

Міністерство освіти і науки України
Льотна академія Національного авіаційного університету
Інститут кібернетики ім. В.М.Глушкова Національної академії наук України
Центральноукраїнський національний технічний університет
Центральноукраїнський державний
педагогічний університет ім.В.Винниченка
Запорізький національний університет
Донецький національний медичний університет
Інститут модернізації та змісту освіти
Національний університет «Запорізька політехніка»
Громадська організація «Системні дослідження»

Матеріали
XXIII Міжнародного науково-практичного
семінару імені А.Я. Петренюка,
*присвяченого 70-річчю Льотної академії
Національного авіаційного університету*

**«КОМБІНАТОРНІ КОНФІГУРАЦІЇ
ТА ЇХНІ ЗАСТОСУВАННЯ»**

13-15 травня 2021 року

Запоріжжя - Кропивницький
2021 р.

УДК 519

К 63

*Рекомендовано до видання Вченою радою
Льотної академії Національного авіаційного університету
(протокол № 6 від 18.05.2021)*

Редакційна колегія:

Донець Г. П., д. ф.-м. н., професор;

Бакурова А.В., д.е.н., професор;

Козін І.В., д.ф.-м.н., професор;

Семенюта М.Ф., к.ф.-м.н., доцент;

Шендеровський В.А., д.ф.-м.н., професор;

Шерман З.О. – к.ф.-м.н.

К 63 Комбінаторні конфігурації та їхні застосування:

Матеріали XXIII Міжнародного науково-практичного семінару імені А.Я. Петренюка, присвяченого 70-річчю Льотної академії Національного авіаційного університету (Запоріжжя – Кропивницький, 13-15 травня 2021 року) / за ред. Г.П. Донця – Кропивницький: ПП «Ексклюзив-Систем», 2021. – 208 с.

ISBN 978-617-7942-07-7

У збірнику подано матеріали XXIII Міжнародного науково-практичного семінару «Комбінаторні конфігурації та їхні застосування» імені А.Я. Петренюка, який відбувся 13-15 травня 2021 року. Коло обговорених проблем: комбінаторні розміщення, графи і задачі оптимізації на них, автомати, скінченні алгебри, математична логіка і теорія множин, теорія ігор і теорія прийняття рішень та їхні застосування в різних галузях; реалізація математичних моделей на дискретних та неперервних множинах (автоматизовані системи керування, системи автоматизованого проектування, системи підтримки прийняття рішень); програмне забезпечення синтезу та аналізу комбінаторних конфігурацій; розпізнавання образів, нейромережі та методи машинного навчання; фрагментарні структури, алгоритми та метаевристики; топологія та історія розвитку української науки; кодування, інформаційна безпека та захист інформації.

Для науковців, викладачів вищих навчальних закладів, а також аспірантів, студентів і фахівців із застосування комбінаторних конфігурацій.

..... УДК 519

ISBN 978-617-7942-07-7 © ПП «Ексклюзив-Систем», 2021

Засновник семінару – Державна льотна академія України

Спонсор XXIII Міжнародного науково-практичного семінару «Комбінаторні конфігурації та їхні застосування» імені А.Я. Петренюка, присвяченого 70-річчю *Льотної академії Національного авіаційного університету – ІТ спільнота м. Кропивницького*

Програмний комітет:

Донець Г. П. – д.ф.-м.н., професор, завідувач відділом економічної кібернетики Інституту кібернетики НАН України; **голова програмного комітету;**

Семенюта М.Ф. – к.ф.-м.н., доцент, завідувач кафедри фізико-математичних дисциплін Льотної академії НАУ; **співголова програмного комітету;**

Бакурова А. В. – д.е.н., професор, професор кафедри «Системного аналізу та обчислювальної математики» НУ «Запорізька політехніка»;

Deineko Vladimir – PhD, DSc. Associate Professor (Reader), Warwick Business School, Great Britain;

Айвазян Е. І. – д.п.н., професор, професор кафедри загальної математики Єреванського державного університету;

Марков К. – співробітник інституту інформаційних теорій і застосувань, Софія, Болгарія;

Безущак О.О. – к.ф.-м.н., доцент, декан механіко-математичного факультету Київського національного університету ім. Т. Шевченка;

Гуляницький Л.Ф. – д.т.н., старший науковий співробітник, завідувач відділом методів комбінаторної оптимізації та інтелектуальних ІТ Інституту кібернетики ім. В.М. Глушкова НАН України;

