїв проводиться їх валідація, перевірка завершення всіх умов і особливостей процесу валідації. Аналізуються експлуатаційні якості системи і спрощуються умови супроводу системи, уточнюється архітектура системи. Система більше відповідає потребам користувачів.

При виявленні будь-яких неточностей треба проводити повторення кроків. Але перед повторним виконанням розроблення та аналізу сценаріїв тестування вимог треба ретельно повторно вибрати і уточнити характеристики якості і критерії його оцінки. Крім цього треба проаналізувати ще й такі атрибути [4] вимог як стабільність і необхідність, які можуть вплинути на реалізацію альтернативних потоків подій в сценаріях.

Висновки. Прецеденти треба застосовувати в процесі проектування тестування, вони забезпечують цілісне уявлення і послідовність узгоджених дій не тільки по виявленню вимог, аналізу, проектування системи, вони сприяють виконанню верифікации та валідації системи за більш короткий термін часу.

Рівень досягнення якості функціонування програмного забезпечення може бути передбачуваним, керованим, залежить від ресурсів, що виділяються на його досягнення. Рівень також залежить від характеристик якості програмного забезпечення і способів їх оцінки, від ефективності використовуваних технології та відповідальності розробників.

Тому необхідній рівень якості програмного забезпечення та надійність його функціонування може бути передбачуваними і керованими. Це позитивно вплине на конкурентоспроможність створюваних сучасних програмних засобів згідно вимог стандартів [5]. При цьому скорочуються витрати на розроблення системи.

ЛІТЕРАТУРА

1. Майерс Г. Надежность программного обеспечения / пер. с англ. Ю.Ю. Галимова. – М. : Мир, 1980. – 356 с.
2. Метрика Холстеда [Електронний ресурс] – Режим доступу до ресурсу: https://studfiles.net/preview/6196491/page:5/
3. Липаев В. В. Качество программных средств / В. В. Липаев. -Москва: Синтег, 2002. 216 с.
4. Конорев Б.М., Пудовкина Л.Ф., Сироджа И.Б., Федорович О.Е. Нормативная база программной инженерии в разработке систем с интенсивным использованием программного обеспечения. Учеб.пособие. – Х.: Нац. аэрокосм. ун-т им. Н.Е. Жуковского «ХАИ», 2001.162 с.
5. Настонови щодо документування комп'ютерних програм. Національний стандарт України ДСТУ 4302: 20004.

ПУДОВКІНА Лариса Федорівна – к.т.н, доцент кафедри інженерії програмного забезпечення Національного аерокосмічного університету імені М. Є. Жуковського «Харківський авіаційний інститут» » Україна, 61070, Харків-70, вул. Чкалова, 17, e-mail l.pudovkina@khai.edu.

Наукові інтереси: *моделі надійності функціонування і моделі оцінки якості ПЗ.*

УДК 004.5 **СЕКЦІЯ 2**

РОСОХАТИЙ К.С.

МОБІЛЬНА GPS-ОРІЄНТОВАНА СИСТЕМА ДОПОВНЕНОЇ РЕАЛЬНОСТІ ДЛЯ ТУРИСТИЧНОЇ СФЕРИ

Розвиток потужності мобільних пристроїв спричинило зростання популярності технології доповненої реальності, зокрема в сфері туризму. Поява та розвиток такої технології полегшує повсякденне життя і дає нові можливості [1].

Дане дослідження присвячене можливості використання доповненої реальності в сфері туризму, в великих та малих містах для зручного орієнтування в реальному часі. Метою дослідження є розробка мобільного програмного забезпечення для сфери туризму, вчасності пошук необхідного місця та побудови шляху доповненої реальності для покращення орієнтування в незнайомому місці або пошук незнайомого об'єкту.

Проаналізувавши декілька мобільних додатків зі схожим функціоналом можна скласти вимоги до апаратного та програмного забезпечення телефону для використання доповненої реальності в сфері туризму (табл. 1). Для забезпечення коректної роботи мобільного додатку необхідне відповідне програмне та апаратне забезпечення.

Табл.1 Функціональні вимоги до апаратних складових мобільного телефону

Складова частина	*Перелік вимог*
Процесор	1) не пізніше 2018 року; 2) знаходиться в переліку AR сумісних процесорів в Google ARCore; 3) кількість ядер два або більше.
Камера	1) мінімум 12 MP; 2) наявність стабілізації.
Датчики	1) наявність акселерометра; 2) наявність гіроскопа; 3) наявність датчика світла; 4) наявність барометра; 5) наявність компаса.
Програмне забезпечення	1) Android версії 8.1 Oreo та вище; 2) для мобільних пристроїв Apple версія iOS 11.0 та вище; 3) встановлений додаток Google ARCore.
Комунікації	1) GPS; 2) GLONAS.
Доступ до інтернету	1) 3G або 4G, LTE чи 5G або Wi-Fi 2) швидкість інтернету під час використання додатку повинна бути не менша за 8 Мбіт/с.

Розглядання способу побудови ядра для додатку доповненої реальності приділяється багато уваги на сьогоднішній день. Наведемо деякі з них.

WebAR - це технологія доповненої реальності, доступ до якої здійснюється через веб-браузер, а не через додаток. Він забезпечує цей веб-інтерфейс AR з використанням таких технологій, як WebGL, WebRTC, WebVR і API. Перевага такого способу є те що не потрібно встановлювати додатки, достатньо відкрити посилання [2].

AR.js - полегшена бібліотека для доповненої реальності в інтернеті, що підтримує створення AR як на основі маркерів, так і просторову. Містить функцію відстеження зображень і являє собою безкоштовний багатоплатформовий репозиторій [3].

Argon.js - JavaScript-фреймворк для додавання AR-можливостей до веб-додатків без прив'язки до використовуваної платформи і технології. Спочатку Argon.js створювався під браузер Argon на iOS з вбудованою підтримкою AR, однак цей фреймворк працює і з іншими браузерами [4].

Awe.js дозволяє створювати AR-додатки для мобільних пристроїв за дещо іншим принципом: на основі маркерів, розташування і датчика руху Leap Motion. Для доступу та взаємодії з доповненою реальністю платформа використовує WebRTC, WebGL і getUserMedia API пристрою [5].

© РОСОХАТИЙ К.С., 2021

ARToolKit - безкоштовна бібліотека з відкритим вихідним кодом для розробки доповненої реальності. Крім Android і iOS, ARToolKit підтримує також додатки на Windows, Linux і OS X. Бібліотеки відстеження відео ARToolKit розраховують реальний стан і орієнтацію камери щодо фізичних маркерів в режимі реального часу. Ще в ньому є мультиплатформенна відеотека з підтримкою декількох джерел і форматів, відстеження декількох камер, проста процедура калібрування, проста графічна бібліотека, модульний API і повний набір шаблонів і утиліт [6].

ARCore - це інструмент для розробки програмного забезпечення, розроблений Google, який дозволяє створювати додатки доповненої реальності. Головними перевагами цього інструменту є: відстеження руху яке дозволяє смартфону зрозуміти своє становище в реальному світі; розуміння навколишнього середовища - воно дозволяє смартфону визначати розмір і місце розташування всіх типів поверхонь - вертикальних, горизонтальних і кутових; оцінка освітленості - дозволяє смартфону оцінити поточні умови освітлення навколишнього середовища [7].

Обґрунтування та вибір інструменту для розробки мобільного додатку

Основним інструментом для розробки мобільного додатку було обрано ARCore Google та Google API через те що даний інструмент може визначати розмір і місце розташування всіх типів поверхонь, що особливо важливо в пошуку шляху доповненої реальності.

В даній роботі для тестування додатку був використаний смартфон Huawei P30 Pro, оскільки відповідає всім вимогам необхідним для використання технології доповненої реальності.

Ядро доповненої реальності було встановлено з Play Market що говорить про те що даний смартфон підтримує інструмент ARCore це можна побачити на рис. 1 де зображено що в телефоні встановлене API AR інструменту.

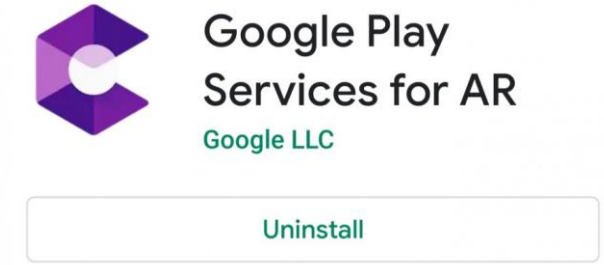


Рис. 1 Google Service AR

В якості API карт було використано Google API, оскільки компанія постійно оновлює та вдосконалює всі свої дані, тому карти Google одні із самих точних в світі при глобальному покритті. Через те що Google Maps API дозволяє використовувати карти, високоякісні знімки, панорамні вулиці, інформацію про завантаженість дороги, цікаві місця та багато іншого, без необхідності інвестувати у створення та оновлення власних базових карт та даних було обрано саме ці сервіси в якості завантаження необхідних місць які потребуватиме знайти користувач.

Було досліджено що API Карт Google підтримує мобільні пристрої, планшети та настільні комп'ютери та не вимагав конфігурації під окремі браузери, а також розміщував засоби для розробки мобільних пристроїв для Android та iOS то ці данні доцільно було використано для даного додатку.

Для роботи с картами було створено API Key, – ключовий інтерфейс прикладного програмування, унікальний ідентифікатор, який використовується для аутентифікації користувача, розробника або виводу програми в API.

Ключ використовується для доступа к API карт Google. Маючи такий ключ, можна отримати ліміт безкоштовних запитів користувачів або виставити необхідний дозвіл на

кількість запитів. Також з допомогою такого ключа можна отримати статистику запитів і контролювати їх при необхідності.

Такий ключ отримуємо на сайті для розробників Google.

В консолі розробника на ключ API було встановлено обмеження на допуск використання тільки API Maps.

Проаналізувавши роботу додатку, можна дійти висновку, що дане мобільне програмне забезпечення полегшить повсякденне життя під час подорожі чи просто орієнтуванню в незнайомому місці в реальному часі, без звернень в компанії які надають туристичні послуги або екскурсії. В такий спосіб можна зберегти час, гроші та бути впевненим в особистій безпеці в умовах карантину.

ЛІТЕРАТУРА

1. Дополненная реальность. [Електронний ресурс] Режим доступу: https://www.ptc.com/ru/technologies/augmented-reality (останнє звернення: 11.03.2021).
2. WebAR: How It’s Changing the Future of Augmented Reality. [Електронний ресурс] Режим доступу: https://perfectial.com/blog/webar/ (останнє звернення: 12.03.2021).
3. Ar.js. [Електронний ресурс] Режим доступу: https://weatherless.ru/javascript/js-other/ar-js/ (останнє звернення: 11.03.2021).
4. A javascript framework for adding augmented reality content to web applications. [Електронний ресурс] Режим доступу: https://www.argonjs.io/ (останнє звернення: 09.03.2021).
5. Дополненная реальность с Awe.js. [Електронний ресурс] Режим доступу: https://coderlessons.com/articles/veb-razrabotka-articles/dopolnennaia-realnost-v-brauzere-s-awe-js (останнє звернення: 11.03.2021).
6. How ARToolKit works? [Електронний ресурс] Режим доступу: http://www.hitl.washington.edu/artoolkit/documentation/userarwork.htm (дата звернення: 14.03.2020).
7. ARCore. [Електронний ресурс] Режим доступу: https://developers.google.com/ar/ (останнє звернення: 16.03.2021).

РОСОХАТИЙ Костянтин Сергійович – студент кафедри теоретичної та прикладної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: rosokhatyi2020ku51@student.karazin.ua; ORCID: 0000-0001-6234-4130.

Наукові інтереси:

– *веб-програмування;*

– *автоматизація та комп’ютерно-інтегровані технології.*

УДК 004.85 **СЕКЦІЯ 3**

РУЖАНСЬКА А. В., ВАСИЛЬЄВА Л. В.

ОГЛЯД ТА АНАЛІЗ БІБЛІОТЕК КОМП'ЮТЕРНОГО ЗОРУ ДЛЯ ОБРОБКИ ЗОБРАЖЕНЬ, ВИКОРИСТОВУВАНИХ В МАШИННОМУ НАВЧАННІ

Вступ

Потреба в цифровій візуалізації зростає все більше в IT-індустрії. Кожен день ми використовуємо ці рішення як користувачі, не звертаючи особливої уваги на складність і високотехнологічність апаратних і програмних систем.

Наприклад, в наші дні відносно нормально мати автопілот в наших автомобілях, використовувати безпілотники для оцінки стану посівів на полях або мати магазини, в яких на касі розпізнаються товари по упаковці. І це лише декілька прикладів, що демонструють можливості цифрової обробки зображень.

Цифрова обробка зображень (image processing) - це використання комп'ютерних алгоритмів, спрямований на поліпшення якості цифрових зображень або вилучення інформації для подальшого використання. Основним завданням обробки є корекція зображення і виконання основних операцій (кадрування, відображення, обертання, сегментація, класифікація, витяг ознак, відновлення і розпізнавання), а також фільтрація та афінні перетворення. Обробка зображень, також називається аналізом зображень, що фокусується на роботі з 2D-зображеннями для перетворення одного цифрового зображення в інше [1].

Data scientists зазвичай попередньо обробляють зображення для того, щоб використовувати набір зображень в машинному навчанні для досягнення бажаних результатів. Крім того, комп'ютерний зір також використовує алгоритми обробки зображень для вирішення багатьох інших завдань.

Найбільш поширим засобом для вирішення завдання обробки зображень є мова Python. Завдяки доступності і зростаючій популярності Python в якості мови науково-технічного програмування, з'явилося безліч першокласних інструментів для обробки зображень. Отже, мета даної роботи – розглянути та проаналізувати можливості різних бібліотек з обробки зображень для подальшого використання в машинному навчанні.

Аналіз бібліотек

Розглянемо найбільш популярні сучасні бібліотеки обробки зображень на мові Python.

OpenCV-Python

OpenCV (Open Source Computer Vision Library) - одна з найпопулярніших бібліотек для додатків по комп'ютерному зору.

OpenCV-Python - це Python-версія інтерфейсу для OpenCV. Наявність коду на C / C++ в бекенді гарантує швидкість бібліотеки, а Python-обгортка у фронтенді забезпечує легкість налаштування і розгортання. Завдяки цьому OpenCV-Python є відмінним рішенням для високонавантажених обчислювальних програм по комп'ютерному зору [2]. Бібліотека має такі основні можливості обробки зображень:

- Обрізка, зміна розміру, масштабування, повороти;
- Градація сірого і поріг, видалення шуму, розмиття / згладжування;
- Малювання геометричних фігур та розміщення тексту на зображенні;
- Розпізнавання обличь та силуетів людей;
- Контурне розпізнавання об'єктів;
- Класифікація зображень;
- Пошук подібних зображень.

© РУЖАНСЬКА А. В., ВАСИЛЬЄВА Л. В., 2021

Це деякі з найбільш основних операцій, які можуть бути виконані за допомогою OpenCV на зображенні. Крім того, OpenCV використовується в завданнях комп'ютерного зору, таких як сегментація зображень, виявлення об'єктів, тривимірна реконструкція, вилучення об'єктів.

Scikit-image

Scikit-image - це набір алгоритмів обробки зображень. Бібліотека є безкоштовною і без будь-яких обмежень. Бібліотека є досить простою і зрозумілою, навіть для тих, хто новачок в екосистемі Python. Scikit-image незамінний за своїми характеристиками для обробки і фільтрації зображень. Крім цього, деякі частини бібліотеки Scikit-image написані на Cython (Cython - це мова програмування, яка є підмножиною мови програмування Python, призначеної для забезпечення продуктивності, подібної мові програмування C) [3]. Основними функціями бібліотеки Scikit-image є:

- Сегментація зображення;
- Геометричні перетворення;
- Маніпуляція колірним простором;
- Математична морфологія;
- Фільтрація;
- Виявлення ознак.

Бібліотека Scikit-image має цінний модуль морфології, який може бути використаний для створення структурованих елементів у зображенні. Сегментація, перетворення, експонування та багато інших алгоритмів роблять цю бібліотеку Python однією з найкращих для обробки зображень. Також Scikit-image використовує інтерфейс NumPy для зображень та OpenCV. Це робить ці дві бібліотеки є сумісними, даючи користувачам можливість комбінувати різні методи для обробки зображень за допомогою обох бібліотек.

PIL/Pillow

PIL (Python Imaging Library) - це бібліотека з відкритим вихідним кодом для простих завдань обробки зображень. Ця бібліотека може бути використана лише для досить простого маніпулювання зображеннями, бо PIL не зазнав жодних змін або розвитку з 2009 року [4]. PIL може виконувати такі завдання над зображенням, як:

- Обрізка, зміна розміру, масштабування, повороти;
- Збереження в різних форматах зображень;
- Відображення та архівування зображень.

Pillow - це форк PIL, яка простіша в установці, працює на всіх основних операційних системах, підтримує Python 3 і перетворилася на поліпшену, сучасну версію бібліотеки PIL. Форк Pillow був зроблений для того, щоб встановлювати і збирати Python Imaging Library на сучасних системах. Pillow є нативним модулем для Python, половина коду якого написана на C, половина — на Python [4]. Основними можливостями бібліотеки є:

- Геометричне перетворення;
- Генератор зображень;
- Фільтрація зображень;
- Перетворення колірного простору;
- Малювання на зображеннях.

Ця бібліотека широко використовується для перетворення зображень у веб-проектах, так як вона більш легка і зручна у використанні, якщо не потрібна більш складна функціональність OpenCV або scikit-image.

Pgmagick

Pgmagick – це обгортка на базі Python для бібліотеки GraphicsMagick. Систему GraphicsMagick іноді називають швейцарським ножем в обробці зображень. Вона пропонує колекцію ефективних інструментів і бібліотек, що підтримують читання, запис і операції із зображеннями в більш ніж 88 основних форматах, включаючи DPX, GIF, JPEG, JPEG-2000, PNG, PDF, PNM і TIFF [5].

Pgmagick володіє широким спектром корисних функцій:

- Перетворення зображення з одного формату в інший.
- Зміна розміру, поворот, різкість, зменшення кольору або додавання спеціальних ефектів до зображення.
- Монтаж мініатюр зображень.
- Створення GIF-анімації.
- Малювання фігур або текст на зображенні.

Цей функціонал робить Pgmagick універсальним і потужним інструментом зображення для багатьох завдань побудови бекенда. Обробка зображень є багатопотоковою, і це означає, що можна масштабувати обробку зображень на декілька процесорів сервера OpenMP. Ця функція є ключовою для пакетної обробки зображень, особливо коли необхідно обробити мільйони файлів.

Mahotas

Mahotas є бібліотекою Python для комп'ютерного зору та обробки зображень. Вона містить:

- Стандартні функції по обробці зображень (фільтри і морфологічні операції);
- Виявлення особливих точок;
- Локальні дескриптори.

Mahotas, як і OpenCV і SimpleCV, покладається на масиви NumPy. Більша частина функцій, реалізованих в Mahotas, може бути знайдена в OpenCV або SimpleCV, але в деяких випадках інтерфейс Mahotas буде простіше у використанні, особливо коли мова йде про пакети функцій [6].

Порівняльний та аналіз бібліотек

При порівнянні бібліотек не можна вказувати в цілому, що якась бібліотека краще і працює швидше, ніж інша. Але можно виділити переваги з окремих функцій, скласти набір операцій і заміряти результати виконання.

В даній роботі було обрано такі операції, як завантаження зображення, поворот на 180 градусів, зменшення зображення в 2 рази, розмиття і збереження зображення в формат JPEG. Результат наведено у Таблиці 1:

Табл.1 Результати порівняння бібліотек для обробки зображення на наборі операцій

Бібліотека	Mpx/s
OpenCV-Python	43.3
Scikit-image	35.6
Pillow	24.7
Pgmagick	12.4
Mahotas	33.2

Даний результат був отриманий на Windows 64-bit, Intel(R) Core(TM) i7-7500U CPU 2.70 GHz. На різних машинах результат може відрізнятися.

Як бачимо, OpenCV-Python на цьому наборі працює найшвидше, найдовше за всіх працює Pgmagick. Швидкість Scikit-image є оптимальною і по результатом на другому місці. У Mahotas більша продуктивність ніж у Pillow.

Висновки

Таким чином у теперішній час працювати з завданнями цифрової обробки зображень,набагато простіше ніж, скажімо, з 5-10 роками тому. Це досягається за рахунок розробки різних бібліотек і методів обробки, а також за рахунок розвитку обчислювальної техніки і процесорів.

Крім того, підвищується зручність використання цих бібліотек і методів. Це досягається за рахунок використання скриптових мов, і при необхідності можна написати свою частину алгоритму на C++ і підключити його до скриптової мови, Це зменшує обсяг коду, який повинен бути написаний для виклику певного методу з бібліотеки.

В результаті роботи було розглянуто та проаналізовано різні можливості бібліотек для обробки зображень з подальшим використанням в машинному навчанні, встановлено, що існує цілий ряд корисних і безкоштовних бібліотек з обробки зображень в Python, де кожна бібліотека має свої цілі та може краще підійти під різні потреби в обробці зображень. Результати цієї роботи можуть бути корисні для початківців в галузі машинного навчання та для студентів

ЛІТЕРАТУРА

1. Л. Шапиро, Дж. Стокман. Компьютерное зрение = Computer Vision. М.: Бином. Лаборатория знаний. 2006. 752 с.
2. Хоузе, Дж., П. Джоши, М. Бэйелер. OpenCV: Computer Vision Projects with Python: Packt Publishing Ltd, 2016. 558 p. URL: https://www.packtpub.com/product/opencv-computer-vision-projects-with-python/9781787125490 (Last accessed: 24.03.2021).
3. Van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J. scikit-image: image processing in python. 2014. PeerJ 2, 453 p. URL: https://doi.org/10.7717/peerj.453 (Last accessed: 25.03.2021).
4. Alex Clark, Fredrik Lundh. Pillow (PIL Fork). 2011. URL: https://pillow.readthedocs.io/en/latest/index.html (Last accessed: 26.03.2021).
5. pgmagick 0.5.1 documentation. URL: https://pgmagick.readthedocs.io/en/latest/#license (Last accessed: 27.03.2021).
6. Coelho, L.P. 2013. Mahotas: Open source software for scriptable computer vision. Journal of Open Research Software. DOI: http://dx.doi.org/10.5334/jors.ac (Last accessed: 27.03.2021).

РУЖАНСЬКА Анастасія В'ячеславівна – студентка кафедри штучного інтелекту та програмного забезпечення факультету комп'ютерних наук Харківського національного університету імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: asynasty1999@gmail.com; ORCID: 0000-0002-2683-6768.

Наукові інтереси:

- *методи машинного навчання;*
- *системи комп'ютерного зору;*
- *штучний інтелект.*

ВАСИЛЬЄВА Лариса Валентинівна – к.б.н., доцент кафедри електроніки та управляючих систем; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: lvvasilieva@karazin.ua.

Наукові інтереси:

- *штучний інтелект.*

УДК 004.032.26 **СЕКЦІЯ 7**

РУЗУДЖЕНК С.Р.