Ісаченко О.М. – к.ф.-м.н., доцент, доцент кафедри інформаційних систем управління Білоруського державного університету;

Козін І.В. – д.ф.-м.н., професор, професор кафедри економічної кібернетики Запорізького національного університету;

Кривий С.Л. – д.ф.-м.н., професор, професор кафедри інтелектуальних програмних систем Київського національного університету ім. Т. Шевченка;

Маляр М.М. – д.т.н., професор, професор кафедри кібернетики і прикладної математики Ужгородського національного університету;

Мелешко Є.В. – д.т.н., доцент, доцент кафедри кібербезпеки та програмного забезпечення Центральноукраїнського національного технічного університету;

Новожилова М.В. – д.ф.-м.н., професор, завідувач кафедри прикладної математики та інформаційних технологій, Харківський національний університет міського господарства ім. О.Бекетова;

Перепелиця В.О. – д.ф.-м.н., професор, професор кафедри економічної кібернетики Запорізького національного університету;

Романова Т.Є. – д.т.н., професор, провідний науковий співробітник відділу математичного моделювання й оптимального проектування Інституту проблем машинобудування ім. А.М. Підгорного НАН України;

Семенова Н.В. – д.ф.-м.н., старший науковий співробітник, провідний науковий співробітник відділу методів дискретної оптимізації, математичного моделювання та аналізу складних систем Інституту кібернетики імені В.М. Глушкова НАН України;

Снитюк В.Є. – д.т.н., професор, декан факультету інформаційних технологій Київського національного університету ім. Т. Шевченка;

Стецюк П.І. – д.ф.-м.н., професор, завідувач відділом методів негладкої оптимізації Інституту кібернетики ім. В.М. Глушкова НАН України;

Теленик С.Ф. – д.ф.-м.н., професор, декан факультету інформатики та обчислювальної техніки Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського»;

Тимофієва Н.К. – д.т.н., старший науковий співробітник, провідний науковий співробітник відділу комплексних досліджень інформаційних технологій Міжнародного науково-навчального центру інформаційних технологій та систем НАН та МОН України;

Турчина В.А. – к.ф.-м.н., доцент, завідувач кафедри обчислювальної математики та математичної кібернетики Дніпровського національного університету ім.О.Гончара;

Ус С. А. – к.ф.-м.н., доцент, професор кафедри «Системного аналізу і управління» Національного технічного університету «Дніпровська політехніка»;

Шендеровський В. А. – віце-президент Українського фізичного товариства, д.ф.-м.н., професор, провідний науковий співробітник Інституту фізики НАН України;

Яковлєв С.В. – заслужений діяч науки і техніки України, д.ф.-м.н., професор, професор кафедри інформатики Національного аерокосмічного університету ім. М. Є. Жуковського "ХАІ".

Члени оргкомітету:

Наумик В.В. – професор, д.т.н., проректор з НР та МД Національного університету «Запорізька політехніка»; голова організаційного комітету;

Корніч Г.В. – д.ф.-м.н., професор, завідувач кафедри «Системного аналізу та обчислювальної математики» НУ «Запорізька політехніка»;

Мікасян Г. С. – к.ф.-м.н., д.п.н., професор, професор кафедри математики і методики її викладання Вірменського державного педагогічного університету ім. Х.Абовяна;

Джума Л.М. – к.т.н., доцент кафедри інформаційних технологій Льотної академії НАУ;

Дмітрієв О. М. – д.т.н., доцент, завідувач кафедри льотної експлуатації, аеродинаміки та динаміки польоту Льотної академії НАУ;

Козіна Г.Л. – к.ф.-м.н., доцент, доцент кафедри «Захист інформації» НУ «Запорізька політехніка»;

Кривцун О.В. – к.ф.-м.н., доцент;

Курапов С.В. – к.ф.-м.н., доцент, доцент кафедри програмної інженерії Запорізького національного університету;

Неділько В. М. – к.т.н., доцент, завідувач кафедри інформаційних технологій Льотної академії НАУ;

Сорока М.Ю. – к.т.н., в.о. заступника начальника Льотної академії НАУ з навчальної науково-методичної та виховної роботи;

Петренюк В.І. – к.ф.-м.н., доцент, доцент кафедри кібербезпеки та програмного забезпечення Центральноукраїнського національного технічного університету;

Плачинда Т. С. – д.п.н., професор, завідувач кафедри професійної педагогіки та соціально-гуманітарних наук;

Рябенко А.Є. – к.ф.-м.н., доцент, доцент кафедри «Системного аналізу та обчислювальної математики» НУ «Запорізька політехніка»;

Терещенко Е.В. – к.ф.-м.н., доцент, доцент кафедри «Системного аналізу та обчислювальної математики» НУ «Запорізька політехніка»;

Шерман З.О. – к.ф.-м.н., старший викладач кафедри медичної фізики та інформаційних технологій №2 Донецького національного медичного університету;

Якуніна І.Л. – к.т.н., доцент кафедри фізико-математичних дисциплін Льотної академії НАУ;

Громадська організація «Системні дослідження».

Шановні колеги! Шановні учасники XXIII Міжнародного науково-практичного семінару імені А. Я. Петренюка «Комбінаторні конфігурації та їхнє застосування», мета якого не змінюється впродовж 15-ти років (невеличкий ювілей) – об'єднання науковців різних напрямків, які займаються дослідженнями комбінаторних конфігурацій та їхнім застосуванням.

Сподіваюся, що є ще багато учасників першого семінару, який відбувся у приміщенні Державної льотної академії України, за ініціативи науковців якої та підтримки вчених Інституту кібернетики ім. В. М. Глушкова НАН України, Дніпропетровського національного університету та Чернівецького національного університету ім. Ю. Федьковича.

Як можемо бачити сьогодні, ідея запровадження такого семінару виявилася успішною і плідною, що уможливило висвітлення результатів окремих вчених і наукових груп, які працюють в галузі дискретної математики та її застосувань; щодо колективного обговорення нових ідей та наукової співпраці. Очевидно, що у кожній ідеї є її автори. Згадаю тут вже як фізик і як історик науки думку видатного вченого-фізика Сергія Вавилова: «Історія науки не може обмежуватися вивченням розвитку ідей – однаковою мірою вона повинна торкатися живих людей, з їхніми особливостями, талантами, залежністю від соціальних умов, країни та епохи».

У контексті цієї думки вважаю, що ідея заснування семінару належала доктору фізико-математичних наук, професору Кіровоградського національного технічного університету, талановитому математику в області теорії скінченних інцидентних структур Анатолію Яковичу Петренюку. Звичайно, цей задум виник у Анатолія Яковича на основі успішної діяльності семінарів такого типу в Інституті математики АН УРСР, спорідненого за тематикою семінару в Одесі та першого комбінаторного семінару у Московському державному університеті, в якому навчався вчений. Тому намір розвивати традиції такого типу математичні семінари було певною мірою визначеним алгоритмом, що і підтримали науковці Інституту кібернетики НАНУ (професор Г. П. Донець)

і Кіровоградської льотної академії національного авіаційного університету (професор С. М. Неділько).

Саме тоді, на першому семінарі, зі вступним словом виступив проректор з навчальної і наукової роботи Державної льотної академії України професор С. М. Неділько.

Мав і я тоді вітальне слово учасникам семінару, а також доповідь про маловідомих або замовчуваних вчених-математиків, яка викликала особливе зацікавлення. Хочу і нині наголосити, що важливе значення для подальшого розвитку наукових ідей має пізнання і вивчення історичного минулого, як засвідчував, зокрема, учитель славетного вченого Миколи Боголюбова математик і механік Микола Митрофанович Крилов. Вивчення біобібліографій вчених дає завжди унікальний емпіричний матеріал, їхній досвід є безцінним при дослідженні науково-технічного потенціалу і його розвитку.

Важливим у цьому аспекті є прийняття постанови Президії Національної академії наук України від 17 лютого 2021 року про активізацію комунікаційної діяльності в НАН України і створення комісії з питань популяризації наукової діяльності.

Для учасників семінару, як і тоді, на першому семінарі, наголошу, що вийшла у світ узагальнююча моя праця про видатних, але маловідомих вчених, яких народила українська земля. Через певні обставини їхні імена замовчували, а наукову спадщину вилучили зі скарбівні української науки. Книга має назву «Вчені України у світовій науці» і містить близько 200 нарисів. Серед них є і нариси про багатьох вчених-математиків, зокрема про Віктора Буняковського, Євгена Вікторовського, Юрія Вороного, Миколу Гулака, Мирона Зарицького, Михайла Кравчука, Михайла Остроградського, Олексія Погорелова, Миколу Чайківського, Пилипа Кулика, Миколу Чеботарьова і багатьох інших.