РОЗРОБКА СТРУКТУРНОЇ СХЕМИ СИСТЕМИ ВЗАЄМОДІЮЧИХ ТЕХНОЛОГІЙ ДЛЯ НАВЧАННЯ НЕЙРОННИХ МЕРЕЖ

Вступ

Штучна нейронна мережа (НМ) представляє собою пристрій паралельних обчислень, який складається з безлічі взаємодіючих процесорів [1]. Такі процесори зазвичай є дуже простими, на відміну від тих, що використовуються у комп'ютерах. Кожен з них працює лише з сигналами, які він отримує та надсилає з певною періодичністю. Проте, при об'єднанні таких локально простих елементів у достатньо велику мережу з керованою взаємодією, з'являється можливість для вирішення широко спектру доволі складних задач.

Найважливішою властивістю НМ є здатність навчатися на основі обробки даних оточуючого середовища і в результаті навчання підвищувати рівень продуктивності системи. Підвищення продуктивності відбувається з часом, відповідно до певних правил. Навчання НМ відбувається шляхом інтерактивного процесу коректування синаптичних ваг та порогових значень. У найкращому випадку НМ отримує нові знання про оточуюче середовище на кожній ітерації процесу навчання.

У ході даної роботи було проведено дослідження технологій, що застосовуються для навчання нейронних мереж без учителя, особливості та методи навчання нейронних мереж для вирішення задач NLP (англ. Natural Language Processing). Метою роботи була розробка структурної схеми взаємодіючих технологій для навчання нейронних мереж на прикладі вирішення двох базових задач обробки природної мови – аналіз тексту (англ. Natural Language Understanding, NLU) та генерація тексту (англ. Natural Language Generation, NLG).

Архітектура нейронних мереж

У задачах обробки природної мови, завдяки можливості використання внутрішньої пам'яті для обробки послідовностей довільної довжини, найбільшого поширення набули рекурентні нейронні мережі (англ. Recurrent neural network, RNN). Рекурентні нейронні мережі – тип нейронних мереж, де зв'язки між елементами утворюють спрямовану послідовність [2]. Така властивість дозволяє обробляти серії подій з часом або послідовні просторові ланцюжки. З багатьох архітектурних рішень для рекурентних мереж найбільш використовуваними є мережі довгої короткострокової пам'яті(англ. Long Short Term Memory, LSTM) та керований рекурентний блок (англ. Gated Recurrent Units, GRU). Мережі довгої короткострокової пам'яті – особливий вид РНС, здатних до навчання довгостроковим залежностям. LSTM спеціально спроектовані таким чином, щоб уникнути проблеми довгострокових залежностей (рис.1).

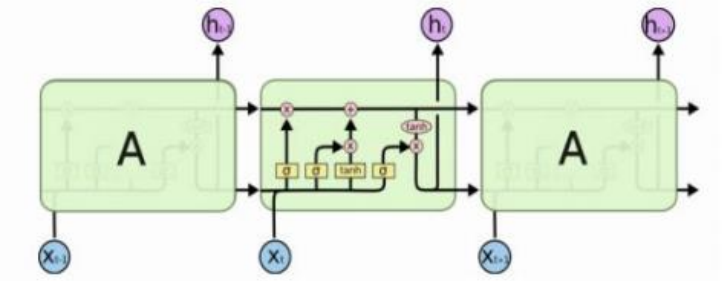


Рис. 1 – Архітектура мережі довгої короткострокової пам'яті

Кожна лінія передає цілий вектор від виходу одного вузла до входів інших. Малі кола представляють точкові оператори, такі як складання векторів, в той час як малі прямокутники

© РУЗУДЖЕНК С.Р., 2021

являють собою навчені шари нейронної мережі. Суцільні лінії позначають конкатенацію, в той час як розгалужені лінії позначають, що їх вміст копіюється, і копії відправляються в різні місця. Мережі типу керований рекурентний блок (рис. 2) об'єднують шлюзи LSTM для створення більш простого правила оновлення даних, що дозволяє знизити складність системи та підвищити її ефективність.

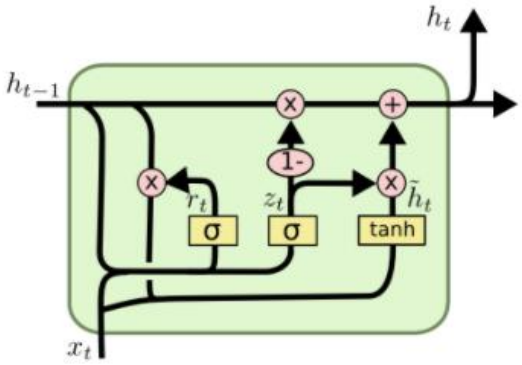


Рис. 2 – Архітектура керованого рекурентного блоку

Порівнюючи дані архітектури важко сказати, яка є більш ефективною, тому вибір архітектури мотивується конкретною задачею, для якої використовують мережу. У загальному випадку, при інших рівних параметрах, використовують керований рекурентний блок, адже він потребує меншої кількості параметрів.

Методи обробки природної мови

Природна мова – це мова, яка еволюціонувала природним шляхом у процесі повсякденного використання людьми з плином часу, без формальної побудови. Вона охоплює широкий набір, що включає розмовні та жестові мови. Природна мова має неоднозначний характер, як у письмовій так і в усній формі. Через цю неоднозначність комп'ютерам складно обробляти і розуміти природну мову. Ця складність особливо поширюється на такі аспекти мови як емоційне забарвлення, а також використання різноманітних тропів. У будь-якій мові існує двозначність в семантиці, граматичній структурі слова та речення, тому використовують різні методи обробки природної мови, які дозволяють вирішити кожну з цих неоднозначностей.

Обробка природної мови (NLP) – термін, що відноситься до різних способів обчислювальної обробки людських мов. Виділяють декілька видів задач класифікації, типових для NLP [3]: визначення початку та кінця слів за символами; виявлення початку та кінця речення за словами; виявлення початку та кінця словосполучень за словами у реченні; виявлення належності слова до певної частини мови; виявлення початку та кінця іменованих сутностей за словами в реченні; виявлення емоційного забарвлення за словами в реченні.

При аналізі природної мови, мовні характеристики часто групуються в набір категорій (рис. 3). Для аналізу тексту цими категоріями є морфологія, лексика, синтаксис, семантика, дискурс і прагматика. Морфологія відноситься до форми і внутрішню структуру слова. Лексичний аналіз передбачає сегментацію тексту на значущі одиниці, такі як слова. Синтаксис відноситься до правил і принципів, що застосовуються до слів, фраз і речень. Семантика – до контексту, який надає сенс реченню. Саме семантика забезпечує ефективність людської мови. Аналіз дискурс застосовується до розмов і відносинам, що існують між реченнями. Прагматика – до зовнішніх характеристик, таких як намір передати контекст.

Для аналізу мови характеристики зазвичай групуються за категоріями акустики, фонетики, фонеміки і просодики. Акустика – це методи, які ми використовуємо для подання звуків. Фонетика аналізує, як звуки зіставляються з фонемами, які служать базовими одиницями мови. Фонеми, також відомі як фонологія, відносяться до того, як фонеми використовуються в мові. Просодика аналізує немовні характеристики, які супроводжують мову, таким як тон, наголос, інтонація і висота звуку. Обчислювальні та статистичні методи, які можуть бути

застосовані до кожного компоненту синхронної моделі, складають основу обробки природної мови.

Рис. 3 – Етапи обробки мови за категоріями

Обробка природної мови прагне зіставити мову з представленнями, які фіксують морфологічні, лексичні, синтаксичні, семантичні або дискурсивні характеристики, які потім можуть бути оброблені методами навчання. Вибір подання може мати значний вплив на подальші завдання і буде залежати від обраного алгоритму навчання для аналізу.

Розробка схеми взаємодіючих технологій

На основі досліджень методів аналізу та генерації тексту було розроблено структурну схему взаємодіючих технологій, що демонструє послідовність етапів для навчання нейронної мережі, що використовується для вирішення задачі обробки та генерації тексту (рис. 4). За поданою схемою аналіз та генерація тексту розподіляється на такі етапи: попередня обробка наданого корпусу (розміченої бази даних); представлення текстових даних у векторній формі, на основі якого формується словник для подальшої роботи; безпосередня генерація тексту.

У ході попередньої обробки, корпус проходить декілька етапів аналізу:

1. Морфологічний аналіз. Два найбільш поширених підходи для вирішення даної проблеми – це стемінг і лематизація.
2. Лексичний аналіз. Для аналізу використовуються токенізація та видалення стоп-слів.
3. Синтаксичний аналіз. Обчислювальна лінгвістика використовує різні підходи для вилучення цих контекстних підказок, таких як теги частини мови, розбиття на частини та синтаксичний аналіз залежностей.
4. Семантичний аналіз. Семантичний аналіз зацікавлений в розумінні значення, перш за все, в термінах відносин між словами і реченнями, за допомогою таких технологій як розпізнавання іменованих сутностей, розпізнавання відносин та подій, аналіз семантичних ролей.
5. Класифікація та кластеризація тексту, які проводяться за допомогою аналізу тональності, лексичних ланцюгів та тематичного моделювання.

На подальших етапах відбувається формування векторного представлення текстових даних. Для цього застосовують такі технології: кодування one-hot encoding, пошук N-грам, формування мішка слів (Bag-of-Words), кодування за допомогою технології Word2Vec. Після завершення даного етапу з використанням мовних моделей формується предикат-твердження (Claim Synthesis), що містить головний сенс. Також може використовуватись функція формування множини нових тверджень (Predicate Recycling). На наступних етапах формується пара предикат-заперечення (Automatic Claim Negation), що надалі обробляються разом. У

створюваному тексті знаходяться факти на підтвердження заперечення або твердження. А на останньому етапі відбувається оцінка знайдених фактів.

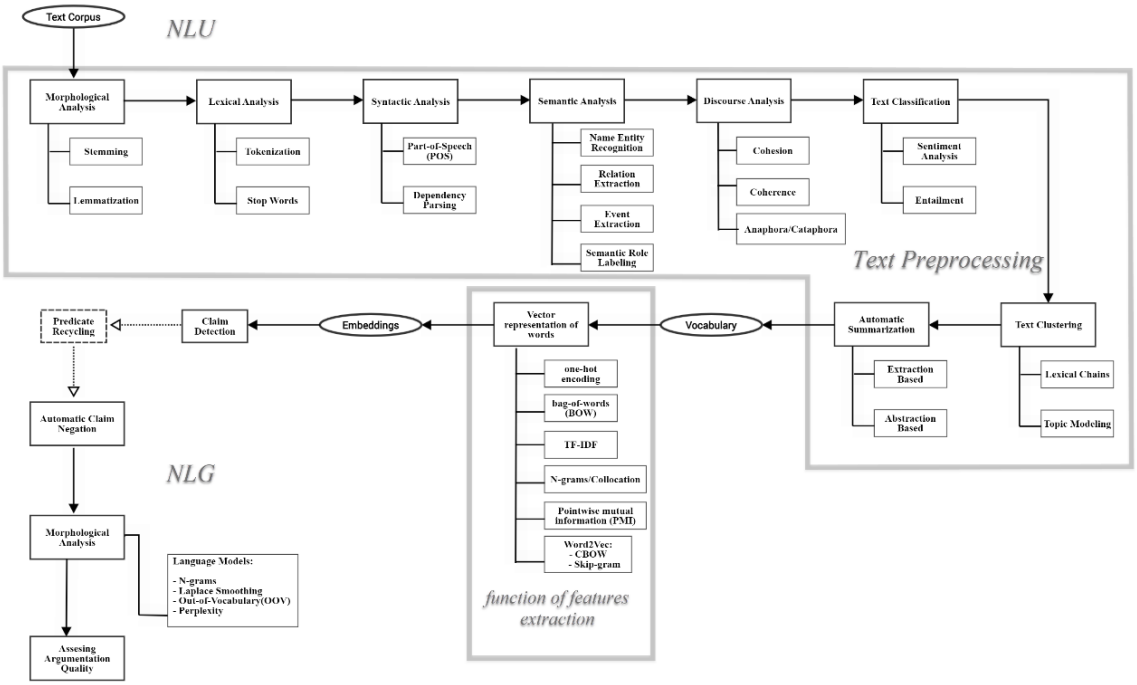


Рис. 4 – Схема взаємодіючих технологій для навчання нейронної мережі

Висновки

У роботі проаналізовано технології для навчання нейронних мереж без учителя, що дозволяють вирішувати задачі аналізу та генерації тексту. Запропонована схема дозволяє зрозуміти принципи навчання мережі для роботи з NLP та демонструє основні технології, що для цього застосовуються.

ЛІТЕРАТУРА

1. Каллан, Р. Основные концепции нейронных сетей. / пер. с англ. /Р. Каллан. – М. : Вильямс, 2001. – 287 с.
2. Реккурентные нейронные сети [Электронный ресурс]: https://habr.com/ru/company/wunderfund/blog/331310/ (Дата звернення 25.01.2021).
3. Нейросетевые методы в обработке естественного языка / пер. с анг. А. А. Слинкина. – М.: ДМК Пресс, 2019. – 282 с.

РУЗУДЖЕНК Сабіна Ровшанівна – студентка кафедри штучного інтелекту та програмного забезпечення факультету комп'ютерних наук Харківського національного університету імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: ruzudzhenk.jb@gmail.com; ORCID: 0000-0002-1749-5762.

Наукові інтереси: *штучний інтелект; навчання нейронних мереж; технології обробки природної мови*.

УДК 004.056.55 **СЕКЦІЯ 5**

СЄМЄНЯКІНА В.С., ЄСІНА М.В.

АТАКИ НА ЕЛЕКТРОННИЙ ПІДПИС DILITHIUM

Вступ

У даній роботі представлено атаки на електронний підпис (ЕП) Dilithium [1]. Було визначено кілька вразливих операцій і описуються конкретні сценарії помилок, які можуть вплинути на відновлення ключа. Для кожного сценарію представлено алгоритм використання та, як результат, ймовірність успішного виконання. Варто також додати, що потрібно припускати можливість впровадження однієї випадкової помилки: пропуск інструкцій (команд), арифметичні дефекти, неполадки при зберіганні і багато іншого. Несправності можуть бути використані протягом значного проміжку часу роботи і не обмежуються конкретними операціями. Для більш успішної атаки, варто мати більше контролю над решітчастим підписом, як приклад, мати можливість переривати замкнутий цикл.

1 Помилки детермінованих решітчастих підписів

Нехай повідомлення *M* буде підписано двічі: під час першого виклику – помилка не вноситься, на виході отримується дійсний і правильний підпис $\sigma = (z,\ h,\ c)$ (тут і надалі будуть використовуватись алгоритми ЕП Dilithium з посиланням на [1]), під час другого – помилка буде внесена; використовується «'», наприклад, z' – для позначення змінних у помилковому (другому) виклику. Тобто, помилка вводиться так, щоб y' не порушувався через детермінізм, який дорівнює y, але $c' \neq c$ і, отже, $z' = s_1 \cdot c' + y$. Генерація компонента підпису:

$$z = s_1 \cdot c + y, \tag{1}$$

де s_1 – секретний елемент; y – поліном, що генерується з дискретного розподілу Гауса.

Таким чином, помилка викликає сценарій відмови від повторного використання. Застосування алгоритму Fiat-Shamir з перериванням вимагає, щоб обчислення як дійсного, так і помилкового підпису закінчувалося однаковою ітерацією циклу переривання. Використовуючи k_f, для позначення кінцевого значення лічильника циклу k, повинна виконуватись умова $\Delta k_f = k_f - k_f' = 0$ [2].

1.1 Випадкова помилка при зверненні до геш-функції (fH)

Кращий спосіб викликати повторне використання – ввести випадкову помилку у процес обчислення $c \in B_{60} := \mathrm{H}(\mu \square \mathrm{w}_1)$. Щоб цього досягти, потрібно маніпулювати одним із вхідних параметрів μ, w_1 перед їх використанням у H або ввести помилку в саму геш-функцію. Припускається, що зловмисник може додати похибку в правильну (останню) ітерацію k_f. Якщо крок відхилення проходить з різним c', s_1 може бути відновлено [2]. Існує два способи перевірити, що $\Delta k_f = 0$, тобто визначити успіх відновлення ключів. Перший метод – відновити s_1 і перевірити $s_1 \in S_\eta^l$. Другий метод – використати малу норму $c \cdot s$, обчислюючи $\|\Delta z\|_2$ ($\Delta \mathrm{z} = \mathrm{z} - \mathrm{z}'$) і перевірити, чи вона не нижче певного порогу. Разом з виконання умови $\Delta k_f = 0$, вимагається, щоб Δc ($\Delta c = c - c'$) був зворотним (що виконуються з високою ймовірністю). Атака не вдається лише за умови $-\Delta c$ не є зворотним, що виявляється при перевірці на зворотність. Як результат, дуже високий відсоток проведення успішної атаки 90% (табл. 1).

1.2 Випадкова помилка у множенні поліномів (fW)

Керуючи обчисленням вхідних даних μ, w_1 у геш-функції H, можна змінити значення $c := \mathrm{H}(\mu, w_1)$. Значення μ також використовується як seed, для функції DeterministicSample [1],

© СЄМЄНЯКІНА В.С., ЄСІНА М.В., 2021

отже внесення похибки в обчислення $\mu := \mathrm{CRH}(tr \square M)$ не може бути використано. Тому внесення похибки відбувається у $w := \mathrm{A}_y$, що призводить до хибного w_1.

За допомогою методу NTT ефективно реалізуються необхідні множення поліномів у R_q. Множення відбувається довше, ніж гешування, тому це більш важлива ціль для атаки. Кількість коефіцієнтів у w, на які впливає одна випадкова помилка, може становити від 1 до $n \cdot k$. Незмінені коефіцієнти w добре проходять етап відхилення, а одного зміненого достатньо для досягнення $\Delta c \neq 0$. Зменшення кількості коефіцієнтів збільшує ймовірність успіху помилки. Таким чином, конкретна позиція помилки має вагомий вплив. Внесення несправності до попереднього NTT в y, який порушує $n \cdot k$ коефіцієнтів w, призводить до низької ймовірності успіху – 26% (табл. 1). Внесення похибки у зворотний NTT, який порушує лише два останні коефіцієнти w, має ймовірність успіху >90% [2].

1.3 Випадкова помилка у розширенні матриці ($\mathrm{fA_E}$) та деформований ρ при отриманні sk ($\mathrm{fA_\rho}$)

Інша можливість досягти помилкового $w = \mathrm{A}_y$ – виконати розширення початкового значення ρ (seed) до матриці A, що в алгоритмі підпису [1] робиться перед входом у цикл переривання. Використовуючи функцію ExpandA [1], можна підвищити вірогідність введення похибок для цього сценарію. Крім того, матриця A зберігається у пам'яті протягом тривалого часу, тому гешуючи значення A, не потрібно повторно викликати функцію ExpandA, для кожної операції підпису. Ще один спосіб досягти змін у матриці A – провести маніпуляції з початковим значенням ρ, наприклад, під час додавання особистого ключа (сценарій fA_ρ) або вносячи похибку у розширення $\mathrm{A} \in R_q^{k \times l} := \mathrm{ExpandA}(\rho)$ (сценарій $\mathrm{fA_E}$). Метод розширення ρ до A (функція ExpandA, що має поліноми $k \cdot l$, які містять A) створюється за допомогою незалежних викликів функції SHAKE [1]. Таким чином, будь-яка помилка перестановки SHAKE призводить лише до одного неправильного полінома. Ймовірності успіху такої помилки приблизно 54%. Внесення похибки у ρ, призводить до зовсім іншого значення матриці A, що змінить значення w. Через це, ймовірність успіху зменшується до 14% (табл. 1). Проте, цей тип помилок має головну перевагу, при успішному виконанні контрзаходів – не може бути виявлений загальним контрзаходом з подвійним обчисленням [2].

1.4 Випадкова помилка у вибірці (fY)

Для поточного сценарію, виконується частково-повторне використання nonce, що призводить до дійсних підписів, тому перевірка підпису не зможе виявити несправність. Для підписів на основі решіток, ця атака є серйозною загрозою. Вона націлена на генерацію випадкового елемента y, що використовується для (1). Ідея атаки полягає в тому, щоб ввести помилку у випадкову вибірку для отримання аномально малого полінома. Порушення генерації y призводить до витоку секретного ключа [3]. Використовуючи цю несправність, отримуються підписи, в яких y – поліном з малим ступенем. Якщо ступінь досить мала, це призводить до розкриття всього секретного ключа. Реалізація атаки: nonce $\mathrm{y} \in S_{\gamma_1 - 1}^l$ породжується функцією DeterministicSample [1]. Вхідне початкове значення s змінюється щоразу, коли підпис відхиляється, а лічильник u змінює окремі елементи y. Функція SHAKE, яка складається з двох фаз: поглинання і стискання, підроблюється таким чином, щоб змінювалось лише кілька коефіцієнтів y_u. Якщо помилка буде введена під час фази поглинання, вихідне значення SHAKE буде відрізнятись. Несправність введена ближче до кінця фази стискання, порушить тільки останні значення f [2]. Якщо помилка внесена правильно і $\Delta k_f = 0$, то ключ s_1 завжди можна відновити. Ймовірність того, що $\Delta k_f = 0$ становить 24.5% (табл. 1). Одного несправного підпису досить, для відновлення всього секретного ключа за допомогою цієї атаки, але іноді може знадобитися кілька повторних спроб.

2 Екзистенційно-фальсифікована атака на Dilithium

Екзистенційно-фальсифікована атака [4] на Dilithium складається з двох частин. Перша частина включає в себе відновлення основного складового секретного ключа – s_1, за допомогою атаки на помилку при додаванні пропуску. Для завершення атаки використовується процедура фальсифікації підписів, яка створює дійсний підпис лише за первинним ключем, незважаючи на те, що секретний ключ Dilithium sk складається з декількох компонентів.

2.1 Алгоритм атаки

Припускається, що зловмисник (далі ZL) має повний реальний доступ до пристрою жертви під час виконання процедури підписання. ZL може багаторазово запускати пристрій, для генерації дійсних підписів, для будь-якого обраного ним повідомлення, а також мати доступ до підписів, що генеруються. Він повинен мати можливість активно втручатись у процес підписання, за допомогою введення даних про несправності та пасивного спостереження за бічними каналами. Основною моделлю збою для атаки є пропуск команд.

Головним чином, атака працює над операцією генерації (1) (вона буде позначатись z_{gen}). Вона використовує детерміновану властивість Dilithium і працює шляхом введення одиничної несправності. Для цієї атаки припускається, що мішень формує P підписів та визначає незалежні многочлени у модулі z. Компонент підпису z генерується наступним чином: $z[i][j] = s_1 \times c[i] + y[i][j]$, при $i \in \{1,\ldots,\mathrm{P}\}$, при $j \in \{0,\ldots,l-1\}$.

Аналіз проводиться над (1), його окремі компоненти розглядаються як поліноми в R_q, бо поліноми обробляються незалежно один від одного. ZL здійснює операцію складання, використовуючи окремі помилки та об'єднує інформацію з безлічі помилкових підписів, отриманих за різних значень коефіцієнтів, для відновлення первинного секрету s_1 [4].