Переконаний, що і науковцям, і аспірантам, і студентам, і фахівцям із застосування комбінаторних комбінацій, і всім учасникам науково-практичного семінару ознайомлення з видатними постатями математичної науки буде стимулюючим фактором у їхній науковій діяльності.

Вітаю нинішніх керівників семінару Марину Фролівну Семенюту та Анну Володимирівну Бакурову, оргкомітет семінару, учасників усіх наукових і освітянських установ і бажаю успішної роботи. Сподіваюся, що обставини, не залежні від нас, минуться, і в майбутньому ми будемо проводити Міжнародний науково-практичний семінар імені Анатолія Яковича Петренюка у реальному вимірі.

*Доктор фізико-математичних
наук, професор, провідний
науковий співробітник Інституту
фізики НАН України
Василь Шендеровський*

Arhipova N. A., Voblyi V. A.

VINITI RAS

THE PROBLEM OF AUTOMATING THE RUBRICATION
IN PRODUCTION OF THE ABSTRACT JOURNAL

"MATHEMATICS" 17

Kondruk N. E.

Uzhhorod National University

CONCEPTUAL MODEL OF AUTOMATED SYSTEMS

FOR DEVELOPMENT OF A PERSONAL DIET PLAN 20

Revyakin A.M., Isachenko A. N.

National Research University of Electronic Technology –

MIET

Belarusian State University

RIGIDITY OF PLANAR TRUSSES WITH REMOVED

FRAGMENTS 22

Riznyk V.V.

Lviv Polytechnic National University

MATHEMATICAL MODELS OF DATA PROCESSING

UNDER THE MANIFOLD COORDINATE

SYSTEMS..... 27

Semeniuta M.F., Sherman Z.A.

National Aviation University

Donetsk National Medical University

ON THE KAMACHI-ARUMUGAM CONJECTURE ON

DISTANCE ANTIMAGIC LABELING OF GRAPHS..... 34

Voblyi V. A.

VINITI RAS

ANALOGUE OF COMBINATORY KOGANOV IDENTITY

FOR ORDERED BELL NUMBERS 37

Бакурова А.В., Слестніков О.П., Кондратов Д.О.

Національний університет «Запорізька політехніка»

ЗАСТОСУВАННЯ МЕТОДУ ПАТЕРНІВ ДО АНАЛІЗУ

ДИНАМІКИ ПОКАЗНИКІВ РОЗВИТКУ СОЦІАЛЬНО-

ЕКОНОМІЧНИХ СИСТЕМ..... 39

Бондар О.П.

Льотна академія НАУ

АРИФМЕТИКА БІТКОІНА У ПРИКЛАДАХ..... 44

Волков Ю.І., Войналович Н.М.

*Центральноукраїнський державний педагогічний
університет імені Володимира Винниченка*

ТРЕСНОМІАЛЬНИЙ РОЗПОДІЛ 48

Глухов О.Д.

Національний авіаційний університет

ПРО ПАРКЕТИ НА ДЕЯКИХ ПОВЕРХНЯХ 51

Гогерчак Г.І.

*Київський національний університет імені Тараса
Шевченка*

ДЕСКРИПТИВНІ ЛОГІКИ ТА ПРИРОДНОМОВНІ
СИСТЕМИ ЗНАНЬ..... 56

Горобець В.І., Дубровін В. І., Твердохліб Ю.В.

*Національний університет “Запорізька Політехніка”,
FRESHCODE*

ПОВЕДІНКОВІ МЕТОДИ ВИЯВЛЕННЯ
НЕСАНКЦІОНОВАНИХ ДІЙ ТА АТАК В МЕРЕЖАХ.... 60

Гринченко П.В.

САОМ, НУЗП

СУЧАСНІ АЛГОРИТМИ ВИЯВЛЕННЯ АНОМАЛІЙ
ТРАФІКУ МЕТОДАМИ ВЕЙВЛЕТ-АНАЛІЗУ 64

Гуляницький Л.Ф., Рясна І.І.

Інститут кібернетики імені В.М. Глушкова НАН України

НЕЧІТКА КОМБІНАТОРНА МОДЕЛЬ ОДНІЄЇ ЗАДАЧІ
ВИБОРУ НА ОСНОВІ НЕЧІТКИХ ВІДНОШЕНЬ..... 67

Джума Л. М., Серeda О.А.

Льотна академія НАУ

ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЕТАЛОННОЇ МОДЕЛІ
СУБ’ЄКТА НАВЧАННЯ..... 72

Доценко С.І.

*Київський національний університет імені Тараса
Шевченка*

КОМБІНАТОРНІ СТРУКТУРИ В НАСТІЛЬНИХ ТА
АЗАРТНИХ ІГРАХ..... 77

Євдокимов І.В., Ус С.А.

Київський національний університет імені Тараса Шевченка

Національний технічний університет «Дніпровська політехніка»

ВИБІР МІСЦЯ РОЗМІЩЕННЯ КІНОТЕАТРУ НА МАПІ МІСТА..... 81

Ізвалов О.В.

Летная академия НАУ

ПОБУДОВА ТА АНАЛІЗ ГРАФУ ЗВ'ЯЗНОСТІ ДОДАТКІВ У МОБІЛЬНОМУ МАГАЗИНІ GOOGLE PLAY..... 85

Кадубовський О.А.

ДВНЗ «Донбаський державний педагогічний університет»

ПЕРЕРАХУВАННЯ НЕІЗОМОРФНИХ ДВОКОЛЬОРОВИХ ХОРДОВИХ О-ДІАГРАМ РОДУ ТРИ З ОДНИМ СІРИМ (АБО ЧОРНИМ) ЦИКЛОМ 89

Караваєв К. Д., Турчина В.А.

Дніпровський національний університет імені Олеся Гончара

ДЕЯКІ УЗАГАЛЬНЕННЯ ЗАДАЧІ ПАРАЛЕЛЬНОГО УПОРЯДКУВАННЯ..... 93

Коваленко Є. О., Турчина В. А

Дніпровський національний університет імені Олеся Гончара

АНАЛІЗ СТРУКТУРИ ГРАФІВ В ЗАДАЧАХ ПАРАЛЕЛЬНОГО УПОРЯДКУВАННЯ З ПЕРЕРИВАННЯМИ..... 98

Козин И. В., Перепелица В.А., Нарзуллаев У.Х.

*Запорожский национальный университет,
Самаркандский филиал Ташкентского университета
информационных технологий*

ФРАГМЕНТАРНЫЙ ПОДХОД В ТРУДНОФОРМАЛИЗУЕМЫХ ЗАДАЧАХ ОПТИМИЗАЦИИ СО МНОГИМИ КРИТЕРИЯМИ..... 102

Козин И. В., Перепелица В.А., Нарзуллаев У.Х. <i>Запорожский национальный университет</i> <i>Самаркандский филиал Ташкентского университета</i> <i>информационных технологий</i>	
ФРАГМЕНТАРНЫЙ ПОДХОД В ТРУДНОФОРМАЛИЗУЕМЫХ ЗАДАЧАХ ОПТИМИЗАЦИИ СО МНОГИМИ КРИТЕРИЯМИ.....	106
Кривий С.Л. <i>Київський національний університет імені Тараса Шевченка</i>	
РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ РІВНЯНЬ В АСОЦІАТИВНИХ КІЛЬЦЯХ З ОДИНИЦЕЮ.....	109
Мазютинець Г.В., Маляр М.М., Шаркаді М.М. <i>Ужгородський національний університет</i>	
ІНФОРМАЦІЙНА МОДЕЛЬ ФІНАНСОВОЇ БЕЗПЕКИ СУБ'ЄКТІВ ГОСПОДАРЮВАННЯ.....	113
Маляр-Газда Н.М., Шаркаді М.М. <i>Ужгородський національний університет</i>	
НЕЧІТКЕ МОДЕЛЮВАННЯ В МЕДИЧНІЙ ДІАГНОСТИЦІ.....	116
Мелешко Є.В. <i>Центральноукраїнський національний технічний університет</i>	
МЕТОД ПРОГРАМНОГО ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ КОРИСТУВАЧІВ ТА БОТІВ У РЕКОМЕНДАЦІЙНІЙ СИСТЕМІ.....	119
Мелешко Є.В., Якименко М.С., Міхав В.В. <i>Центральноукраїнський національний технічний університет</i>	
ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ГРАФОВОЇ БАЗИ ДАНИХ NEO4J ДЛЯ ЗАСТОСУВАННЯ У РОЗРОБЦІ РЕКОМЕНДАЦІЙНИХ СИСТЕМ СОЦІАЛЬНИХ МЕРЕЖ.....	124

Пасенченко Ю. А.