2.2 Перша складова – відновлення первинного ключа s_1

Компонент z підпису генерується наступним чином:

$$\begin{aligned} z &= s_1 \cdot c \\ z &= z + y \end{aligned}. \tag{2}$$

Знаючи один з коефіцієнтів $(s_1, c)_t$, ZL може побудувати модульне лінійне рівняння з n невідомими, при цьому невідомими є коефіцієнти $(s)_t$ первинного секрету:

$$(\hat{z})_t = \langle s_1, \mathrm{Rot}(c,t) \rangle, \tag{3}$$

де $\mathrm{Rot}(c,t)$ – вектор коефіцієнта обертання c, повернений на t раз в антициклічному режимі.

Випадково ZL може визначити різницю між числом ітерацій закінченої і незакінченої процедури підписання, перевіривши рівність вихідних поліномів виклику c у складі справних і несправних підписів. Мета ZL – сформувати n пар P_i для $i \in \{0,\ldots,n-1\}$, розклавши на декілька різних коефіцієнтів. Вирішивши цю проблему, можна дізнатись первинний ключ. Тому, кількість підписів необхідних для відновлення первинного секрету дорівнює n.

Кількість несправних підписів дорівнює кількості відновлюваних коефіцієнтів. ZL доводиться вводити одиничні помилки повторюючи їх, протягом декількох проходів процедури підпису, щоб отримати достатньо інформації, для відновлення первинного секрету Dilithium.

2.3 Друга складова – реалізація екзистенційної підробки

Секретний ключ sk складається з декількох компонентів $(\rho, K, tr, s_1, s_2, t_0)$, але знання первинного ключа s_1 достатньо для проведення екзистенціальної атаки. Знаючи s_1 та секрет $K \in \{0,1\}^{256}$, ZL може створити дійсний підпис нового повідомлення. K застосовується для генерації тимчасового y в процедурі підпису, так як верифікатор не перевіряє правильність y. ZL рівномірно випадковим чином вибирає y з $S^l_{\gamma_1 - 1}$ та обчислює z за допомогою відновленого s_1. Для перевірки дійсності підпису вимагається знання w_1, що може бути обчислено

$$\mathrm{UH}_q\left(\mathrm{MH}_q\left(u,r,2\gamma_2\right),r,2\gamma_2\right)=\mathrm{HB}_q\left(u+r,2\gamma_2\right)=w_1, \quad (4),$$

де HB_q – процедура HighBits $\mathrm{HB}_q\left(r,\alpha\right)$, з алгоритму округлення Dilithium; MB_q – процедура MakeHint $\mathrm{MH}_q\left(u,r,\alpha\right)$; UH_q – процедура UseHint $\mathrm{UH}_q\left(h,r,\alpha\right)$ демонструє використання біта h, як підказки для отримання HB_q.

ZL не може виконати додаткову перевірку w_1, яку може зробити дійсний підписант. Таким чином, для фальшивої схеми є нормою виробляти деякі недійсні підписи з невеликою ймовірністю, яка може бути проігнорована [4].

Табл.1 Ймовірність успіху атак на помилки, у відсотках

Сценарій помилки	fH	fW	fA_E	fA_ρ	fY
Ймовірність успіху	91%	26-90%	55%	14%	25%

Висновки

За результатами, представленими у табл. 1, можна підвести підсумок розглянутих атак. Несправність при розширенні матриці A, у результаті дає складну і часозатратну задачу, тому атака fA_ρ має найменшу ймовірність успіху. Для сценаріїв fY, fH та fW припускалось, що помилка вводиться під час останньої ітерації k_f. У сценарії fW було проаналізовано найкращі та найгірші можливі результати, тому шанс на успіх становить від 26% до 90% відповідно. Сценарій fH легше всього реалізувати і він має найвищу ймовірность успіху. fY призводить до дійсних підписів і може використовуватись порушником. Також, вводячи тільки одиничний дефект у процедуру підпису, можна відновити ключі і виконати екзистенційну атаку.

ЛІТЕРАТУРА

1. Dilithium / B. Shi, D. Léo, K. Eike. 2021. URL: https://pq-crystals.org/dilithium/data/dilithium-specification-round3-20210208.pdf. (дата звернення: 17.03.2021).
2. Espitau T., Pierre-Alain Fouque, Benoît Gérard Loop-Abort Faults on Lattice-Based Signatures and Key Exchange Protocols. *IEEE Transactions on Computers PP(99aa):1-1*. 2018. P. 1535 - 1549
3. Groot Bruinderink, L., Pessl, P. Differential Fault Attacks on Deterministic Lattice Signatures. *IACR Transactions on Cryptographic Hardware and Embedded Systems*. 2018(3). P. 21-43.
4. Prasanna R., Mahabir Prasad Jhanwar, James H. Exploiting Determinism in Lattice-based Signatures: Practical Fault Attacks on pqm4 Implementations of NIST Candidates. *Asia CCS'19: Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*. 2019. P. 427–440

СЄМЄНЯКІНА Владислава Сергіївна – студентка кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: xa11867828@student.karazin.ua; vladasemenyakina@ukr.net; ORCID: 0000-0002-0310-1835.

Наукові інтереси:

– *постквантові алгоритми електронного підпису.*

ЄСІНА Марина Віталіївна – к. т. н., доцент кафедри безпеки інформаційних систем і технологій; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: m.v.yesina@karazin.ua; ORCID: 0000-0002-1252-7606.

Наукові інтереси:

– *захист інформації, постквантова криптографія, криптографія на алгебраїчних решітках.*

УДК 504.3 **СЕКЦІЯ 1**

СЄРІКОВА О.М., СТРЕЛЬНІКОВА О.О.

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ПАРАМЕТРІВ РУКАВНИХ ФІЛЬТРІВ

Забруднення навколишнього середовища, зокрема атмосфери, відбувається переважно за рахунок впливу промислових підприємств. Недосконаліть технологічних процесів і устаткування діючих підприємств призводить до екологічно небезпечного підвищення концентрацій шкідливих речовин в атмосферному повітрі. Проблема зменшення негативного техногенного впливу на навколишнє середовище є актуальною на сьогодні [1-2]. В роботі було проведено математичне моделювання параметрів рукавних фільтрів з метою підвищення рівня екологічної безпеки промислових територій, на яких здійснюються викиди забруднюючих речовин в атмосферне повітря

У результаті обробки експериментальних даних в роботі отримано рівняння для визначення ефективності пилоочищення, η, % [3].

$$\eta = 13.172 w^{0.772} m^{0.34} \quad (1)$$

В цьому рівнянні w - швидкість газу, m- щільність зрошення. Побудуємо залежність ефективності пилоочищення від параметрів w та m

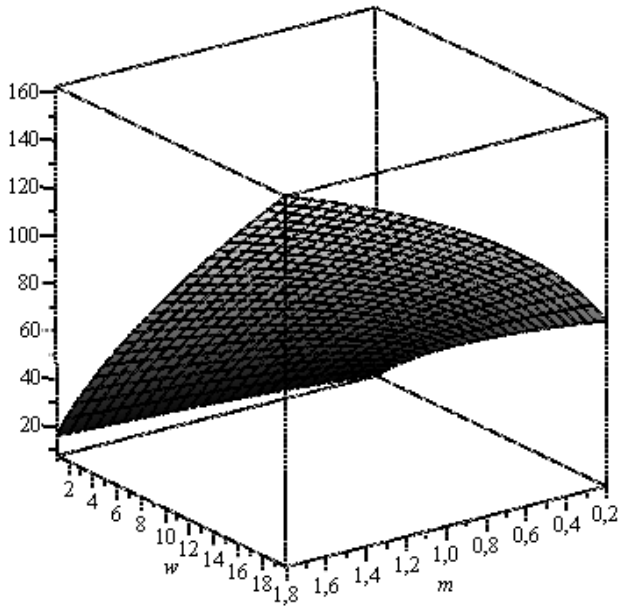


Рисунок 1 – Залежність ефективності пилоочищення від швидкості газу та щільності зрошення

На підставі цих даних можна зробити висновок відносно області оптимальних режимних параметрів здійснення процесу пилоочищення, а саме швидкість газу має бути від 12 до 20м/сек, а щільність зрошення наближатися до 1.5-1.8 дм3 /м3

Таким чином, прийняте в розрахунках значення швидкості газу знаходиться в межах оптимальних режимних параметрів. Для цього значення побудуємо залежність ефективності пилоочищення від щільності зрошення при w=15 м/сек [3].

© СЄРІКОВА О.М., СТРЕЛЬНІКОВА О.О., 2021

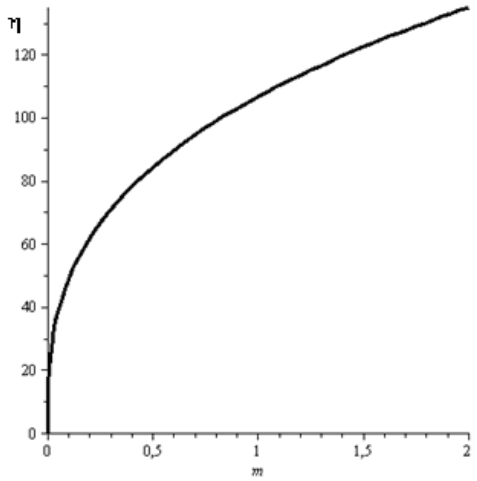


Рисунок 2 – Залежність ефективності пилоочищення від щільності зрошення при w *=15 м/сек.*

Далі з'ясуємо залежність величині

$$\tau_{\Phi} = 1000 \cdot \frac{\frac{\Delta P_{\Phi}}{w_{\Phi} \cdot \mu} - A}{B \cdot w_{\Phi} \cdot c_{\Phi}}, \qquad (2)$$

яка характеризує мінімальну тривалість періоду фільтрування між двома регенераціями при отриманих вище значеннях параметрів швидкості газу,щільності зрошення та параметрів ΔP_{Φ} =15000, μ= $23.3 * 10^{-6}$; A= 261978342.05; B= 9964590677.8; в залежності від w_{Φ}, c_{Φ} .

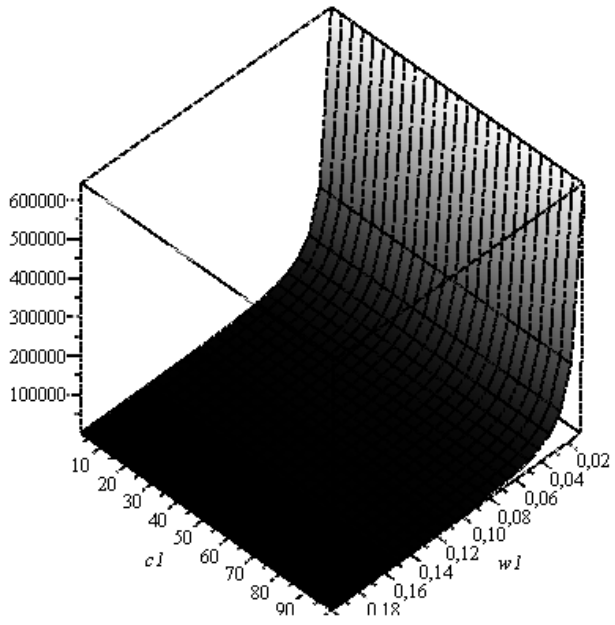


Рисунок 3 – Мінімальна тривалість періоду фільтрування

З отриманих даних бачимо, що мінімальна тривалість періоду фільтрування практично не змінюється , якщо w_{Φ} >0.08 та практично не залежить від c_{Φ} .[3]

Нижче на рисунку 3.4а подано ту ж саме залежність, але при 0.01< w_{Φ} <0.07 та на рисунку 3.4б при 0.05< w_{Φ} <0.06

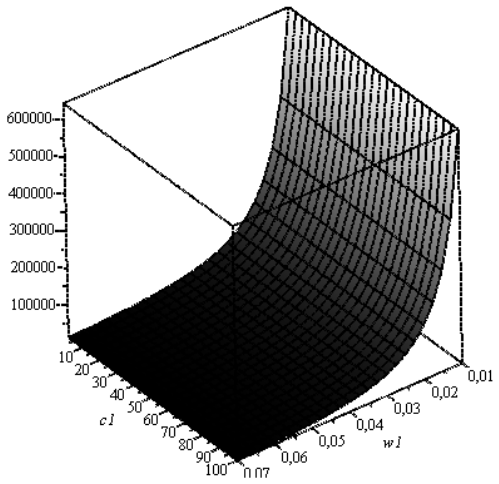


Рисунок 4а - Мінімальна тривалість періоду фільтрування при 0.01< $w_Ф$ <0.07

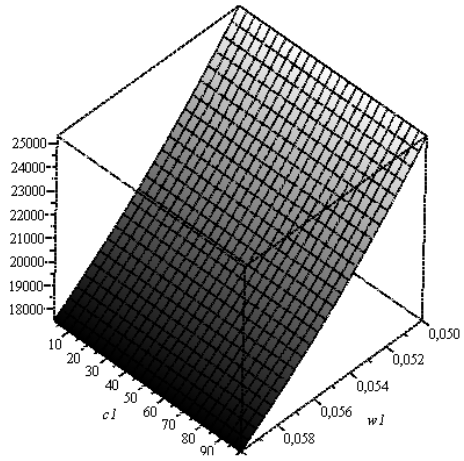


Рисунок 4б - Мінімальна тривалість періоду фільтрування при 0.05< $w_Ф$ <0.06

Зазначимо, що отримане в результаті розрахунків значення для $\tau_Ф$ попадає в область найменших значень для мінімальної тривалості періоду фільтрування. Тобто отримане значення для швидкості фільтрації $w_Ф$ =0,0583 м/с є оптимальним.

ЛІТЕРАТУРА

1. Сєрікова, О. М. Прогнозування і управління рівнем ґрунтових вод для підвищення екологічної безпеки забудованих територій України: дисертація канд. техн. наук, спец.: 21.06.01 – екологічна безпека / О. М. Сєрікова; наук. кер. В. В. Яковлєв. Х.: Харківський нац. ун-т міськ. госп-ва ім. О. М. Бекетова, 2019. 166 с.
2. Сєрікова О. М. Підвищення рівня екологічної безпеки забудованих територій України, схильних до підтоплення [Текст] : монографія / О. М. Сєрікова, О. О. Стрельнікова, В. Ю. Колосков – Х.: ФОП Бровін О.В., 2020. – 142 с.
3. Гурець Л. Л. Науково-методологічні основи екологічної безпеки при забрудненні атмосферного повітря газопиловими викидами промислових підприємств : дис. ... д-ра техн. наук : спец. 21.06.01 "Екологічна безпека" : Дата захисту 20.04.17 / Л. Л. Гурець. - Суми, 2017. - 313 с.
4. Pelyk L. The use of technical textiles for industrial filters. Bulletin KNUTD. 2008. № 5 (43). P. 258 - 261.

СЄРІКОВА Олена Миколаївна – к. т. н., старший викладач кафедри прикладної механіки та технологій захисту навколишнього середовища; Національний університет цивільного захисту України, вул. Чернишевська, 94 м. Харків, 61023; e-mail: elena.kharkov13@gmail.com; ORCID: 0000-0003-0354-9720.

Наукові інтереси:

- *ґрунтові води;*
- *підтоплення*
- *системи управління екологічною безпекою*
- *техногенно-екологічна безпека*

СТРЕЛЬНІКОВА Олена Олександрівна – д. т. н., професор, провідний науковий співробітник зі спеціальності механіка деформівного твердого тіла; *Інститут проблем машинобудування ім. А. М. Підгорного* НАН України вул. Пожарського, 2/10, м. Харків, 61046; e-mail: elena15@gmx.com; ORCID: 0000-0003-0707-7214.

Наукові інтереси:

– теорія і методи розв'язання сингулярних інтегральних рівнянь;
– динамічні задачі гідропружності елементів машинобудівних конструкцій;
– теорія тріщин в пружних і пружно-пластичних тілах.

УДК 681.518 **СЕКЦІЯ 7**

СТЄРВОЄДОВ М.Г.,АРТЮХ С.О.

ЦІЛІ ТА ЗАВДАННЯ СТВОРЕННЯ ІНФОРМАЦІЙНО-УПРАВЛЯЮЧИХ СИСТЕМ ДЛЯ ПРИСКОРЮВАЧІВ ЗАРЯДЖЕНИХ ЧАСТОК

Актуальність теми. Прискорювачі елементарних часток є основним експериментальним обладнанням сучасного наукового пізнання і унікальними, часто безальтернативними, інструментами для виконання технологічних операцій, медичних досліджень і терапії. На жаль, їх відрізняє висока вартість і швидке фізичне та моральне старіння. Особливо це відноситься для систем збору і аналізу даних, інформаційних систем та систем управління виконавчими пристроями.

Із розвитком інформаційних технологій та мікроелектроніки потреба оновлювати середовище систем збору і аналізу даних виникає раз у середньому в 5-7 років. Застосування новітніх схемотехнічних компонентів підвищує функціональність контролерів, які застосовуються, спрошує процес розробки, який при відносно малих ресурсах приводе до значного поліпшення систем керування. Із ряду економічних причин провідні виробники мікроелектроніки припиняють серійний випуск застарілих виробів, тому обслуговування застарілого обладнання стає проблематичним, що також обумовлює необхідність розробки нових підходів до інформаційно-вимірювальних і управляючих систем.

Зростання потужності обчислювальних мереж тягне за собою зміни підходів рішення проблеми управління і телекомунікації. Розвиток протоколів верхнього рівня управління висуває певні вимоги до програмного забезпечення польового рівня. Поява цілого сімейства комунікаційних стандартів, так званих польових шин PROFIBUS, CAN, інтерфейсу Profinet, EtherCat змінило концепції побудови систем керування. Система керування, яка побудована за допомоги цих шин, може являти собою сукупність інтелектуальних датчиків, об'єднаних послідовною перешкодостійкою магістраллю. Датчики територіально прив'язані кожен до свого об'єкту управління. Процес керування усією системою датчиків і актуаторів відбувається від центрального диспетчерського пункту.

Таким чином, актуальність розробки або реінжинірингу існуючих систем керування радіаційно-фізичними комплексами, до яких виноситься прискорювачі заряджених часток, викликана необхідністю вирішення наступних завдань:

- реінжиніринг морально і фізично застарілих систем керування прискорювачами;
- оновлення елементної та мікроконтролерної апаратної бази на більш сучасну;
- інтеграції технології Ethernet у якості головної магістралі обміну даними;
- використання новітнього спеціалізованого ПЗ для систем збору даних і управління;
- застосування бездротових систем комунікації, хмарних сервісів та туманних технологій.

Мета роботи та галузь застосування результатів. Основною метою роботи є створення сучасної архітектури системи керування прискорювача заряджених часток, комп'ютерних моделей і алгоритмів функціонування апаратно-програмних компонентів системи автоматизації, яка дозволяє здійснювати децентралізований віддалений контроль і моніторинг параметрів пристроїв установки та пересилати їм команди управління з активним використанням хмарних сервісів та створення цифрових двійників датчиків і актуаторних схем для емалювання роботи підсистем керування із задачею оптимізації управління вузлами прискорювача.

Для досягнення мети необхідно вирішити задачі, до яких належать:

1. Розроблення архітектури та моделі автоматизованої системи керування прискорювачем із можливістю дистанційного контролю, діагностики та моніторингу параметрів установки.
2. Розроблення комп'ютерних моделей інтелектуальних датчиків параметрів пучків та контролерів для емуляції роботи електронного обладнання об'єкта керування.
3. Розроблення моделі і реалізація комп'ютерного пульта керування прискорювачем.

© СТЄРВОЄДОВ М.Г.,АРТЮХ С.О., 2021

4. Розроблення апаратно-програмних модулів для дистанційного керування.
5. Розробка діючих лабораторних макетів інтелектуальних датчиків і актуаторів.

Об'єкт дослідження – системи управління електрофізичних установок.

Предметом дослідження - архітектура й апаратно-програмні компоненти автоматизованої системи управління прискорювачем заряджених часток.

Методи досліджень. Методи дослідження включають в себе методи проектування інформаційно-керуючих систем та побудови графічних людино-машинних інтерфейсів, методи комп'ютерного моделювання електронних схем і систем, методи емуляції апаратного забезпечення на базі QEMU, методи лабораторних експериментів.

Наукова новизна потенційного наукового результату дослідження:

У рамках запропонованого напрямку досліджень, плануються отримати наступні наукові результати.

Вперше розробити:

- архітектуру інтелектуальної інформаційно-вимірювальної і керуючої системи малим прискорювачем заряджених часток та структурні схеми її SMART- компонентів на основі концепції кіберфізичних систем та із застосуванням технології хмарних обчислень 2-го покоління;
- цифрові копії інтелектуальних контролерів, датчиків, актуаторів і електронних схем для емалювання роботи підсистем керування із задачею подальшого удосконалення управління вузлами прискорювача.

Удосконалити:

- метод діагностики та дистанційного контролю процесів формування пучків і опромінювання мішеневих пристроїв шляхом розробки та використання спеціалізованого контролера, що дасть змогу покращити повторюваність результатів експериментальних досліджень.

Практичне значення отриманих результатів:

Практичне значення результатів досліджень полягає в тому, що запропоновані моделі та методи можуть бути використані для модернізації й удосконалення автоматизованої системи управління малими прискорювачами наукового, технологічного та медичного призначення, у тому числі, прискорювача Ван де Граафа Харківського національного університету імені В.Н. Каразіна.

Зв'язок роботи з науковими програмами, планами, темами.

Дослідження, які будуть проводитися, відповідають плану науково-дослідної роботи, яка виконується в Харківському національному університеті імені В.Н. Каразіна: «Моделі інформаційних процесів та методи їх обробки», номер державної реєстрації 0116U003141 (01.2016 – 12.2020).

СТЄРВОЄДОВ Микола Григорович – к.т.н., доцент, завідувач кафедрою ЕУС факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: n.styervoyedov@karazin.ua; ORCID: 0000-0003-0136-6437.

Наукові інтереси:

- *автоматизовані системи керування технологічними процесами та програмне забезпечення для моделювання окремих модулів цих систем.*

АРТЮХ Сергій Олегович – аспірант 1-ого курсу факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: s.artyukh96@gmail.com; ORCID: 0000-0002-5856-3741.

Наукові інтереси:

- *автоматизовані системи керування технологічними процесами та програмне забезпечення для моделювання окремих модулів цих систем.*

УДК 004.7 **СЕКЦІЯ 7**

TEROKHIN V., STYERVOYEDOV M.

THE INTELLIGENT DETECTOR FOR SENSOR NODES OF THE RADIATION MONITORING NETWORK

Introduction

For the registration of ionizing radiation, semiconductor detectors have been widely used. They have outstanding detection performance, including sensitivity, energy resolution and signal-to-noise ratio. This is achieved by using crystals produced by new, advanced technologies.

Cadmium telluride (CdTe, CdZnTe) semiconductor detectors are considered promising in the field of γ-radiation detection and in isotope devices of technological control [1]. Their advantage over detectors based on Ge and Si crystals is operation at room temperature.

Recently, among the manufacturers such as REDLEN Technologoes, Canada, Kromek Group plc (formerly eV products), Great Britain, USA, Imdetek Corporation Ltd (Imdetek), People's Republic of China, Eurorad SA, France, Acrorad Co., Ltd ., Japan (manufacturer of detector CdTe), Ukrainian manufacturer entered. Scientists of NSC KIPT have developed detectors of γ-radiation (Fig. 1) with the energy range of the measured gamma radiation from 20 to 3000 keV. [2,3] Fig. 1.

Fig. 1. CdZnTe detectors, which are developed in NSC KIPT [2]

Process of detection dosimeter

Electronic and software detection systems do not meet all the requirements for dosimeter and radiometric equipment today.

This paper proposes a new architecture and an improved algorithm for detector equipment. The hardware part of the equipment is made with the use of modern microcontrollers. The algorithm of its work is shown in Figure 2 in the form of the UML diagram of activities.

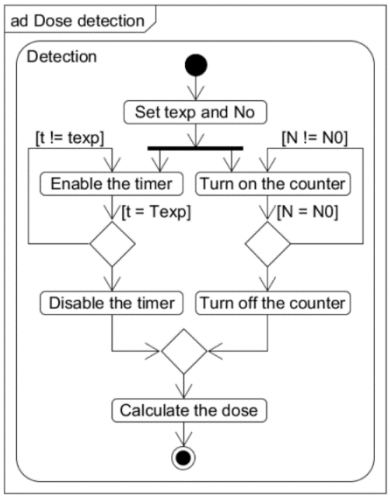


Fig. 2. Activity Diagram the process of detection dosimeter

© TEROKHIN V., STYERVOYEDOV M., 2021

The operation of the algorithm for calculating the dose of ionizing radiation includes measurements with a set value of exposure time and a set value of statistical uncertainty, which are performed in parallel.

Parallel counting begins with setting and activating the exposure time timer and pulse counter. The calculation takes place until one of the conditions [4] - the expiration of the exposure time (t = t0) or until the end of the set of the required number of measurements (N = N0)

After counting, the detection information is stored in the device's memory and transmitted through an intermediary to ThingSpeak cloud.

In Fig. 3 presents a modified block diagram of the spectrometer - dosimeter, which corresponds to the proposed method of measuring and calculating the radiation dose. Dose calculation is performed using data on the spectral sensitivity of the detector.

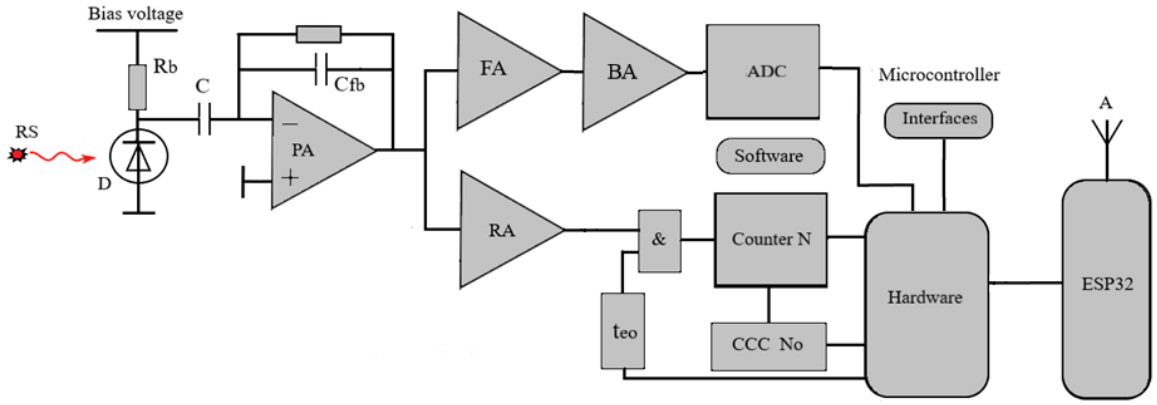


Fig. 3. Block diagram of a microcontroller dosimeter.

The scheme of the spectrometer - dosimeter consists of two components - the block of analog processing of signals from the semiconductor detector and the microcontroller module of analog to digital conversion and digital processing.

Figures 4 and 5 shows the energy spectra of the radiation obtained with the help of the equipment that has been developed. The calculation of the radiation exposure dose is based on the total number of recorded events. This takes into account the type of spectral dependence.

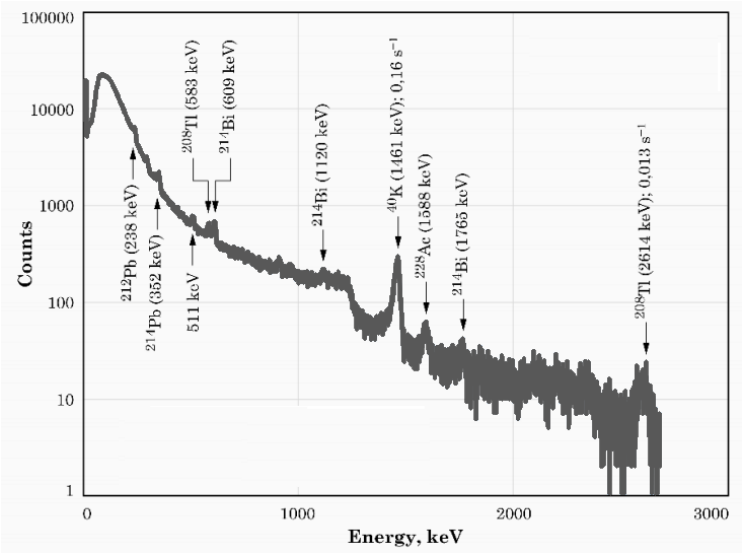


Fig. 4. The spectrum of background gravel radiation.

Calculation of the exposure dose of gamma radiation De and the exposure dose rate $X_e = D_e/t$ taking into account the dependence of the sensitivity of the CdZnTe detector on the energy of radiation. To do this, after recording the spectrum, the program calculates D_e in [μR], according to the formulas [4]:

$$D_e = \left(\mathrm{A} < E_{pulse} > + B\right) * N_{total}, \quad (1)$$

where the average pulse amplitude $< E_{pulse} >$ is calculated by using the formula (2):

$$< E_{pulse} > = \sum i \ (i * n_i / N_{total}), \quad (2)$$

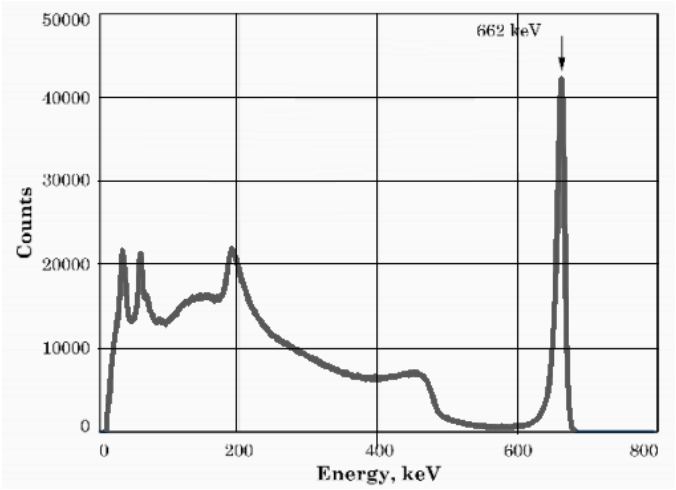


Fig. 5. The control spectrum of Cs - 137.

and $N_{total} = \sum_i n_i$– the total sum of pulses in the spectrum, or the final (3):

$$D_e = A \sum i \ i * n_i + B * \sum i \, n_i \,, \quad (3)$$

The values of coefficients A and B are determined during calibration of the detector system using standard gamma-ray isotope sources.

The exposure dose rate of gamma radiation X_e in [μr/s] is defined as:

$$X_e = D_e / t \,, \quad (4)$$

REFERENCES

1. V. I. Ivanov, A. E. Luchanskiy CdZnTe detectors and devices based on them for various applications: state of the art. XV Anniversary International Meeting "Problems of Applied Spectrometry and Radiometry" Kazan, October 7 - 11, 2019. PowerPoint Presentation (ritec.lv).
2. VE Kutny, AV Rybka, DV Kutny, IN Shlyakhov, DV Nakonechny. Development of dosimetric and spectrometric blocks for gamma radiation registration based on CdTe semiconductor compounds (CdZnTe) for Ukrainian NPPs // Questions of Atomic Science and Technology (VANT). Physics of Radiation Damage and Radiation Materials Science, 2004, vol. 3 (85), pp.96-100.
3. V. M. Azhazha, V. E. Kutny, A. V. Rybka, L. N. Davydov, I. N. Shlyakhov, A. A. Zakharchenko, D. V. Kutny, D. V. Nakonechny. Devices based on CdTe and CdZnTe for technological control and monitoring of the radiation situation at NPP // Science and Innovation, 2006, 2, No. 6, pp.31-38.
4. V. Terokhin, M. Styervoyeodov, O. Ridozub, S. Fomin. Intelektualniy vuzol of sensory framing of radio monitoring. Materials of the conference CMNT - 2020. Kharkiv, 2020. S. 49-51.
5. Zakharchenko A.A. Modeling the dosimetric properties of gamma-ray detectors based on high-resistance semiconductors. Dissertation for Candidate of Science physical - mat. sciences. Kharkov, 2009.

ТЕРЬОХІН Віталій Леонідович – аспірант кафедри електроніки та управляючих систем Харківський національний університет імені В.Н.Каразіна, Майдан Свободи 4,Харків-22,Україна, 61022 e-mail: kbs-com@karazin.ua; ORCID:0000-0001-7653-4488

Наукові інтереси: *Розробка основних підходів до створення апаратно-програмних засобів інформаційних систем радіаційного моніторингу*.

СТЕРВОЄДОВ Микола Григорович – к.т.н., доцент; завідуючий кафедри електроніки та управляючих систем Харківський національний університет імені В.Н.Каразіна, Майдан Свободи 4,Харків-22,Україна, 61022 e-mail: styervoyedov@yahoo.com; ORCID: 0000-0003-0136-6437. Наукові інтереси:

– Розробка підходів до створення апаратно-програмних засобів інформаційних систем.

УДК 621.391 **СЕКЦІЯ 7**

ТЕСЛЕНКО А.Ю., ПОГОРІЛА К.В.

МАТЕМАТИЧНА МОДЕЛЬ ПОБУДОВИ ТА ОБРОБКИ OFDM СИГНАЛІВ НА ФІЗИЧНОМУ РІВНІ WIFI ПРОТОКОЛІВ

Вступ

На теперішній час модифікації бездротового зв'язку розробники стикаються з рядом проблем. Одна з них, перешкоди від багатопроменевого прийому. Представлена проблема пов'язана з тим, що при передачі або прийомі інформації відбувається багатократне віддзеркалення сигналів від різних багатоповерхових споруд. Кардинальним вирішенням представленої проблеми стала відмова від використання систем з однією несучою частотою і впровадження технологій в основі яких лежить використання багаточастотних конструкцій.

Виходячи з вищесказаного, у розробників з'явилася ідея модифікувати вже наявну технологію OFDM, яка з'явилися в середині 60-х років, 20 століття. На той час, обчислювальні потужності наявних машин не задовольняли мінімальним можливостям для подальшого вивчення і, в зв'язку з цим, представлений метод використовувався у військових системах: KINEPLEX, ANDEFT, KATHRYN для паралельної передачі корисної інформації.

Технологія OFDM та її принцип

OFDM (англ. Orthogonal frequency-division multiplexing) - мультиплексування з ортогональним частотним розподіленням каналів. Представлену модель відносять до методів цифрової модуляції. Для більш детального уявлення про складові даної системи розглянемо технологію OFDM поелементно.

Метод OFDM використовується у бездротових мережах, таких як LTE, DVB-T, WiMax, та у провідних, таких як ADSL. Технологія представляє з себе ортогональне (квазіортогональне) частотне розподілення каналів з мультиплексуванням, використовуючи безліч несучих частот. Тобто вся доступна частота розбивається на кілька каналів для передачі даних, щоб канали були досить незалежні і не взаємодіяли один з одним, використовують відстані між потоками даних, що розраховується відносно частотного інтервалу:

$$\Delta f = \frac{1}{T}\,, \tag{1}$$

де T – тривалість інформаційного сигналу.

Дотримання представленої умови має назву ортогональність піднесних. Високошвидкісний потік даних розподіляється на деяку кількість паралельних бітових послідовностей, кожен з яких, моделюється своєю несучою. Для модуляції, зазвичай, використовують тривіальні схеми цифрової модуляції, наприклад QPSK, 16QAM та ін.

Функції, які описують несучі сигнали всіх частотних підканалів ортогональні один одному. З точки зору математики, це означає, що взаємна проекція один на одного цих функцій, обчислена в деякій метриці, дорівнює нулю. Наприклад, в метриці Гілберта:

$$\int_0^T \sin(2\pi f t + \varphi_1) \cdot \sin\left(2\pi\left(f \pm \frac{k}{T}\right)t + \varphi_2\right) dt \equiv 0; \tag{2}$$

$$\int_0^T \sin(2\pi f t + \varphi_1) \cdot \cos\left(2\pi\left(f \pm \frac{k}{T}\right)t + \varphi_2\right) dt \equiv 0, \tag{3}$$

де f – частота (Гц); $k = 1,2,3\ldots$ – ціле число; φ_1, φ_2 – довільні початкові фази.

Так як відрізки сигналів будь-якої довільної пари з ортогональних підканалів $S_i(t), S_j(t)$ на тривалості OFDM символу T можуть бути представлені у вигляді суми довільного числа гармонійних коливань з різною початковою фазою то з (2) і (3) випливає їх взаємна ортогональність:

© ТЕСЛЕНКО А.Ю., ПОГОРІЛА К.В., 2021

$$\int_0^T S_i(t), S_j(t)\, dt \equiv 0, \tag{4}$$

В якості тривіального сигналу використовується гармонічна функція - синусоїда. Для уявлення про те, що з себе представляє сигнал OFDM будемо вважати наступне:

$$S(t,\varphi) = \sum_{i=0}^{N-1} \sin(2\pi(f_{\text{н}} + \Delta f)t * \varphi_i), \tag{5}$$

де t – момент часу, φ – фаза сигналу, N – кількість піднесних, $f_{\text{н}}$ – це початкова частота, також це нижня частота; $\Delta f = \frac{i}{T}$ – відстань між сигналами.

Ключовим принципом використання OFDM є можливість використання захисного інтервалу, це обумовлено досить великою тривалістю передачі одного символу, а швидкість потоку - зменшується. Це робить представлений метод більш стійким до міжсимвольної інтерференції, тобто один сигнал не накладається на інший. Також має місце збільшення спектральної ефективності, цьому сприяє розподілення енергії на піднесні.

Слід зауважити, що при багаточастковому сигналі захисний інтервал використовується після групи OFDM-символів, коли при послідовному – між кожними символами. Тобто, з'являється можливість зменшити необхідну кількість захисних інтервалів.

Другою цінністю застосування є те, що даний метод стійкий до частотного затухання. Це означає, що не є проблемою відновити загублені дані, тому що інформація дублюється на безліч несучих та передається паралельно.

Практична реалізація технології

У стандарті розробки OFDM сигналу при демодуляції використовують швидке перетворення Фур'є (Fast Fourier Transform - FFT), але при реалізації даної ідеї у середовищі MathCad, були виявлені два вагомих недоліка. Перший мінус закладається у тому, що для правильної роботи FFT необхідно розраховувати таку кількість вимірювань, щоб їх чисельність була кратна ступені двійки. Це пов'язано з тим, що таке перетворення для розрахунку використовує два значення, тобто кількість операцій розраховується, як

$$K_o = 2^n; \tag{6}$$

З першого пункту випливає другий недолік. Через таку кількість обчислень, з'являється безліч коефіцієнтів в остаточній відповіді. Наприклад, якщо уявити, що необхідно передати 16 біт повідомлення, тобто використати таку ж кількість піднесних. Для демодуляції сигналу необхідно підібрати число, яке буде відповідати ступені двійки. У рамках прикладу передачі 16 біт інформації, взяти 2^{11}, то кількість вимірів буде дорівнювати 2048 згідно з формулою (6). При використані перетворення Фур'є отримаємо 1024 розрахункових значень з яких 16 коефіцієнтів буде співпадати з початковим повідомленням. Тобто надмірність розрахунків складає 98 процентів.

Для того, щоб прибрати таку велику надмірність пропоную для демодуляції використати систему лінійних алгебраїчних рівнянь.

$$A * X = B, \tag{7}$$

де змінна A буде мати вигляд матриці розміром 16 на 16: стовпці – кількість піднесних, рядки – виміри значень у конкретний момент часу; змінна B вектор, якій містить фазу кожного переданого біту повідомлення; змінна X – розрахунки демодулятора.

Якщо представити модель сигналу у виді рівнянь з невідомими згідно з формулою (7), то для нашого прикладу з 16 піднесними, знадобиться така же кількість рівнянь. Цей метод вирішає одразу два поставлених недоліка, тобто необхідно зробити 16 вимірів замість 2048 і відповідно пропадає надмірність розрахунків. Також не має сенсу у пошуку, з 1024 отриманих коефіцієнтів при використанні FFT, переданих бітів інформації. Звичайно, виникає можливість швидше

проводити розрахункові операції: така різниця не помітна при малому обсягу оброблювальної інформації, але, якщо суттєво збільшити повідомлення, то перепад часу буде відчутний.

Висновки

Використання OFDM технології виключає міжсимвольну інтерференцію, поліпшує спектральну ефективність. Описані переваги роблять модель на порядок краще других існуючих методів, тим паче, якщо при демодуляції використовувати запропоновану систему лінійних алгебраїчних рівнянь.

ЛІТЕРАТУРА

1. Маглицкий Б.Н. Основы технологии OFDM: Учебное пособие. Сибирский государственный университет телекоммуникаций и информатики, Новосибирск: 2016. 116 с.
2. Таненбаум Э., Уэзеролл Д. Компьютерные сети. Питер: 5-е изд. СПб, 2012. 690 с.
3. Олифер В., Олифер Н. Компьютерные сети. Принципы, технологии, протоколы : Учебник для вузов. Питер: 5-е изд. СПб, 2016. 992 с.
4. Лосев Ю. І., Руккас К. М., Шматков С. І. Комп’ютерні мережі: навч. посіб. / за редакцією Ю. І. Лосева. Харків : ХНУ імені В. Н. Каразіна, 2013. 248 с.
5. Лапшова Н.А. Мультиплексирование с ортогональным частотным разделением (OFDM). *«Научно-практический электронный журнал Аллея Науки».* 2017. Вип. 10. URL: https://www.elibrary.ru/download/elibrary_29736629_40709206.pdf (Last accessed:05.04.2021)

ТЕСЛЕНКО Альона Юріївна – студентка 4 курсу, факультету комп’ютерних наук, напрямок підготовки 125 кібербезпека; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, Україна, 61000; e-mail: xa11867833@student.karazin.ua; ORCID: 0000-0002-8982-7439.

Наукові інтереси:

– комп’ютерні мережі, захист комп’ютерних мереж, завадостійкість і безпека даних в інформаційних і комунікаційних системах.

ПОГОРІЛА Каріна Валеріївна – студентка 4 курсу, факультету комп’ютерних наук, напрямок підготовки 125 кібербезпека; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, Україна, 61000; e-mail: xa11867830@student.karazin.ua.

Наукові інтереси:

– соціальна інженерія, інсайдерські загрози, DDos-атаки, VPN-технології, SQL-ін'єкції.

УДК 519.7 **СЕКЦІЯ 2**

ТКАЧЕНКО А.М., СТРІЛЕЦЬ В.Є.

ШТУЧНІ НЕЙРОННІ МЕРЕЖІ У ЗАДАЧАХ КЛАСИФІКАЦІЇ

Задача класифікації – це задача розбиття множини об'єктів або спостережень на апріорно задані групи, називані класами, всередині кожної з яких вони вважаються схожими один на одного, та мають приблизно однакові властивості й ознаки [1].

Класифікація є однією з найважливіших задач інтелектуального аналізу та машинного навчання. Серед численних використань задачі класифікації, можна виділити такі: класифікація даних, розпізнавання рукописного тексту, класифікація зображень, розпізнавання мови, медична діагностика, виявлення несправностей, можливість банкротства тощо [2].

Задачу класифікації можна вирішити за допомогою великої кількості методів. До них відносяться: байєсівський класифікатор, статистичні класифікатори, метод найближчого сусіда, нейронні мережі, дерева рішень, метод опорних векторів та ін. [3].

Безумовно, кожен з цих методів має як свої переваги, так і недоліки. Розглянемо більш детально штучні нейронні мережі та особливості їх використання у задачах класифікації.

Штучна нейронна мережа – сукупність з'єднаних між собою нейронів, що моделюють структуру й функції біологічних нейронів. У загальному випадку, нейронна мережа може містити вхідний шар, на який надходять зовнішні сигнали, вихідний, що відображає результуючу реакцію мережі на вхідні сигнали, а також приховані шари (Рис.1) [4].

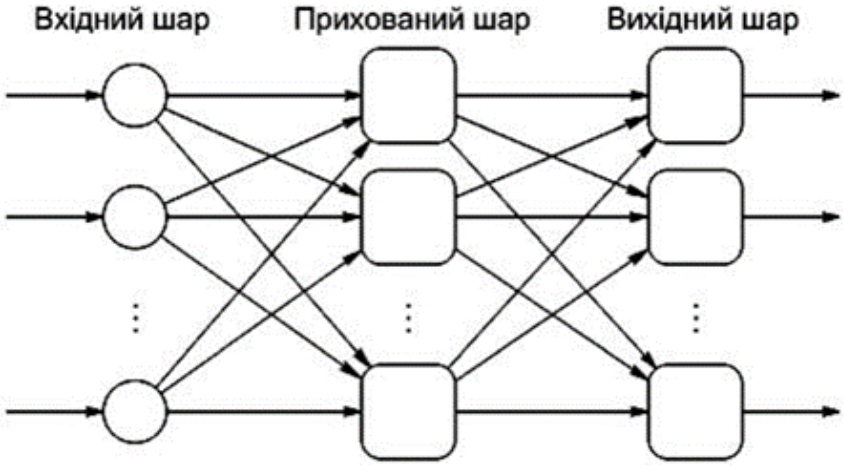


Рис.1 – Схема штучної нейронної мережі

Важливою характеристикою нейронних мереж є їх здатність до навчання або тренування, яке полягає у визначенні таких параметрів мережі, при яких вона формує необхідні вихідні сигнали. Протягом цього процесу інформація поширюється в мережі, спричиняючи процес адаптації синаптичних ваг, що залежить від значень вихідних сигналів та найменувань відповідних класів.

Опишемо алгоритм побудови класифікатора на основі нейронних мереж.

Умовно алгоритм розділяється на чотири частини:

- робота з даними;
- попередня обробка;
- конструювання, навчання та оцінка якості мережі;
- використання та діагностика.

На етапі роботи з даними необхідно скласти базу даних із прикладів, характерних для даної задачі та розбити всю сукупність даних на дві множини: навчальну і тестову.

Попередня обробка полягає в виборі системи ознак, характерних для даної задачі, і перетворенні даних відповідним чином для подачі на вхід мережі (нормування, стандартизація

© ТКАЧЕНКО А.М., СТРІЛЕЦЬ В.Є., 2021

і т.д.). В результаті повинно бути отримано лінійно-відокремлюваний простір множини зразків. Ще одним кроком на етапі попередньої обробки є вибір системи кодування вихідних значень.

На наступному етапі виконується значна частина роботи:

– вибір топології мережі (визначення кількості шарів, числа нейронів у шарах і т.д.);

– вибір функції активації нейронів (наприклад «сигмоїду»);

– вибір алгоритму навчання мережі;

– оцінка якості роботи мережі на основі підтверджуючої множини або іншим критерієм,

– оптимізація архітектури (зменшення ваг, проріджування простору ознак і т.д.);

– вибір варіанту мережі, що забезпечує найкращу спроможність до узагальнення та оцінка якості роботи по тестовій множині.

Останній етап складається з вирішення чотирьох завдань: з’ясування ступеню впливу різних факторів на прийняте рішення (евристичний підхід); підтвердження того, що мережа дає необхідну точність класифікації (кількість неправильно розпізнаних прикладів нечисленна); зміні способу представлення зразків або бази даних при необхідності та практичне використання мережі для вирішення поставленої задачі.

Для розв’язання задачі класифікації може бути використаний як багатошаровий перцептрон, так і мережі з самоорганізацією на основі конкуренції, що використовують алгоритм навчання без учителя. Використання самоорганізації не вимагає апріорного завдання класів і дозволяє виділити необхідну їх кількість в процесі навчання і функціонування. Контрольоване навчання можливе не тільки на початковому етапі при підготовці до роботи, але і безпосередньо в процесі роботи. При появі нових класів мережі, що самоорганізуються, не потребують перенавчання, а здатні адаптуватися на підставі нових значень.

Розглянемо особливості деяких нейронних мереж, які застосовують у задачах класифікації.

Багатошаровий перцептрон. На входи нейронної мережі подається вхідний вектор ознак; відповідно до алгоритму зворотного поширення помилки, відбувається коригування ваг згідно заздалегідь зумовленого класу вхідного вектору. Після навчання на певній кількості вхідних образів відбувається перевірка мережі на зразках, які не брали участі в навчанні.

Мережа Кохонена. На входи нейронної мережі подаються значення ознак обраного об'єкта. Нейромережа обробляє ці сигнали, після чого в вихідному шарі визначається нейрон-переможець. Нейрон-переможець вихідного шару визначає клас об'єкта, ознаки якого були подані на входи нейромережі. Оскільки кожному класу в процесі навчання мережі присвоюється класифікаційний код, то при подачі на входи нейронної мережі вектору ознак невідомого об'єкта, мережа здатна визначити його код. Якщо нейрон-переможець не визначає клас об'єкта, то для нього створюється новий клас [5].

Ймовірнісні нейронні мережі. Ймовірнісна мережа вчиться оцінювати функцію густини ймовірності, її вихід розглядається як очікуване значення моделі в даній точці простору входів. Це значення пов’язане з густиною ймовірності спільного розподілу вхідних і вихідних даних. При розв’язання задач класифікації можна оцінити густину ймовірності для кожного класу, порівняти між собою ймовірності приналежності до різних класів і обрати модель з параметрами, при яких густина ймовірності буде найбільшою. Істотним недоліком мережі є її розмір, оскільки вона фактично вміщує в собі всі навчальні дані, потребує багато пам’яті і може повільно працювати.

Висновок

У роботі приведено основні поняття задачі класифікації. Наведено розбір методу нейронних мереж. Описано головні відомості відносно нього та алгоритм побудови класифікатора.

Для того, щоб побудувати якісний класифікатор, необхідно мати якісні дані. Жоден з методів побудови класифікаторів, заснований на нейронних мережах або статистичний, ніколи

не дасть класифікатор потрібної якості, якщо наявний набір прикладів не буде достатньо повним і представницьким для задачі, з якою доведеться працювати системі.

Описано три різновиди нейронних мереж, які вирішують задачу класифікації. Саме ці архітектури мереж було обрано для подальшого дослідження.

ЛІТЕРАТУРА

1. Інформаційні системи та технології в управлінні Методичні вказівки, теоретичні відомості і завдання до лабораторних робіт для студентів та магістрів денної форми навчання спеціальності 7.803060101 Менеджмент організацій і адміністрування. Частина 3. Класифікація в бізнес-аналітиці. / Укл.: Біла Н.І. – Запоріжжя: ЗНТУ, 2014. – с. 50.

2. Марченко О.О., Россада Т.В. Актуальні проблеми Data Mining: Навчальний посібник для студентів факультету комп'ютерних наук та кібернетики. — Київ. — 2017. — 150 с.

3. Han J. Data Mining: Concepts and Techniques / J. Han, M. Kamber. – Edition 2, San Francisco : Elsevier Inc., 2006. – 743 p.

4. Lippman R.P. An Introduction to Computing with Neural Nets / R.P. Lippman // IEEE ASSP Magazine. – 1987. – Vol. 3. No 4. – Pp. 4-22.

5. ami.nstu.ru - Использование нейронных сетей в Data Mining [Електронний ресурс] Режим доступа: https://ami.nstu.ru/~vms/lecture/data_mining/n_set.htm

ТКАЧЕНКО Андрій Максимович – студент факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: tkachenko2020ku51@student.karazin.ua; ORCID: 0000-0002-0547-3101.

Наукові інтереси:

- *машинне навчання;*
- *штучні нейронні мережі.*

СТРІЛЕЦЬ Вікторія Євгенівна – к.т.н.; доцент кафедри прикладної і теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: viktoria.strilets@karazin.ua; ORCID: 0000-0002-2475-1496.

Наукові інтереси:

- *методи машинного навчання;*
- *методи штучного інтелекту.*

УДК 004.051 **СЕКЦІЯ 6**

ТОВСТОКОРЕНКО О.Ю.

ПРО УПРАВЛІННЯ ВАРІАБЕЛЬНІСТЮ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У РОЗРОБЦІ ТА СУПРОВОДІ СИСТЕМ «РОЗУМНИЙ БУДИНОК»

Технологія «Інтернету речей» з кожним роком стає все більш використовуваною не тільки на підприємствах, а і серед приватного споживача за рахунок як здешевлення комплектуючих так і розширення модельного комплексу та сценаріїв використання[1-3]. Враховуючи гетерогенність систем «Розумного будинку» та необхідність використання певної кількості наборів компонент різних виробників, актуальності набуває необхідність застосування концепції управління варіабельністю такого класу систем[4]. В результаті аналізу певного набору систем управління «Розумним будинком» (Smart Home Management System SHMS)[5], з відкритим програмним кодом, було побудовано загальну компонентну архітектуру такої системи (рис. 1).

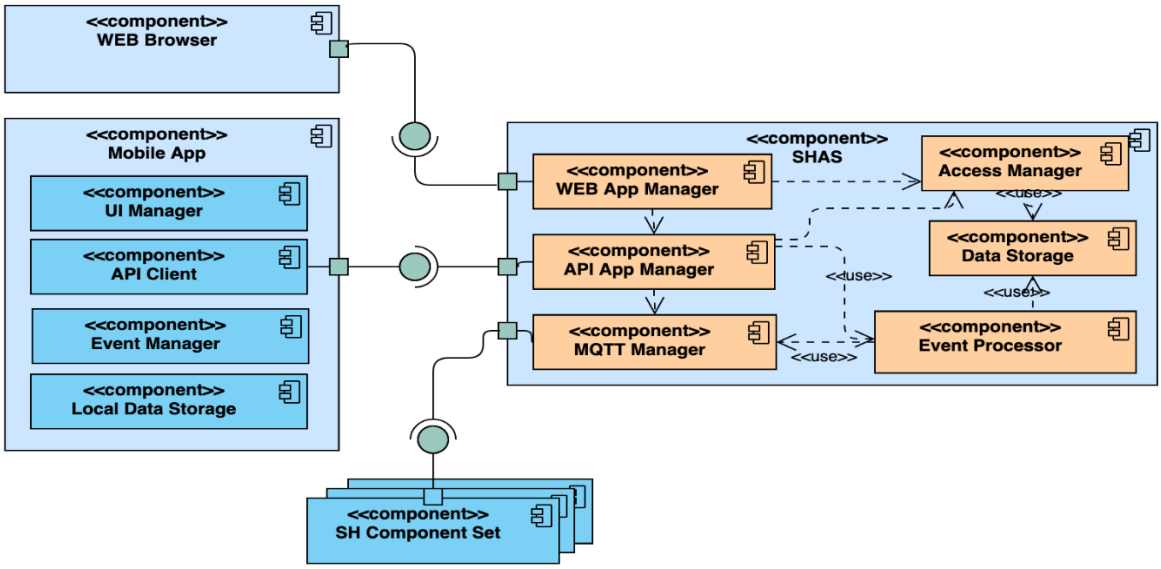


Рис. 1 Загальна архітектура SHMS

SHMS завжди мають сховище даних (Data Storage), де зберігаються данні про поточний стан системи та данні користувачів; компонент обробки станів (Event Processor), який потрібен для оновлення та перевірки станів сенсорів, актуаторів та сценаріїв SHMS; компонент MQTT Manager використовується для управління повідомленнями про зміну станів сенсорів та актуаторів. API App Manager – компонент web – інтерфейсу для взаємодії з зовнішніми програмними системами (наприклад мобільні пристрої); WEB App Manager – компонент для користувацького доступу через веб браузер, в якості основи для роботи з даними використовує API App Manager.

Зважаючи на зазначену архітектуру SHMS, для впровадження підтримки певного фізичного компоненту (сенсора або актуатора), як на етапі розробки, так і на етапі супроводу, необхідно виконати наступний набір кроків (рис. 2). В процесі розробки зазначені кроки не потребують додаткових змін процесу. Але на етапі підтримки, зважаючи на динамічність комплектації систем «Розумним будинком» (SHS) процес розширення SHMS необхідно буде неодноразово відтворювати з новими вимогами. Звідси витікає необхідність впровадження методів управління варіабельністю.

© ТОВСТОКОРЕНКО О.Ю., 2021

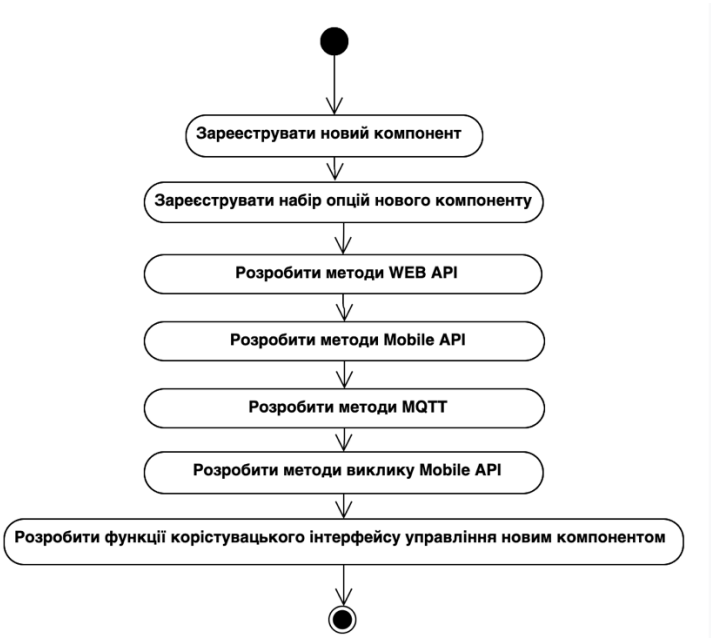


Рис. 2 Послідовність кроків для розширення системи певним фізичним компонентом

В описаному випадку точкою варіабельності є підтримка функціоналу управління певним фізичним компонентом (мається на увазі прості команди, які можна застосовувати для певного фізичного компоненту). Також іншим видом точок варіабельності будуть динамічно створювані сценарії[5].

В результаті аналізу певного набору систем управління «Розумним будинком» (Smart Home Management System SHMS) [5-8], з відкритим програмним кодом, було побудовано загальну компонентну архітектуру з додатковими компонентами, які дозволяють урахувати процес управління варіабельністю (виділено рожевим кольором).

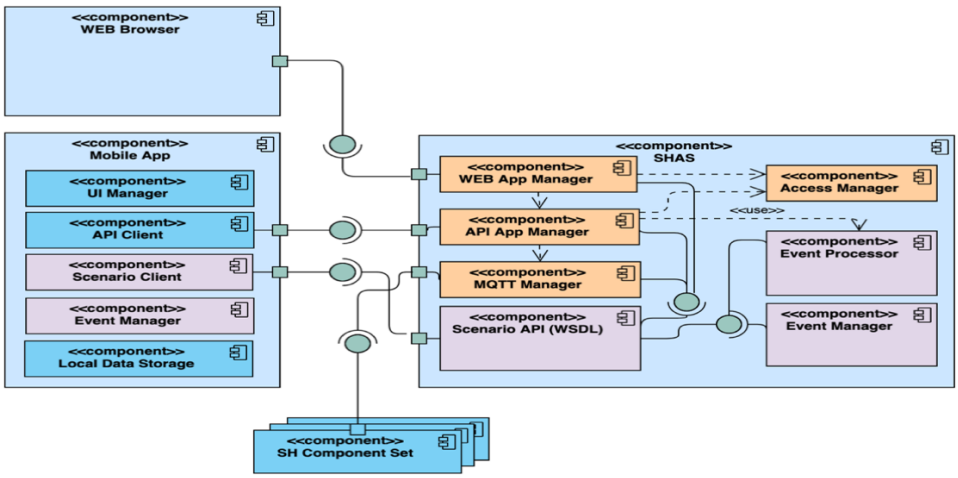


Рис. 3 Загальна архітектура SHMS з урахуванням процесу управління варіабельністю

За допомогою запропонованих програмних компонент пропонується часткова автоматизація зазначених на діаграмі (рис. 2) кроків, за рахунок додавання до основного функціоналу SHMS можливості динамічної адаптації функціоналу управління новим інтегрованим фізичним компонентом. З використанням Event Manager компонента планується створювати/редагувати функціональну логічну модель управління компонентами SHMS. Далі, на основі логічної моделі, за допомогою Scenario API, в автоматичному режимі будуть згенеровані API моделі опису (наприклад WSDL/YAML) логіки запропонованої на попередньому етапі. На основі моделей опису, також в автоматичному режимі, можна вже генерувати API інтерфейси для зовнішньої

інтеграції та методи для роботи з MQTT процесами. Опубліковані моделі опису також будуть використовувати мобільні пристрої, на основі отриманих моделей буде побудовано каталог нового функціоналу та відповідний користувацький інтерфейс. За рахунок використання запропонованої архітектури та впровадження процесів управління варіабельністю планується скоротити час та затрати на підтримку SHMS. В якості подальших досліджень планується побудувати кастомізовану мову доменного моделювання для Event Manager / Scenario API компонентів. Для підтримки таких процесів можна застосувати доменне моделювання (domain modeling)[8].

ЛІТЕРАТУРА

1. Mohsen A., The Internet of Things: Limitless Opportunities for Business and Society. Journal of Strategic Innovation and Sustainability Vol. 12(1) 2017.
2. Berger Th., Chechik M., Kehrer T. Software Evolution in Time and Space: Unifying Version and Variability Management. Dagstuhl Seminar Reports. 2019, vol. 9, issue 5, pp. 1–31.
3. Steinberger M., Reinhartz-Berger I., Tomer A. Cross Lifecycle Variability Analysis: Utilizing Requirements and Testing Artifacts. Journal of Systems and Software. 2018, vol. 143, pp.208–230.
4. Товстокоренко О.Ю., Гамзаєв Р. О., Ткачук М. В., Використання концепції лінійок програмних продуктів в контексті процесу проектування систем «Розумний дім». Вісник Національного технічного університету "ХПІ" - Харків: НТУ ”ХПІ”. – 2020. - № XX (1298) – с. 3-10. ISSN 2079-0023 (print), ISSN 2410-2857 (online)
5. Товстокоренко О.Ю., Ткаченко М.В., Використання концепції лінійок програмних продуктів в контексті процесу проектування систем «Розумний дім». «Наукові підсумки 2019 року», XXXVII Міжнародна науково-практична інтернет-конференція. – м. Вінниця, 09 грудня 2019 року. – Ч.14, с. 84.
6. Mazzara M., Afanasyev I., Sarangi S. R., Distefano S., Kumar V., Ahmad M. A Reference Architecture for Smart and Software-Defined Buildings. IEEE International Conference on Smart Computing (SMARTCOMP 2019). Washington DC, USA. IEEE Publ., 2019. P. 167–172. DOI: 10.1109/SMARTCOMP.2019.00048.
7. Pandit D., Pattanaik S. Software Engineering Oriented Approach to Iot-Applications: Need of the Day. International Journal of Recent Technology and Engineering (IJRTE). 2019. Vol. 7, issue 6. P. 886– 895.
8. Gamzayev R. O., Karaçuha E., Tkachuk M. V., Tovstokorenko O. Y. An Approach to Assessment of Dynamic Software Variability in Mobile Applications Development. Visnyk XNU imeni V. N. Karazina [Bulletin of KhNU named after V.N. Karazina]. Kharkiv, 2018, no. 40. pp.14–25.

ТОВСТОКОРЕНКО Олег Юрійович – аспірант кафедри програмної інженерії та інформаційних технологій управління, Національний технічний університет «Харківський політехнічний інститут», вул. Кирпичова, 2, Харків, Україна, 61022; старший викладач кафедри моделювання систем і технологій, Харківський національний університет імені В. Н. Каразіна., Проспект Незалежності, 6, Харків, , Україна, 61000; e-mail: oleg.tovstokorenko@karazin.ua; ORCID: 0000-0003-2664-1650.

Наукові інтереси:

– *процеси управління варіабельністю програмних систем «Розумний будинок»*.

UDC 65.0(075.8) **SECTION 2,7**

TOLSTOLUZKYI Y. D., BERDNIKOV A. G.

INVESTIGATION OF PARAMETERS OF SEMANTIC-NUMERICAL SPECIFICATIONS OF PARALLEL PROCESSES

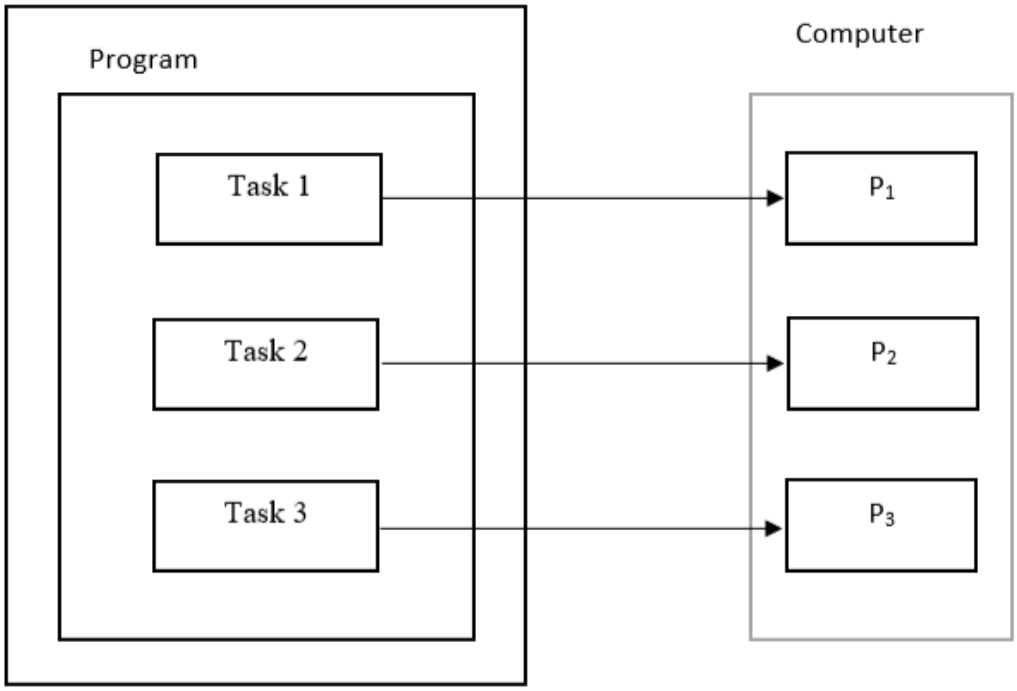
Introduction

In the age of rapidly developing digital technologies, the number and scale of any calculations is increasing. In this regard, there is an acute problem of reducing the time spent on the implementation of computations. One of the possible solutions of this problem is to use parallel technologies. Parallel calculations allow to reduce the time needed to solve the problem as compared to the sequence of computations.

Current trends in the development of processors consist in increasing the number of cores, rather than an increase in processor speed. This leads to the fact that the time difference computing and parallel algorithms will only increase. As a result, it can be noted that the topic of articles related to the investigation of the parameters of semantic-numerical specifications of parallel processes is relevant.

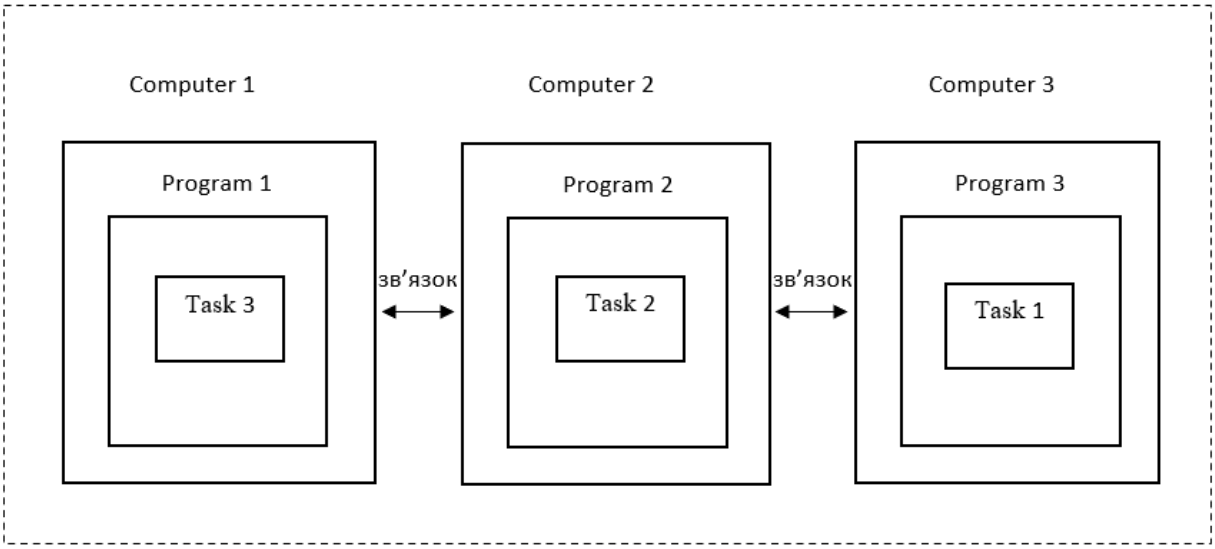
Problem statement

Parallelism can be achieved by two methods: parallel programming method and distributed programming method. The essence of parallel programming consists in the possibility of calculating the distribution between two or more processors within the same physical (or virtual) of the computer, while the essence of the distributed programming consists in the possibility of calculating the distribution between two or more processors, but with the ability to use more than one computer.



Pict. 1. Parallel Applications

© TOLSTOLUZKYI Y. D., BERDNIKOV A. G., 2021



Pict. 2. Distributed running applications.

Advantages of parallel programming is the speed. Parallel program execution time is much less than their successive versions. The disadvantages are expressed in the complexity of the implementation of these types of programs. Parallel programming has gained the greatest popularity in the scientific industry, since it is necessary to process huge amounts of information in parallel there.

The advantage of distributed programming is the ability to use resources located on the network (global or local). The disadvantage, like parallel programming, is the difficulty in creating such programs. Many services presented on the Internet are distributed (calculations and calculations are performed on third-party servers). [1-4]

The following software tools are used to implement parallel and distributed programs:

• OpenMP is an application interface standard for parallel shared memory systems. OpenMP implements computation using multithreading, in which the main thread creates a group of secondary threads, and the task is distributed among them. Threads are assumed to run in parallel on a machine with multiple processors.

• Windows API - multithreaded applications for C ++. Provides a direct way for user applications to interact with the Windows operating system.

• MPI (Message Passing Interface) is a standard for message passing systems between parallel executable processes. MPI is the most widespread data exchange interface standard in parallel programming; its implementations exist for a large number of computer platforms. Used in the development of programs for clusters and supercomputers. The main means of communication between processes in MPI is passing messages to each other. MPI is focused on distributed memory systems, that is, when data transfer costs are high, while OpenMP is focused on shared memory systems.

• PVM (Parallel Virtual Machine) allows you to combine a heterogeneous (but networked) set of computers into a common computing resource. It has more advanced capabilities than its popular MPI counterpart in terms of control of computations: there is a specialized parallel system control console and its graphical equivalent XPVM, which allows you to clearly demonstrate the operation of the entire system.

It should be noted that all these software tools are not suitable for automatic parallelization of algorithms and programs. One of the new promising devices for the formal specification of parallel computing, which is being developed at the Department of Theoretical and Applied Systems Engineering, is the structure of the Semantic-Numerical Specification.

Model

The apparatus of structures of the Semantic-Numerical Specification (SCS) is a means of formal specification of static and temporary objects of synthesis and analysis of parallel computer systems, such as: parallel process models, complex information processing systems, and others. Picture 3 provides a classification of objects, their formal specification.

Pict. 3. Classification of objects of algebra space-time semantic-numerical specification.

Such a large number of classes of objects is necessary due to factors affecting the efficiency of parallel computing systems. These factors include: the features of algorithms, their concept, principles for constructing algorithms, the composition of methods for parallel data processing, requirements and restrictions, and others.

The apparatus of structures of the Semantic-Numerical Specification (SNS) contains the following functional parts:

• static and temporal semantic-numerical structures of arrays of semantic, numerical, static, temporal and spatial characteristics of the specified object of synthesis / analysis;

• a set of operations providing static, temporal and spatial transformations of semantic-numerical structures and objects of synthesis and analysis of parallel hardware and software and parallel computing systems in general;

• a library of programs of operations that support equivalent transformations of static, temporal and spatial semantic-numerical structures and objects of synthesis and analysis of parallel hardware and software and parallel computing systems in general;

• a library of programs that provide automatic generation of semantic-numerical structures from the source code of C programs. [5]

© TOLSTOLUZKYI Y. D., BERDNIKOV A. G.

Thus, the apparatus of structures of the Semantic-Numerical Specification can and is expediently used for parallel computations. This will significantly simplify the work and reduce the time spent on the implementation of algorithms. The need to perform tasks in parallel is important not only for computing, but also for IT projects. Investigation of the SNS parameters will allow the use of this apparatus of structures for automatic parallelization of IT project tasks, which will lead to a decrease in the time spent on project implementation and to the emergence of new risks.

REFERENCES

1. Гергель, В.П., Стронгин, Р.Г. (2003, 2 изд.). Основы параллельных вычислений для многопроцессорных вычислительных систем. - Н.Новгород, ННГУ.
2. Воеводин В.В., Воеводин Вл.В. (2002). Параллельные вычисления. – СПб.: БХВ-Петербург.
3. Немнюгин С. (2009). Модели и средства программирования для многопроцессорных систем– СПб.: БХВ-Петербург.
4. Хьюз К., Хьюз Т. (2004). Параллельное и распределенное программирование на C++.: Пер. с англ. – М.: Издательский дом «Вильямс», 2004. – 672с.
5. Поляков Г.А., Шматков С.И., Толстолужская Е.Г., Толстолужский Д.А. (2012). Синтез и анализ параллельных процессов в адаптивных времяпараметризованных вычислительных системах. – Х.: ХНУ имени В.Н. Каразина, 2012.-672с.

TOLSTOLUZKYI Yevhen– student of the Computer Science Department, Kharkiv National University, V.N. Karazin, Maidan Svoboda, 4, M. Kharkiv, Ukraine, 61022; e-mail: evventol@gmail.com ; ORCID: 0000-0002-2039-0267.

Science interests:

- *Application of decision theory methods in the management of complex telecommunications systems..*

BERDNIKOV Anatolyi – Ph.D., associate professor of the Department of TPS, Associate Professor; Kharkiv National University, V.N. Karazin, Maidan Svoboda, 4, M. Kharkiv, Ukraine, 61022; e-mail: a.berdnikov@karazin.ua; ORCID 0000-0001-7923-0754

Science interests:

- *Application of decision theory methods in the management of complex telecommunications systems, project management.*

УДК 004.4 **СЕКЦІЯ 2**

ТУРЕНКО М.С., СТРІЛЕЦЬ В.Є.

АНАЛІЗ СТРАТЕГІЙ ТЕСТУВАННЯ ВЕБ-САЙТІВ

Тестування програмного забезпечення – процес перевірки відповідності заявлених до продукту вимог і реально реалізованої функціональності, який здійснюється шляхом спостереження за його роботою в штучно створених ситуаціях і на обмеженому наборі тестів, обраних певним чином.

Можна визначити такі основні цілі тестування програмного забезпечення:

- надання інформації про якість програмного забезпечення кінцевому замовнику;
- підвищення якості програмного забезпечення;
- запобігання появи дефектів.

Головним завданням тестування є пошук дефектів.

Тестування завжди було невід'ємною частиною розробки програмного забезпечення. Воно було присутнє навіть 50 років тому, наприклад у Водоспадній моделі розробки, яка полягає в тому, що кожна наступна стадія починається тільки після того, як закінчується попередня. Так і в більш сучасному підході до розробки – Agile, який включає в себе практики, підходи і методології, що допомагають створювати продукт максимально ефективно.

Забезпечення якості програмного забезпечення є складним процесом, який включає в себе безліч різних моментів, таких як:

- визначення параметрів якості;
- розробка стратегії тестування;
- проведення перевірок продукту;
- виявлення та опис дефектів;
- вимірювання отриманих результатів;
- оптимізація процесів;
- внесення необхідних змін.

Ручне й автоматизоване тестування

Варто зазначити, що одним з найважливіших моментів в розробці програмного забезпечення є вибір того, чи варто впроваджувати автоматизацію або виконання перевіркок буде виконуватися тільки вручну. Обидва підходи мають свої переваги та недоліки, які варто враховувати при прийнятті вище сказаного рішення.

Для кращого розуміння різниці в ручному і автоматизованому підходах введемо таке визначення як тест-кейс.

Тестовий випадок (тест-кейс) – це артефакт, який описує сукупність кроків, конкретних умов і параметрів, необхідних для перевірки реалізації функції, яка тестується, або її частини.

При ручному підході тест-кейси виконуються вручну без використання програмних засобів. Власне, це зрозуміло з назви. При автоматизованому тестуванні проходження тест-кейсів здійснюється за допомогою спеціально розроблених скриптів. Час, вартість і якість є трьома найважливішими факторами на будь-якому проекті. Дуже важливо максимально скоротити вартість і час зі збереженням якості продукту. Спираючись на це можна зробити вибір серед двох підходів до тестування, оскільки кожен з них може впоратися з цим завданням краще іншого.

Нижче порівняємо ці два підходи у вигляді таблиці за наступними основними характеристиками:

- надійність;
- швидкість виконання;
- вартість;
- регулярність запуску тест-кейсів.

© ТУРЕНКО М.С., СТРІЛЕЦЬ В.Є., 2021

Табл.1 Порівняння ручного та автоматизованого підходів до тестування

Характеристика	Ручне тестування	Автоматизоване тестування
Надійність	Результати менш надійні через вплив людського фактора	Безпомилкові результати через відсутність впливу людського фактора
Швидкість виконання	Повільна швидкість, так як потрібні великі трудовитрати	Так як тестування відбувається автоматично, і в ідеальному варіанті тести виконуються паралельно, швидкість зростає на порядок
Вартість	Оплата праці інженерів з ручного тестування	Інструменти для розробки рішення по автоматизації, оплата праці інженерів по автоматизації тестування
Регулярність запуску тест-кейсів	Тест-кейси виконуються кілька разів за весь час без багаторазових повторень	Тест-кейси виконуються регулярно протягом довгого періоду часу

Варто відзначити, що ручне тестування рекомендується у таких видах тестування:

– дослідницьке тестування (сценарії для тестування вибираються виходячи з досвіду інженера);

– тестування юзабіліті (перевірки зручності використання програмного забезпечення);

– інтуїтивне тестування (виконується без заздалегідь розробленого сценарію і певних результатів, тестувальник імпровізує і покладається на здоровий глузд, свій досвід і знання продукту).

В той час як, автоматизоване тестування краще використовувати у нижче наведених видах тестування:

– регресійне тестування (спрямований на перевірку змін, зроблених в додатку або навколишньому середовищу, для підтвердження того факту, що існуюча раніше функціональність працює як і раніше);

– навантажувальне тестування (перевірки зручності у використанні програмного забезпечення);

– тестування локалізації (виконується без заздалегідь розробленого сценарію і певних результатів, тестувальник імпровізує і годиться на здоровий глузд, свій досвід і знання продукту).

Особливості тестування веб-сайтів

Тестування веб-сайтів дозволяє перевірити працездатність усіх функцій відповідно до технічного завдання, швидкість і стабільність роботи, читабельність інформації, зручність переходів по сайту, захищеність від веб-атак та багато іншого.

Виділяють такі основні види робіт при тестуванні веб-сайтів:

– відповідність фактичного дизайну затвердженому завданню;
– перевірка верстки на валідність;
– функціональне тестування;
– навантажувальне тестування;
– тестування безпеки сайту;
– тестування сумісності.

Крім того при тестування сайту можна провести пошук помилок у контенті і перевірку його на унікальність, тестування швидкості завантаження сайту, юззабіліті, відповідність мета-тегів тощо.

Вибір підходу тестування веб-сайтів

Виходячи з вище перерахованих висновків, для максимально ефективного тестування веб-сайту, який знаходиться у використанні, було обране автоматизоване тестування.

За основні завдання тестування веб-сайту прийняті:

– створення якісного фреймворка, який можна буде підтримувати в перспективі майбутнього;

– наповнення фрейморка максимально можливою кількістю описаних сторінок і тестів;

– виявлення дефектів для подальшого їх виправлення командою розробників;

– надання інформації про якість програмного забезпечення.

Автоматизоване тестування було обрано у зв'язку з тим що, після виправлення знайдених дефектів необхідно буде повторно використовувати регресійне тестування для перевірки того, що дефекти були дійсно виправлені.

В перспективі створений фреймворк можна буде розширювати і доповнювати новими тестами після внесення в програмне забезпечення змін, для тестування нового функціоналу.

ЛІТЕРАТУРА

1. Куликов С.С. Тестування програмного забезпечення. Базовий курс / С.С. Куликов. – Мінськ : Чотири чверті, 2017. – 312 с.
2. Винниченко І.В. Автоматизація процесів тестування / І.В. Винниченко. – М. : Пітер, 2005. – 204 с.
3. Роберт Мартін. Чистий код. Створення, аналіз і рефакторинг / Мартін Роберт. – М. : Пітер, 2019. – 465 с.
4. Грабко О. Особенности тестирования веб-приложений / О. Грабко // URL: https://quality-lab.ru/blog/key-principles-of-web-testing/ (останнє звернення 06.04.2021)
5. Ручное и автоматизированное тестирование: как выбрать эффективный подход // URL: https://www.a1qa.ru/blog/ruchnoe-i-avtomatizirovannoe-testirovanie-kak-vybrat-effektivnyj-podhod/ (останнє звернення 04.04.2021)
6. Модели и методологии разработки ПО // URL: https://gb.ru/posts/methodologies (останнє звернення 05.04.2021)
7. Заметки Автоматизатора // URL: https://automation-remarks.com/ (останнє звернення 04.04.2021)

ТУРЕНКО Михайло Сергійович – студент доцент кафедри прикладної та теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: xa11867700@student.karazin.ua; ORCID: 0000-0001-6875-515X.

Наукові інтереси:

– *методи тестування програмного забезпечення;*

СТРІЛЕЦЬ Вікторія Євгенівна – к.т.н.; доцент кафедри прикладної та теоретичної системотехніки; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: viktoria.strilets@karazin.ua; ORCID: 0000-0002-2475-1496.

Наукові інтереси:

– *методи машинного навчання;*

– *методи штучного інтелекту.*

УДК 004.415.2 **СЕКЦІЯ 2,7**

УМАНЕЦЬ К.В.

ПРОБЛЕМИ РАНЖИРУВАННЯ РЕЛЕВАНТНИХ РЕЗУЛЬТАТІВ, ЩО ПОВЕРТАЮТЬСЯ ПО РІЗНИХ ПОШУКОВИХ ЗАПИТАХ

Вступ

У 1960-і роки дослідники тестували пошукові системи приблизно на 1,5 мегабайтах текстових даних. Тепер у нас є мільярди веб-сторінок і колосальні дані, а пошукові системи все ще далекі від досконалості. Вони стикаються з серйозними проблемами, намагаючись розшифрувати наші наміри і знайти сторінки, які найкраще відповідають нашим запитам. Для вирішення цих проблем, дослідники пропонують моделі пошуку.

Модель пошуку – це формальне подання процесу зіставлення запиту і документа. Це основа алгоритму ранжирування, яка використовується в пошуковій системі для створення ранжируваного списку документів. Хороша модель пошуку знайде документи, які можуть бути полічені релевантними для людини, яка відправила запит. Тобто документи будуть відповідати інтенту (пошуковому наміру), закладеному в запиті, і видачі в пошуковій системі, отриманої в результаті цього запиту [1]. Деякі моделі пошуку фокусуються на актуальній релевантності, але пошукова машина, розгорнута в реальному середовищі, повинна використовувати алгоритми ранжирування, які включають релевантність для користувачів.

Проблеми оцінки IR моделі

Сиріл Клевердон в шістдесяті роки розробив модель IR (Information retrieval), яка і до цього дня використовується дослідниками [2]. Оцінка IR на сьогоднішній день є проблемою, оскільки рейтинг залежить від того, наскільки добре він відповідає очікуванням користувачів. Основними показниками такої оцінки є точність (precision) і повнота (recall). Вони фокусуються на порівнянні отриманого набору результатів на основі ранжирування, що визначається вручну або за допомогою неявного зворотного зв'язку з даними про поведінку користувачів.

Метрики для невпорядкованої безлічі документів засновані на бінарній класифікації документів «релевантний/не релевантний» стосовно вибраного запиту [3]. Дані метрики ґрунтуються на матриці класифікації:

Таблиця 1 Основні категорії документів відповіді системи

	релевантні	не релевантні
знайдено системою	a	b
не знайдено системою	c	d

де: a – кількість документів, знайдених системою і релевантних з точки зору експертів;

b – кількість документів, знайдених системою, але не релевантних з точки зору експертів;

c – кількість релевантних документів, що не знайдені системою;

d – кількість нерелевантних документів, що не знайдені системою.

Повнота – це міра того, наскільки добре пошукова система повертає потрібні користувачеві документи, але не враховує кількість нерелевантних документів, які йому видаються. Чим більше релевантних результатів не вистачає в пошуку, тим менше ймовірність повноти. При використанні повноти передбачається, що всі відповідні документи для даного запиту відомі, а обчислюється вона як відношення знайдених релевантних документів до загальної кількості релевантних документів:

$$r = \frac{a}{a + c}, \quad (1)$$

© УМАНЕЦЬ К.В., 2021

Наприклад, якщо повнота дорівнює 50%, то це означає, що половина релевантних документів системою не знайдено.

Точність – це міра того, наскільки ефективно пошукова система видає в списку результатів тільки релевантні документи. Чим більше нерелевантних результатів, тим нижче точність. Обчислюється як відношення знайдених релевантних документів до загальної кількості знайдених документів:

$$p = \frac{a}{a+b}, \quad (2)$$

Наприклад, якщо точність дорівнює 50%, то це означає, що серед знайдених документів половина релевантних і половина – нерелевантних.

В ідеалі пошукова система повинна ідентифікувати всі релевантні документи та не повертати нерелевантні (100% точність і 100% повноти). На практиці це виявилося неможливим, оскільки точність і повнота обернено пропорційні [4].

На щастя, більшість пошукових систем більше турбує точність, особливо в першій десятці результатів. Мало хто з нас продовжує пошук після першої пари сторінок результатів пошуку.

Недоліки метрики на послідовних документах

Метрики для списку документів, відсортованого за релевантністю, враховують не тільки факт наявності документа в списку знайдених документів, а й його місце в цьому списку. Впорядкування списку результатів відбувається таким чином, щоб записи, які, швидше за все, зацікавлять користувача, були попереду. Це спрощує пошук для користувачів, оскільки їм не доведеться витрачати багато часу на пошук та перегляд записів, які зацікавлять їх.

Точність на рівні n документів визначається як кількість релевантних документів серед перших n виданих документів, поділене на n [3].

Якщо система видала більше n документів, то ця величина дорівнює точності системи на перших n документах результатів запиту. Якщо система видала менш n документів, то точність на рівні n документів буде свідомо не вище точності системи.

Точність на рівні n документів характеризує здатність системи видавати релевантні документи на початку списку результатів. Наприклад, якщо система видає не більше 10 документів на першій сторінці, то precision (10) відображає якість результатів системи, одержаних на першій сторінці.

Ця метрика має ряд недоліків. Зокрема, для різних запитів метрики precision (n) можуть бути непорівнянні. Наприклад, для «ідеальної» системи, яка видає тільки релевантні документи, precision (100) = 0.2 для запиту, за яким існує 20 релевантних документів, і precision (100) = 0.3 для запиту, за яким існує 30 релевантних документів. Незважаючи на певні недоліки, точність на рівні до сих пір є незамінною метрикою сучасних систем пошуку так як, зокрема, дозволяє оцінити корисність першої сторінки відповіді системи для користувача.

Спам

Фактично, спам також є однією з проблем, що впливають на результати пошуку [5]. Спам в контексті IR – це невідповідна або нерелевантна інформація в документі, що вводить в оману та призначена для комерційної вигоди. Спам настільки важливий в веб-пошуку, що ціла тема, яка називається змагальним пошуком інформації, була розроблена для роботи з методами пошуку колекцій документів, якими маніпулюють сторони з різними інтересами.

Якщо алгоритми для перших пошукових машин був сконцентрований на пошуку за ключовими словами та частоту оновлення то, у зв'язку з підвищенням активності використання цих методів для їх омани, змусило більш пізні версії пошукових алгоритмів ігнорувати ці вказівки та критично ставитись до кожної знайденої сторінки сайту, що спровокувало ослаблення пошуку сторінок з рідкісним та якісним контентом, з заданими ключовими словами.

Висновок

Традиційними показниками ефективності пошуку є повнота (повнота пошуку) і точність (чистота пошуку). Емпіричні дослідження ефективності пошуку показали тенденцію до зниження точності по мірі збільшення повноти.

Головна проблема, породжувана спамом, полягає в тому, що з'являється величезна кількість недоброякісного контенту, що не несе жодної цінності, ускладнюючи при цьому ефективність роботи пошукових машин. Що не менш важливо, він робить релевантність пошукових результатів та ранжування інтернет-ресурсів неточними. У підсумку це знецінює мережу Інтернет як основне джерело отримання правдивої інформації.

ЛІТЕРАТУРА

1. Relevance (information retrieval) [Електронний ресурс] // Wikipedia, the free encyclopedia – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Relevance_(information_retrieval).
2. Information retrieval [Електронний ресурс] // Wikipedia, the free encyclopedia – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Information_retrieval.
3. Агеев М. Официальные метрики РОМИП’2004 [Електронний ресурс] / М. Агеев, И. Кураленок – Режим доступу до ресурсу: http://romip.ru/docs/romip_metrics.pdf.
4. Buckland M. The Trade-off between Recall and Precision [Електронний ресурс] / Michael Buckland – Режим доступу до ресурсу: https://www2.sims.berkeley.edu/research/projects/oasis/trade.html.
5. Buckland M. The Trade-off between Recall and Precision [Електронний ресурс] / Michael Buckland – Режим доступу до ресурсу: https://www2.sims.berkeley.edu/research/projects/oasis/trade.html.

УМАНЕЦЬ Катерина Віталіївна – студентка кафедри теоретичної та прикладної системотехніки, факультет комп’ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків, Україна, 61022; e-mail: xa11867701@student.karazin.ua; xa11867701@student.karazin.ua; ORCID: 0000-0002-2201-9074.

Наукові інтереси:

– *оцінки якості роботи пошукової системи, засновані на аналізі результатів роботи системи*

УДК 593.3 **СЕКЦІЯ 1**

УСАТОВА О.О.

МОДЕЛЮВАННЯ РУХУ РІДИНИ В ЦИЛІНДРИЧНИХ ОБОЛОНКАХ

Вступ

В роботі розглянуто рух в'язкої нестисливої рідини в пружній циліндричній оболонці. Рівняння руху пружної оболонки за відсутністю зовнішніх збурень отримано на основі принципу Остроградського – Гамільтона. За припущеннями наявності течії Хагена – Пуазейля отримано вираз для кінетичної енергії рухомої рідини в нанотрубці та побудовані рівняння для визначення частот коливань трубки з рідиною, що дає змогу дослідити стійкість руху. Отримані числові значення частот коливань пружної циліндричної оболонки без урахування наявності рідини, а також з рідиною при різних значеннях перепаду тиску.

Рух рідини – це явище, яке притаманне багатьом технічним процесам. Відомо, що нерухома рідина в сосуді згідно закону Паскаля передає зовнішній тиск за усіма напрямками, і усі точки об'єму рідкого середовища знаходяться в незмінних та однакових умовах. Однак, коли рідина протікає по трубі без тертя, то площа поперечного перерізу на різних ділянках відрізняється. Тиск виявляється неоднаковим вздовж труби [1]. Існує ефективний спосіб визначення руху рідини в трубках, заснований на використанні формули Пуазейля. Згідно з рівнянням Бернулі, при стаціонарному русі рідини по прямолінійній горизонтальній трубці постійного перерізу, тиск рідини має бути однаковим по усій довжині трубки. Але в дійсності, тиск рідини падає в напрямку її руху. Для забезпечення стаціонарності необхідно підтримувати на кінці трубки постійну різницю тисків, урівноважуючи сили внутрішнього тертя, які виникають при русі рідини [2, 3], це й дозволяє забезпечити підхід Пуазейля. Рух рідини – це явище, яке притаманне багатьом технічним процесам. Відомо, що нерухома рідина в сосуді згідно закону Паскаля передає зовнішній тиск за усіма напрямками, і усі точки об'єму рідкого середовища знаходяться в незмінних та однакових умовах. Однак, коли рідина протікає по трубі без тертя, то площа поперечного перерізу на різних ділянках відрізняється. Тиск виявляється неоднаковим вздовж труби [1]. Існує ефективний спосіб визначення руху рідини в трубках, заснований на використанні формули Пуазейля. Згідно з рівнянням Бернулі, при стаціонарному русі рідини по прямолінійній горизонтальній трубці постійного перерізу, тиск рідини має бути однаковим по усій довжині трубки. Але в дійсності, тиск рідини падає в напрямку її руху. Для забезпечення стаціонарності необхідно підтримувати на кінці трубки постійну різницю тисків, урівноважуючи сили внутрішнього тертя, які виникають при русі рідини [2, 3], це й дозволяє забезпечити підхід Пуазейля.

Моделювання течії Пуазейля. Стан питання

Особливості течії Пуазейля в мікротрубках, коли молекули у твердій стінці і атоми стінки при кінцевій температурі стінки піддані хаотичному руху, полягають в тому, що в проміжному інтервалі чисел Кнудсена спостерігається ковзання частинок рідини у стінки. У роботі американських авторів E.Lauga і ін.[4] описані три можливі випадки:

1) рідина може бути нерухома (немає прослизання);

2) ковзає щодо стінки (потік з прослизанням);

3) реалізується профіль потоку, коли тертя об стінки повністю відсутнє (повне прослизання).

Згідно граничній умові Нав'є, величина швидкості прослизання рідини, пропорційна градієнту швидкості на стінці:

$$\vartheta\big|_{z=R} = L_s \left.\frac{d\vartheta}{dz}\right|_{z=R} \qquad (1)$$

Внутрішня поверхня нанотрубки є гладкою на різних масштабах, навіть на атомному. Але макроскопічні поверхні пористих матеріалів мають шоркості на значно менших

© УСАТОВА О.О., 2021

масштабах. Тож газ, який розповсюджується по внутрішній порожнині нантрубки зазнає суттєво менший опір з боку поверхні. Також нанотрубки можуть пропускати не лише газоподібні, а й рідкі речовини. В класичній гідродинаміці, не враховується атомарна (молекулярна) будова рідини, та не надається опис руху рідини в наноканалах з шириною порядку 10 нм і менших молекулярних діаметрів. Однією з головних проблем є те, що нанотехнологія оперує такими величинами, на які закони класичної фізики не розповсюджуються.

Дослідження потоку рідини через мікро і нано трубках представляє фундаментальний інтерес для багатьох біологічних і технічних пристроїв і систем. У зв'язку з цим, в даний час інтенсивно досліджуються течія в каналах нанометрового розміру. На першому етапі треба дослідити можливості застосування класичних моделей оболонок.

Формулювання задачі та метод розв’язання

Оскільки рівняння гідродинаміки є нелінійними, навіть при постійних коефіцієнтах, унаслідок того, що в інерційні члени рівняння входить квадрат швидкості, точних розв’язків відомо відносно мало. Тому потрібно знайти такі наближені рішення, які були б набором базових течій, що присутні в технологічних процесах.

Використовуючи формулу Хагена – Пуазейля (2), можна знайти середню по перерізу швидкість у трубі

$$U_{сер} = \frac{G}{\pi R^2} \qquad (2)$$

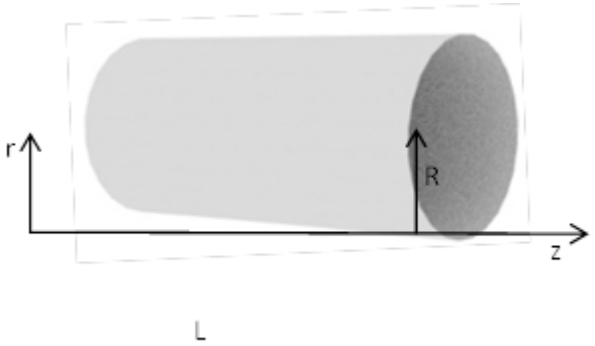


Рис. 1 Система координат для течії Хагена – Пуазейля

Тут представлена слоїста течія з єдиною відмінною від нуля z- компонентою швидкості, а саме

$$U_z = U \neq 0 \qquad (3)$$

Характерна ознака слоїстої течії – наявність лише однієї, відмінної від нуля компоненти швидкості. Це означає, що всі частинки рідини рухаються за траєкторіями, які повторюють одну з координатних ліній [5,6,7].

Рівняння руху пружної оболонки за відсутністю зовнішніх збурень, може бути отримано на основі принципу Остроградського – Гамільтона:

$$\int_{t_1}^{t_2} \left(\delta\Pi - \delta T\right) dt = 0 \qquad (4)$$

де П, Т- потенційна та кінетична енергії.

Аналіз числових результатів

Розглянуто пружну циліндричну оболонку з такими геометричними та фізичними параметрами: R=0.05м, L=3м, μ=1.004·10-3Па·с, Δp=0.5Па, h=0.005м, модуль Юнга E=20000Мпа, коефіцієнт Пуассона υ=0.3, pf =1000кг/м3 , ps=7800кг/м3. Вважаємо, що оболонка жорстко закріплена при z =0.

В таблиці 4.1 наведені значення перших п’яти частот коливань для пружної порожньої оболонки, коливань оболонки з урахуванням приєднаних мас рідини (при $\Delta p=0$) та при $\Delta p=0.5$Па за наявності рухомої рідини.

Таблиця 4.1. Частоти коливань оболонки

ρ_f, *кг/м*^{3}	Δp,Pa	n=1	n=2	n=3	n=4	n=5
0	0	261.699	422.243	785.098	1266.42	1308.50
1000	0	181.964	293.579	545.895	880.595	909.827
1000	0.5	(0,181.964i)	(-1.8580, 293.579i)	(0,545.895i)	(-1.3854, 880.595i)	(0,909.827i)

З наведених даних бачимо, що наявність рухомої рідини веде до появи комплексних частот. Дійсні частини цих частот є від’ємними, тобто відбувається демпфування коливань. Наявність рідини веде до зменшення частот коливань оболонки.

Висновки

В роботі розглянуто рух в’язкої нестисливої рідини в пружній циліндричній оболонці. Побудовані вирази для кінематичної та потенціальної енергії. Отримані рівняння для визначення частот коливань рідини за припущеннями течії Хагена-Пуазейля. Визначені частоти коливань оболонки з урахуванням наявності рухомої рідини. Розроблено підхід до дослідження стійкості руху пружних трубок з рідиною.

ЛІТЕРАТУРА

1. Сивухин Д.В. Общий курс физики. Т. I. — М.: Наука, 1996. Гл. XII, §§ 93, 94, 95. https://learn.ztu.edu.ua/pluginfile.php/93163/mod_resource/content/1/Sivuhin_D.V._Kurs_obwej_fiziki._Tom_2-arpg75zl9bj.pdf
2. Хайкин С.Э. Физические основы механики. — М.: Наука, 1971. Гл. XVI, §§ 123, 124. https://www.twirpx.com/file/1922952/
3. Стрелков С.П. Механика. — М.: Наука, 1975. §§ 100–106. https://ikfia.ysn.ru/wp-content/uploads/2018/01/Strelkov1975ru.pdf
4. Eric Lauga,Willow R. DiLuzio, George M. Whitesides, and Howard A. Stone. Swimming in Circles: Motion of Bacteria near Solid Boundaries. Biophys J. 2006 Jan 15; 90(2): 400–412. doi: 10.1529/biophysj.105.069401 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1367047/
5. Байков В.И., Павлюкевич Н. В., Федотов А.К., Шнип А.И. Минск: Институт тепло- и массообмена имени А. В. Лыкова НАН Беларуси, 2014. – 370 с. https://www.itmo.by/doc/publication/full_teplofiz2.pdf
6. Еселева Е.В., Гнитько В.И., Стрельникова Е.А. Собственные колебания сосудов высокого давления при взаимодействии с жидкостью. Пробл. машиностроения. Т. 9. №1, 2006, С.105-118. http://mia.univer.kharkov.ua/16/30224.pdf
7. Medvedovskaya T. Free hydroelastic vibrations of hydroturbine head covers/ Medvedovskaya T., Strelnikova E., Medvedyeva K // Int. J. Eng. and Advanced Research Technology.– 2015.– Vol. 1.– No 1.– P.45-50. DOI: 10.13140/RG.2.1.3527.4961 https://journal-me.com/archive-ukr/vol21-2018-iss1-paper5/?print=print

УСАТОВА Ольга Олександрівна – аспірант 3-го року навчання, Інститут проблем машинобудування ім. А.М. Підгорного НАН України, вулиця Пожарського, 2/10, Харків, 61046, Україна, ORCID 0000-0001-1267-2723, e-mail: usatova.olia@gmail.com.

Наукові інтереси:
– *комп’ютерне моделювання руху рідини;*
– *рух рідини в трубках і нанотрубках.*

УДК 004.7(075) **СЕКЦІЯ 2**

ШАРОВ В.О., БЕРДНІКОВ А. Г.

МОДЕЛЮВАННЯ КОРИГУВАЛЬНОГО КАСКАДНОГО КОДУ В КАНАЛАХ ПЕРЕДАЧІ ДАНИХ СИСТЕМИ УПРАВЛІННЯ

Вступ

В сучасних системах управління широко застосовуються інтегральні цифрові мережі (Integrated Services Digital Network - ISDN), які забезпечують передачу всіх видів дискретної інформації в стандартному форматі.

Як приклади мереж з інтегрованим обслуговуванням можна привести універсальні протоколи ATM (Asynchronous Transfer Mode - асинхронний режим передачі) або Broadband ISDN (B-ISDN), які забезпечують передачу будь-якого типу трафіку, як комп'ютерного, так і мультимедійного.

Технологія ATM використовується на багатьох магістралях найбільших операторів зв'язку підтримує всі види трафіку, може використовуватися безпосередньо прикладним рівнем протоколів і працювати автономно без протоколів TCP / IP і TCP / UDP, однак наявність досить складного власного протоколу маршрутизації призвело до того, що в сучасних комп'ютерних мережах для об'єднання мереж використовуються, як правило, протоколи TCP / IP і TCP / UDP, але ATM залишається однією з технологій, на основі якої працюють багато локальних мереж, що утворюють складну складену мережу.

Отже, вдосконалення і розвиток протоколу з метою його подальшого застосування і вивчення представляє інтерес, і розробка моделі перешкодостійкого каналу передачі даних, використовуваного в інтегральної мережі, представляється актуальною.

Постановка задачі

Для того, щоб почати розгляд моделі передачі, необхідно розцінити, по-перше, доцільність та головну перешкоду достовірної передачі інформації. Основною перепоною для достовірності передачі є перекручування розрядів, в результаті якого, при декодуванні отриманого повідомлення отримується не та інформація, яка надсилалася і була закодована. У каналах без пам'яті, тобто у каналах, у яких вірогідність появи символу на виході з каналу залежить тільки від символу на вході до каналу, дуже легко вирахувати вірогідність отримання будь-якої послідовності символів на виході, при заданій послідовності на вході, дуже легко можна підрахувати вірогідність перекручування n розрядів(1).

$$P_k = C_n^k \cdot P_{\text{пом}}^k \cdot (1 - P_{\text{пом}})^{n-k}, \quad (1)$$

де P_k - це ймовірність k-тої помилки в кодової комбінації, n -кількість розрядів у повідомленні, C - кількість поєднань.

В таблиці 1.1 розташовані вірогідності перекручувань з базовою вірогідністю перекручування у каналі 10^{-3},[1] що за сучасними мірками є дуже поганим показником.

Таблиця 1
Результати розрахунку по формулі 1

Загальна кількість розрядів у комбінації.	Кількість перекручених розрядів.	Вірогідність перекручування на виході.
8	1	0,0079
8	2	0,00002
8	3	0,00000005
16	1	0,0157
16	2	0,0001
16	3	0,0000008

Задля наочності наведено графік на рисунку 1

© ШАРОВ В.О., БЕРДНІКОВ А. Г., 2021

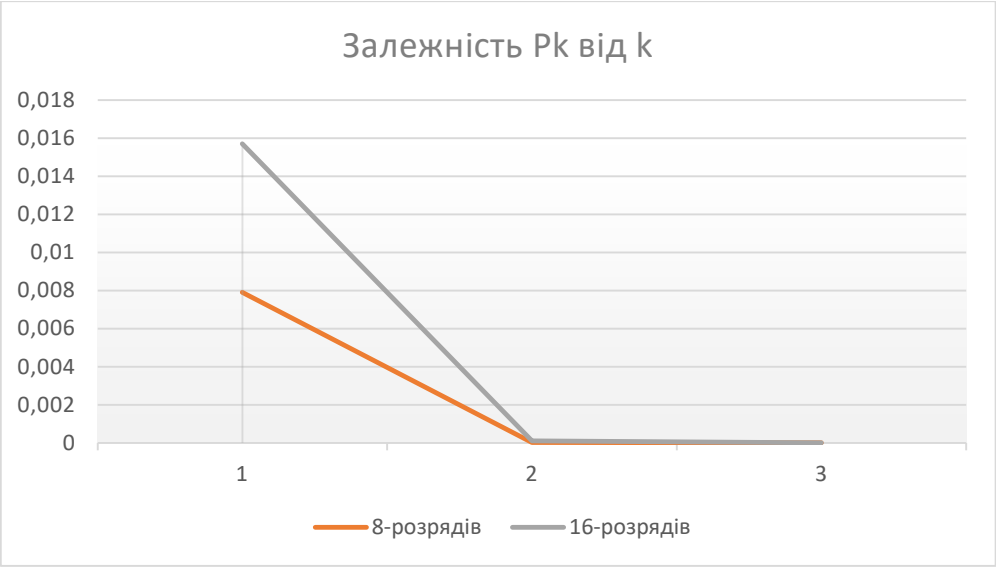


Рис. 1 Вірогідності перекручування розрядів

Як можна побачити, основна частина перекручувань припадає на 1-кратні та 2-кратні помилки. Оскільки у сучасних каналах передачі базова вірогідність значно менша[1], ніж у розглянутому прикладі, помилками кратності більше 2 можна знехтувати.

Каскадні коди

Для боротьби з перекручуваннями використовують багато методів, наприклад: підвищення відношення сигнал/завада, використання зворотного зв'язку, але у даній роботі увага зосереджується на методі використання завадостійких кодів. Завадостійким називають той код, який здатний, у межах свого повідомлення, виявляти / виправляти помилки, які трапились при передачі.

Завадостійких кодів розрізняється дуже багато – усі вони мають право на реалізацію, залежно від умов, які ставляться до систем.

Основною перевагою каскадних кодів є відносна невелика складність їх технічної реалізації та можливість виправлення пакетів помилок, що сильно економить час при передачі повідомлень. Використання каскадних кодів може бути різноманітним. Наприклад, можлива реалізація каскадного коду, де внутрішній код є кодом Хеммінга (7/4), а у зовнішньому коді йде перевірка на парність.

Для кращого уявлення наведено графічний приклад на рис. 2:

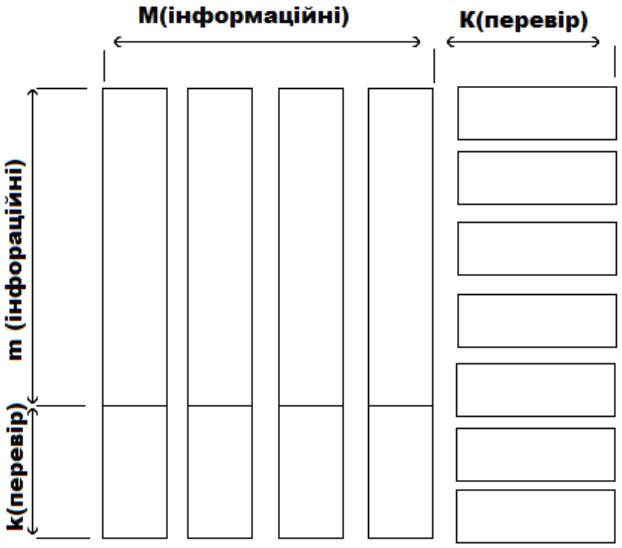


Рис. 2 Загальна схема каскадного кодування

де m - інформаційні біти внутрішнього коду, k- перевірочні біти внутрішнього коду, M - інформаційні біти зовнішнього коду, K - перевірочні біти зовнішнього коду.

Виходячи з простоти реалізації, каскадні коди є логічним продовженням використання завадостійкого кодування, адже вони не просто виявляють/виправляють помилки у повідомленнях, а також борються с пакетами помилок.

Оскільки поєднань кодів може бути дуже багато, з'являється питання вибору оптимального коду. Зазвичай, вибір кодів суворо залежить від задач і системи, у котрих вони будуть використовуватись. Виходячи з розглянутого аналізу помилок, де основними є помилки однократні та двократні, для прикладу розглянемо код, який може виправляти однократні помилки, а також є простим у реалізації та легко може досліджуватись.

Суть моделі

За основу першого ступеня було обрано код Хеммінга, який здатний виправляти однократну помилку.

Щодо розмірності, можна обирати повідомлення будь-якої довжини, якщо воно задовільняє формулі (максимальна кількість інформаційних символів у повідомлені для виправлення однократної помилки.)

$$2^m \leq \frac{2^n}{n+1} \quad , \tag{2}$$

де m -кількість інформаційних символів, n - загальна кількість символів у повідомленні.

Оскільки можна отримати точну кількість інформаційних та перевірочних розрядів у комбінації, необхідно також вирішити на які самі місця необхідно становити перевірочні розряди.

Загальний алгоритм кодування кодом Хеммінга описується наступним чином: у коді Хеммінга кожен з k перевірочних символів утворюється в результаті перевірки на парність певного сполучення інформаційних m символів (число перевірок дорівнює k). Правило розстановки можна описати правилом, що перевірочний розряд має бути на z-тому місці та контролювати z через z розрядів. Сума охвачених однією перевіркою символів завжди є парне число k_b:

$$m_u + m_h + \cdots + m_o = k_b, \tag{3}$$

При прийомі комбінація також підпадає під k перевірок, виходячи з чого кожна перевірка контролює певне сполучення інформаційних символів і також відповідний перевірочний символ.

$$m_u + m_h + \cdots + m_o + k_b = \mathrm{c}_y \tag{4}$$

Саме тому, якщо $\mathrm{c}_y = 0$, можна робити висновок, що при передачі помилки не трапилось, при $\mathrm{c}_y \neq 0$, при передачі сталось перекручування розряду.

Для спрощення, першим каскадом(ступенем) буде код Хеммінга (15/11), а другим ступенем буде код Хеммінга (16/11), додаючи біт перевірки на парність, що дозволить виявляти двократну помилку. (15/11) - означає, що при загальних 15 бітах у повідомленні, 11 бітів є інформаційними (І), а 4 перевірочними (П).

Таблиця 2

Загальна розстановка у кодах Хеммінга, де стовпчики - № розряду, а строки - № перевірки

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	X		X		X		X		X		X		X		X
2		X	X			X	X			X	X			X	X
3				X	X	X	X					X	X	X	X

4								X	X	X	X	X	X	X	X

По призначенню у кодовій комбінації розряди будуть розташовуватись як у табл.3.

Таблиця 3
Розстановка по позиціям у коді Хеммінга (15/11)

Номер позиції	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Призначення позиії	п1	п2	і1	п3	і2	і3	і4	п4	і5	і6	і7	і8	і9	і10	і11

Система породжуючих рівнянь для перевірочних розрядів має вигляд:

п1=і1⊕ і2⊕ і4⊕ і5⊕ і7⊕ і9⊕ і11;

п2=і1⊕ і3⊕ і4⊕ і6⊕ і7⊕ і10⊕ і11;

п3=і2⊕ і3⊕ і4⊕ і8⊕ і9⊕ і10⊕ і11;

п4=і5⊕ і6⊕ і7⊕ і8⊕ і9⊕ і10⊕ і11

Де"⊕ " – підсумовування за модулем 2.

Для перевірки у коді Хеммінга (15/11) також застосовується матриця (табл.2), де "Х" - номера розрядів, які підсумовуються за модулем 2.

У такому випадку, система перевірочних рівнянь виглядає наступним виглядом:

1к.р.= р.к.1⊕ р.к.3⊕ р.к.5⊕ р.к.7⊕ р.к.9⊕ р.к.11⊕ р.к.13⊕ р.к.15⊕ ;

2к.р.= р.к.2⊕ р.к.3⊕ р.к.6⊕ р.к.7⊕ р.к.10⊕ р.к.11⊕ р.к.14⊕ р.к.15⊕ ;

3к.р.= р.к.4⊕ р.к.5⊕ р.к.6⊕ р.к.7⊕ р.к.12⊕ р.к.13⊕ р.к.14⊕ р.к.15⊕ ;

4к.р.= р.к.8⊕ р.к.9⊕ р.к.10⊕ р.к.11⊕ р.к.12⊕ р.к.13⊕ р.к.14⊕ р.к.15⊕ ;

де "к.р."– контрольний розряд, а "р.к.(q)" – просто q-тий розряд закодованої за допомогою таблиці 2 комбінації.

Таким чином, Утворюється двійкова комбінація з трьох перевірочних біт. Наразі, якщо результат перевірки не дорівнює "0000", то можна зробити висновок, що у комбінації є помилка, причому номер розряду з помилкою і буде результатом. Наприклад: результат "001" свідчить про помилку у першому розряді.

Задля того, щоб код мав змогу виявляти двократні помилки, необхідно до коду (15/11) додати біт перевірки на парність (8/4). Породжуюче рівняння має вигляд:

п4=п1⊕ п2⊕ і1⊕ п3⊕ і2⊕ і3⊕ і4⊕ п4⊕ і5⊕ і6⊕ і7⊕ і8⊕ і9⊕ і10⊕ і11.

Використовуючи дану модель, у випадку, коли перевірка на парність помилки не виявила, але перевірочна комбінація не дорівнює "0000", можна робити висновок, що сталося двократна помилка. Якщо перевірка на парність виявила помилку, можна зробити висновок,що трапилась однократна помилка, відбросити 16-й розряд і при декодуванні отримати розряд помилки.

Після кодування, до коду Хеммінга (15/4) стає можливо, також, каскадом додати не біт перевірки на парність, а циклічний код. Кодування циклічним кодом подібне до кодування кодом Хеммінга, більш детально алгоритм кодування розглянутий у джерелі [1,2]. При кодуванні циклічним кодом з'явиться модель схожа на вже розглянутий рис.2. В результаті кодування циклічним кодом, за допомогою породжуючого полінома буде отримана кодова комбінація, яка також здатна виправляти однократні помилки, або, в залежності від кількості перевірочних розрядів, виявляти пакетів помилок. Циклічний код є дуже зручним, оскільки для будь-яких цілих чисел m,t існує код значності

$$n = 2^m - 1, \tag{5}$$

де n-загальна кількість розрядів, з кодовою відстанню d

$$d \geq 2t + 1, \tag{6}$$

при цьому кількість перевірочних символів r не перевищує величину mt

$$r \leq mt, \tag{7}$$

Такий код гарантовано виправляє усі помилки кратності t, а також виявляє усі помилки кратності 2t, а також виявляє усі пакети помилок завдовжки r, адже при декодуванні, якщо при діленні отриманого полінома на утворюючий поліном виникає помилка (синдром), можна буде точно дізнатись номер розряду, у котрому ця помилка трапилась, адже синдром для кожної помилки, при певному поліномі, буде унікальним.[5]

Висновок

Розглянута модель є окремим випадком використання каскадних кодів, яка демонструє доцільну простоту використання і практичну цінність даного типу кодування, що є базою для подального розширення дослідження по даній темі.

ЛІТЕРАТУРА

1. ВВЕДЕНИЕ В ТЕОРИЮ ПОМЕХОУСТОЙЧИВОГО КОДИРОВАНИЯ: Составители: Давыдов А.В., Мальцев А.А. Учебно-методические материалы для магистрантов и аспирантов Исследовательской школы «Колебательноволновые процессы в природных и искусственных средах». – Нижний Новгород: Нижегородский госуниверситет, 2014. – 123 с.
2. Лосев Ю. І. Основи теорії інформації / Ю. І. Лосев, С. І. Шматков: навчальний посібник. – Х.:ХНУ імені В. Н. Каразіна, 2009. – 126 с
3. Завадостійке кодування, навчальний посібник URL: https://scask.ru/h_book_codb.php?id=42 (Last accessed: 07.04.2021).
4. Олифер В., Олифер Н. Компьютерные сети. Принципы, технологии, протоколы : Учебник для вузов. Питер: 5-е изд. СПб, 2016. 992 с.
5. Лосев Ю. І., Руккас К. М., Шматков С. І. Комп’ютерні мережі: навч. посіб. / за редакцією Ю. І. Лосева. Харків : ХНУ імені В. Н. Каразіна, 2013. 248 с.

ШАРОВ Владислав Олегович – студент групи КУ-51 факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободі, 4, Харків-22, Україна, 61022; e-mail: WyCTPiY@gmail.com; ORCID: 0000-0003-3152-0650.

Наукові інтереси:

- Програмування;
- Управління проектами.

УДК 004.75 **СЕКЦІЯ 3, 7**

ШВИДКИЙ Ю.К. СПОРОВ О.Є. ГУЩІН І.В.

ДОСЛІДЖЕННЯ КОМПРОМІСІВ ПРИ ПРОЕКТУВАННІ РОЗПОДІЛЕНИХ СИСТЕМ ІЗ ВИКОРИСТАННЯМ CAP ТЕОРЕМИ

Вступ

На даний час, коли все більше використовуються Big Data, компанії, що активно розвиваються, стикаються з проблемами розширення інформаційних систем. Розширення може відбуватись як в горизонтальному, так і в вертикальному напрямках. Після певного порогу горизонтальне масштабування стає набагато ефективніше та дешевше, ніж вертикальне [1]. Якщо система має класичну 3-х рівневу архітектуру, то в сучасних реаліях масштабування компонентів із клієнтською та бізнес-логікою буде доволі нескладним; за умов правильного проектування системи відповідна зміна структури майже не буде вимагати внесення змін до програмного коду. Зазвичай основні проблеми починаються на рівні даних, якщо на початку розвитку система має не оптимальну структуру: наприклад, в найгіршому випадку, містить лише один сервер бази даних. Для більш аргументованої зміни архітектури розподіленої комп'ютерної системи в контексті великих навантажень може бути використана CAP теорема. Ця теорема є однією із фундаментальних теорем у галузі розподілених обчислень. Вона надає базове розуміння можливих компромісів при аналізі та проектуванні розподілених систем, що є дуже корисним для архітекторів-початківців. Незважаючи на те, що теорема була опублікована понад 20 років тому [2] та є доволі абстрактною, вона залишається релевантною при дизайні сучасних систем, особливо на початкових стадіях розробки. Слід зазначити, що CAP теорема може бути використана навіть при проектуванні багатоядерних процесорів [3]. В межах цієї роботи CAP теорема буде використана для аналізу архітектури Big Data систем на базі розподілених сховищ. **Мета роботи:** використати CAP теорему та її сучасні модифікації для дослідження компромісів при проектуванні розподілених систем на прикладах сучасних сховищ даних.

CAP теорема

У простому формулюванні теорема зазначає, що для розподіленого сховища даних неможливо одночасно забезпечити більше двох з наступних властивостей:

- Узгодженість (**Consistency**) – кожен успішний запит на читання отримає результат останнього запису (з будь-якого вузла в кластері). Тобто повинна бути висока ступінь узгодженості або іншими словами – **лінеарізуємість**.
- Доступність (**Availability**) – кожен вузол, що коректно працює, завжди успішно виконує запити (як на читання, так і на запис). Тобто, незалежно від часових рамок, запит обов'язково повинен завершитись – **повна доступність**.
- Стійкість до розділення (**Partition Tolerance**) – навіть якщо між вузлами немає зв'язку (мережевий розділ), вони продовжують працювати незалежно один від одного, попри довільну кількість втрат повідомлень.

Важливо зазначити, що стійкість до розділу (літера ***P***) не є властивістю, від якої можна відмовитися. У розподіленій системі завжди існує ризик мережевого розділу. Якщо це станеться, то система повинна приймати рішення або продовжувати працювати, підтримуючи узгодженість даних (літера ***C***) або припинити роботу, таким чином, компрометуючи доступність (літера ***A***). Однак немає такої речі, як стійкість до розділення в умовах збереження двох цих властивостей. В результаті, можна зробити висновок, що згідно з теоремою CAP ***при наявності розділу, розподілена система може бути або узгодженою, або доступною***.

Класифікація систем за теоремою

Теорема дозволяє ввести класифікацію систем в залежності від того, які властивості система порушує під час мережевого розділу. Так, можна виділити 2 основні категорії: **CP** та **AP**. Крім того, можлива і третя категорія – **CA**: зазвичай, це системи з одним сервером бази даних. У випадку мережевого розділу така архітектура призведе до порушення функціонування всієї

© ШВИДКИЙ Ю.К. СПОРОВ О.Є. ГУЩІН І.В., 2021

системи, тому цю категорію не використовують для створення розподілених систем. Важливо відзначити, що вибір між ***C*** та ***A*** потрібно робити лише під час мережевого розділу. У будь-який інший час, ці обидві властивості можуть бути забезпечені одночасно.

На рисунку 1 показаний мережевий розділ між двома дата-центрами під час реплікації та синхронізації даних. Наявність такого розділу змушує розробників обирати компроміси.

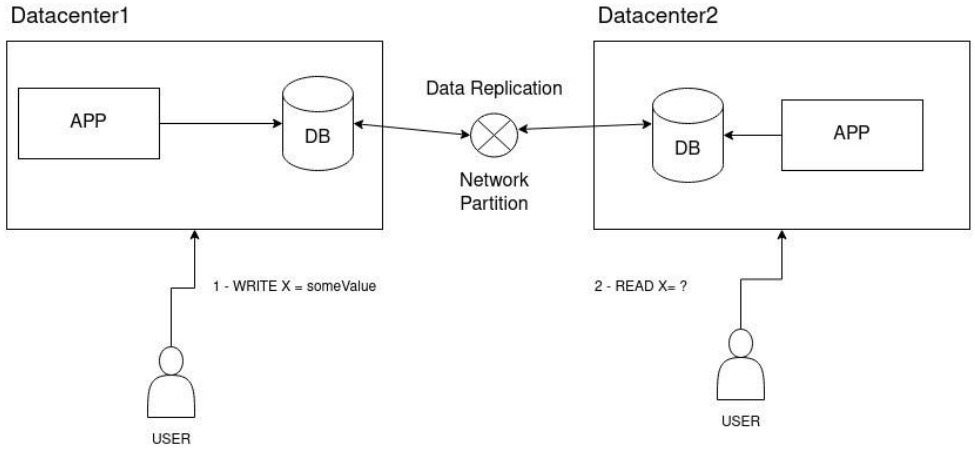


Рисунок 1. Базовий приклад розділу мережі (network partition) у розподіленій системі.

В цьому прикладі – у разі якщо система відноситься до **CP** типу, репліки не будуть консистентними допоки проблеми з мережею не будуть виправлені та повертатимуть помилку кінцевим користувачам (проблема доступності). **AP** система залишається доступною, але повертає дані у неконсистентному стані. При цьому теорема не розглядає просто **P** системи, які в практиці трапляються дуже часто. Як приклад розглянемо систему, що наведена на Рис. 1. Припустимо, що у **Datacenter1** знаходиться *Master* вузол, який приймає запити на запис, у **Datacenter2** – вузол *Standby*, який приймає запити на читання. Якщо користувач втратив зв'язок з *Master*, клієнт може читати з *Standby*, але не може писати – немає **Availability**. Таким чином, наведена система є **CP** системою, але якщо *Master* реплікує дані асинхронно, то клієнт, може зробити запит до *Standby* раніше успішної синхронізації – в такому разі втрачається і **Consistency**. Таким чином отримуємо чисту **P-систему**.

Компроміси у теоремі

У більшості сучасних розподілених систем завжди необхідно керувати компромісами, щоб задовольнити всім вимогам проекту. CAP дає добре розуміння основних проблем розвитку розподілених систем, але при проектуванні та налаштуванні **CP** та **AP** систем слід брати до уваги наступне:

- ***C*** в CAP, як вже зазначалося, розуміється як лінеарізуємість, що є найвищим принципом узгодженості [4] яка впливає на продуктивність системи та ступінь її складності. У більшості випадках непотрібна така сильна модель узгодженості. Зазвичай високий рівень ***C*** властивості потрібен у банківський або медичній сфері, де кожен байт даних може бути важливим, особливо їх порядок. Втім, для соціальних мереж цілком прийнятною є ситуація, коли у один період часу, пост може відображати 1000 лайків для одної групи користувачів та, наприклад, 1002 для іншої. Тому для таких систем можливе зменшення рівнів узгодженості. Існує багато моделей як у транзакційних [5], так і в нетранзакційних [4] системах.
- Інший підхід передбачає взаємодію **CP** та **AP** систем там, де вони дійсно потрібні – у компонентах з високими вимогами до консистенції та доступності (аналог концепції Polyglot persistence [6]). Хорошим таким прикладом може слугувати взаємодія Apache Kafka, що дає низьку затримку (**AP**) та Zookeeper, що може працювати у лінеарізуємому режимі (**CP**) [7].
- ***A*** в CAP означає поняття, яке найбільш близьке до терміну повна доступність; при цьому відсутні обмеження на затримку (latency) – при великих значення latency систему не

можна назвати доступною (наприклад, обробка одного запиту на годину). У кожного проекту можуть бути власні вимоги до рівня послуг, таким чином тут потрібно находити консенсус з замовником, щоб обрати адекватну модель доступності.

- Використання технік які зможуть підвищити доступність сховищ даних – поділ даних за ключем, налаштування мережевої та фізичної топології сховища. Як приклад розглянемо GitHub – розподілену систему, що реалізує протоколи Git. Github не можна назвати неузгодженою системою; але при цьому система є високо доступною. Для досягнення такого результату використовується багато невеликих блокуючих механізм [8], що призводять до різних компромісів.

BASE та PACELC

Через свою високу абстрактність, теорема CAP потенційно може викликати непорозуміння при проектуванні, тому для оцінки більш сучасних систем, були запропоновані альтернативні варіанти – теореми BASE та PACELC [9].

Теорема BASE вносить більш практичні компроміси та стверджує, що у реальних високонавантажених системах неможливо досягнути справжньої лінеарізуємості. Вводить наступні поняття:

- **Basic availability** – означає що система надасть відповідь на запит у будь-якому випадку, але ця відповідь може містити помилку або неузгоджені дані.
- **Soft state** – стан в якому система постійно змінюється через те що сама приводить себе до консистентного стану. Наприклад, цілком прийнято що 2-5% системи будуть у неузгодженому стані.
- **Eventually consistent** – система, в кінцевому підсумку, стане узгодженою.

Таким чином, більш сучасні сховища, наприклад як Cassandra [10], швидше відповідає BASE концепції, ніж CAP в її початковому трактуванні. Звісно, цю альтернативу не слід враховувати при проектуванні критично важливих компонентів у сфері фінансів, медицини тощо.

Теорема PACELC – розширення CAP теореми, що пропонує при проектуванні додати до розгляду затримку (L - latency). В певному сенсі низька затримка має сенс рівня доступності та уточнює CAP теорему. Саму теорему можна описати у вигляді логічного блоку псевдо-коду:

```
If there is Network Partition :
    Choose between Availability and Consistency
Else
    Choose between Latency and Consistency
```

В результаті кожна гілка цієї теореми створює 2 підкатегорії систем. Перша частина теореми визначає дві категорії, які вже були описані: **AP** та **CP**. Друга частина визначає дві нові категорії: **EL** та **EC**. Наприклад, Cassandra, за рахунок master-less моделі [10] , відноситься до категорії **AP** / **EL,** де бере у пріоритет доступність під час мережевого розділу та низькі затримки на будь-який запит при нормальній роботі.

У більшості випадків системи розробляються у відповідності до загальної вимоги, яка, зазвичай, є або продуктивність (низька затримка) та доступність, або узгодженість даних. В результаті більшість систем, потрапляють в категорії **AP** / **EL** або **CP** / **EC**.

Крім того, існують системи, які не можуть бути чітко віднесені до якоїсь з цих категорій, оскільки вони використовують різні механізми налаштування для досягнення необхідних станів. Незважаючи на це, розглянуті теореми можуть успішно використовуватись для аналізу компромісів при дизайні розподілених систем.

Висновки

В результаті проведеного дослідження показано, що проектування високонавантаженої розподіленої системи необхідно починати з аналізу компромісів. Для цього потрібно спочатку сформувати вимоги, виходячи з технічного завдання, а вже потім класифікувати систему виходячи з особливостей її налаштування. Для класифікації систем можна успішно використовувати CAP теорему, яка допомагає формалізувати основні обмеження, що накладені

на систему, та дозволяє визначити основні компроміси між доступністю та узгодженістю. Через те, що CAP теорема не описує всіх компромісів, що виникають в практичний розробці сучасних розподілених систем, можна використати сучасні її різновиди, найбільш розповсюдженні є теореми BASE та PACELC. За допомогою цих теорем можна додатково проаналізувати компроміси між затримкою, масштабованістю, безперебійною роботою, підтримкою, функціональністю, експлуатаційною простотою, в деякій мірі проблемами із багатопоточністю, та іншими аспектами системи, що присутні при проектуванні сучасних високонавантажених розподілених систем.

ЛІТЕРАТУРА

1. For Software Engineers - Scalability Introduction, URL: https://dev.to/tomerbendavid/for-software-engineers---scalability-introduction, September 2017. Дата доступу (29.03.2021)
2. Seth Gilbert and Nancy Lynch, "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services", ACM SIGACT News, Volume 33 Issue 2 (2002), pg. 51–59.
3. Hacker News: CAP theorem discussion, URL: https://news.ycombinator.com/item?id=14647233, June 2017. Дата доступу (29.03.2021)
4. Paolo Viotti, Marko Vukolic, "Consistency in Non-Transactional Distributed Storage Systems", arXiv:1512.00168v4, April 2016, pg. 4–24.
5. D. Dziuma, P. Fatourou, E. Kanellou, A. Kasra, "Survey on consistency conditions", FORTH-ICS TR 439, DECEMBER 2013, pg. 4–24.
6. Martin Fowler, Polyglot Persistence, URL: https://martinfowler.com/bliki/PolyglotPersistence.html, November 2011. Дата доступу (29.03.2021)
7. Neha Narkhede. Kafka: The Definitive Guide: Real-Time Data and Stream Processing at Scale 1st Edition. O'Reilly Media, 2017. С. 15–50.
8. Does git lock a remote for writing when a user pushes?, URL: https://stackoverflow.com/questions/52662020/does-git-lock-a-remote-for-writing-when-a-user-pushes/52668468#52668468, October 2019, Дата доступу (29.03.2021).
9. Kanstantsin Hontarau, All you didn't know about the CAP theorem https://www.slideshare.net/KonstantinGontarev/all-you-didnt-know-about-distributed-db-tradeoffs, February 2017. Дата доступу (29.03.2021)
10. Official Documentation of Apache Cassandra Documentation. URL: https://cassandra.apache.org/doc/latest/. Дата доступу (29.03.2021)

ШВИДКИЙ Юрій Костянтинович – студент кафедри штучного інтелекту та програмного забезпечення; ХНУ імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: yura.shvidkoy33@gmail.com; ORCID: 0000-0003-4473-7189.

Наукові інтереси:

– *розробка програмного забезпечення, розподілені обчислення.*

СПОРОВ Олександр Євгенович – кандидат фізико-математичних наук, доцент кафедри штучного інтелекту та програмного забезпечення; ХНУ імені В.Н. Каразіна, майдан Свободи, 6, ауд. 5-20, Тел. 707-50-76, м. Харків, Україна, 61022; e-mail: sporov@karazin.ua; ORCID: 0000-0002-4610-9656.

Наукові інтереси:

– *штучний інтелект, розподілені обчислення, операційні системи, комп'ютерне моделювання фізичних процесів.*

ГУЩІН Іван Валерійович – старший викладач кафедри штучного інтелекту і програмного забезпечення; ХНУ імені В.Н. Каразіна, майдан Свободи, 6, м. Харків, Україна, 61022; e-mail: gushchin.iv@gmail.com; ORCID: 0000-0002-1917-716X.

Наукові інтереси:

– *розробка програмного забезпечення в області аналізу даних, комп'ютерне моделювання фізичних процесів.*

V.N. KARAZIN KHARKIV NATIONAL UNIVERSITY
NATIONAL SCIENCE CENTER KHARKIV INSTITUTE OF PHYSICS AND TECHNOLOGY
MAX PLANCK INSTITUTE OF MICROSTRUCTURE PHYSICS
TARAS SHEVCHENKO NATIONAL UNIVERSITY OF KYIV
INSTITUTE OF NUCLEAR CHEMISTRY AND TECHNOLOGY (Warsaw, Poland)
RIVNE STATE HUMANITARIAN UNIVERSITY
NATIONAL AEROSPACE UNIVERSITY NAMED AFTER N.E. ZHUKOVSKY (Kharkiv)
CJSC "INSTITUTE OF INFORMATION TECHNOLOGY" (Kharkiv)
KHERSON NATIONAL TECHNICAL UNIVERSITY
LLC "BUREAU IRIS" (Kyiv)
TEAM INTERNATIONAL SERVICES, INC. (Lake Mary, USA)

PROGRAM COMMITTEE:

Azarenkov M.O., Acad. Of the NAS of Ukraine, Prof., Dr. Sc, Kharkiv, Chairman
Bardachov Yu. M., prof., Prof., Dr. Sc, Kherson
BombaA.Ya., Prof., Dr. Sc, Rivne
Buy D. B., Prof., Dr. Sc, Kyiv
VaninV.A., Prof., Dr. Sc, Kharkiv
Gorbenko I.D., Prof., Dr. Sc, Kharkiv
Dolya G.M., Prof., Dr. Sc, Kharkiv
Zholtkevich G.M., Prof., Dr. Sc., Kharkiv
Kuklin V.M., Prof., Dr. Sc, Kharkiv
Lazurik V.T., Prof., Dr. Sc, Kharkiv
Rossomakhin S.G., Prof., Dr. Sc, Kharkiv
Savula Y. H., Prof., Dr. Sc, Lviv
Sporov O. E.,Assoc., Ph.D., Kharkiv
Styervoyedov A. Dr., Halle, Germany
Styervoedov M.G., Assoc., Ph.D., Kharkiv
Tolstoluzhska O.G., Prof., Dr. Sc, Kharkiv
Tkachuk M.V., Prof., Dr. Sc, Kharkiv
Kharchenko V.S., Prof., Dr. Sc, Kharkiv
Khomchenko A.N. Prof., Dr. Sc, Mykolaiv
Shmatkov S.I., Assoc., Dr. Sc, Kharkiv
Shulga M.F., Acad. Of the NAS of Ukraine, Prof., Dr. Sc, Kharkiv
Zimek Z., Ph.D., Warsaw, Poland
Yanovsky V.V., Prof., Dr. Sc, Kharkiv

ORGANIZING COMMITTEE:

Lazurik V.T., Dr. Sc, Prof., Dean of the CSD of V.N. Karazin Kharkiv National University, Chairman,
Sporov O. E., Assoc., Ph.D., V.N. Karazin Kharkiv National University, Deputy Chairman,
Tolstoluzhska O.G., Dr. Sc, Prof. of V.N. Karazin Kharkiv National University, Deputy Chairman,
Tkachuk M.V., Dr. Sc, Prof., Head of MST Dept of V.N. Karazin Kharkiv National University,
Kuklin V.M., Dr. Sc, Prof. Head of the AIS Dept of V.N. KarazinKharkiv National University,
Dyuldya S.V., Ph.D., Kharkiv Institute Of Physics And Technology,
Esin V.I., Dr. Sc, Prof. of V.N. Karazin Kharkiv National University,
Artyukh A.A., Head of Laboratory of V.N. Karazin Kharkiv National University,
Shevtsov S. O., CEO LLC "Bureau Iris" (Kyiv)
Zholtkevich G.M.,Dr. Sc, Prof. Dean of the MCS of V.N. Karazin Kharkiv National University,
Vanin V.A., Dr. Sc, Prof. NTU “KhPI”,
Zinoviev D.V., Senior Lecturer of V.N. Karazin Kharkiv National University,
Rassomakhin S.G., Dr. Sc, Prof., Head of ISTS Dept of V.N. Karazin Kharkiv National University,
Styervoyedov A. Dr.,Max Planck Institute of Microstructure Physics (Germany),
Petersen S., CEO TEAM International (Kharkiv),
Styervoedov M.G., Ph.D., Head of ECS Dept of V.N. Karazin Kharkiv National University,
Shmatkov S.I., Dr. Sc, Prof. of V.N. KarazinKharkiv National University,
Kruhol M.M., assistent of Power Plant Department, NTU KhPI.

http:// www.univer.kharkov.ua
http:// www-csd.univer.kharkov.ua

Наукове видання

«Комп'ютерне моделювання в наукоємних технологіях» (КМНТ-2021).

Збірник наукових праць міжнародної науково-технічної конференції, м. Харків, 21-23 квітня 2021 року

Українською та англійською мовами

Комп'ютерне верстання О.О. Афанасьєва

Підписано до друку 10.06.2021 р.
Формат 60×84/8. Папір офсетний. Друк ризограф.
Ум. друк. арк..- 30,2
Обл.- вид.арк.- 38,4
Наклад 20 пр. Зам. № 227/2021
Ціна договірна

61022, м. Харків, майдан Свободи, 4
Харківський національний університет імені В.Н. Каразіна.

Надруковано ФОП «Петров В.В.»
61144 Харків-144, вул. Гв.Широнінців, 79 в, к.137, тел. 362-01-52
Свідоцтво про державну реєстрацію ВОО №948011 від 03.01.03