*Військовий інститут телекомунікацій та
інформатизації ім. Героїв Крут*

ПРО ДЕЯКІ ЗАДАЧІ ВЕКТОРНОЇ
ОПТИМІЗАЦІЇ..... 127

Петренюк В. І., Петренюк Д. А.

*Центральноукраїнський національний технічний
університет*

ПРОТОТИПИ ГРАФІВ ОБСТРУКЦІЙ ЗАДАНОГО
НЕОРІЄНТОВАНОГО РОДУ..... 130

Самарай В.П.

*Відкритий міжнародний університет розвитку людини
“Україна”*

МЕРЕЖІ ПЕТРІ ДЛЯ МОДЕЛЮВАННЯ РОЗПОДІЛУ
МАТЕРІАЛІВ 141

Семенова Н, Колєчкін В.

*Інститут кібернетики імені В.М. Глушкова НАН
України*

ПІДХІД ДО РОЗВ’ЯЗАННЯ ЗАДАЧІ ВЕКТОРНОЇ
ОПТИМІЗАЦІЇ НА КОМБІНАТОРНІЙ КОНФІГУРАЦІЇ
ПЕРЕСТАНОВОК З НЕЧІТКО ЗАДАНИМИ ДАНИМИ... 145

Семенюта М.Ф., Дмитриев О.Н., Сорока М.Ю.

Летная академия НАУ

ПРО ДОСТАТНЮ УМОВУ ІСНУВАННЯ
ДИСТАНЦІЙНОЇ МАГІЧНОЇ РОЗМІТКИ ГРАФІВ..... 149

Семенюта М.Ф., Донець Г.П., Марков К.

Летная академия НАУ,

*Інститут кібернетики імені В.М. Глушкова НАН України
Інститут інформаційних теорій і застосувань, Софія,
Болгарія*

ЗАДАЧІ, ЩО ПРИВЕЛИ ДО ПОНЯТТЯ РОЗМІТКИ
ГРАФА..... 151

Сікірда Ю.В., Шмельова Т.Ф., Касаткін М.В.

Летная академия НАУ

Національний авіаційний університет

Харківський національний університет Повітряних Сил

ім. І. Кожедуба

ЗАСТОСУВАННЯ МЕТОДУ ПАРЕТО ДЛЯ
РОЗВ'ЯЗАННЯ ЗАДАЧІ ОПТИМІЗАЦІЇ ВЗАЄМОДІЇ
ПІЛОТА ТА АВІАДИСПЕТЧЕРА В ПОЗАШТАТНИХ
СИТУАЦІЯХ 156

Сіряк А.В., Турчина В.А.

Дніпровський національний університет імені Олеся

Гончара

ПРО ОДНЕ УЗАГАЛЬНЕННЯ ЗАДАЧІ
ПАРАЛЕЛЬНОГО УПОРЯДКУВАННЯ..... 160

Скрябіна А. В., Стєганцева П. Г.

Запорізький національний університет

ПЕРЕРАХУВАННЯ УСІХ T_0 -ТОПОЛОГІЙ, ЗАДАНИХ
ОДНИМ ВЕКТОРОМ, ТА ПІДРАХУНОК
 T_0 - ТОПОЛОГІЙ ПЕВНОЇ ВАГИ..... 165

Соломко М. Т.

Національний університет водного господарства та

природокористування

МІНІМІЗАЦІЯ БУЛЕВИХ ФУНКЦІЙ У
ПОЛІНОМНОМУ ТА ЗМІШАНОМУ БАЗИСАХ
МЕТОДОМ ОБРАЗНИХ ПЕРЕТВОРЕНЬ..... 168

Стецюк П.І., Хом'як О.М.

Інститут кібернетики імені В.М. Глушкова НАН України

ЗАСТОСУВАННЯ ПАКЕТУ GLPK ДЛЯ
ЗНАХОДЖЕННЯ ВСІХ РОЗВ'ЯЗКІВ ЗАДАЧІ ПРО
К- ПЛЕКС..... 174

Терещенко Е.В.

Національний університет «Запорізька політехніка»

ФОРМАЛІЗАЦІЯ ЗАДАЧІ ПОШУКУ СПОРІДНЕНOSTІ
ЗА МОРФОЛОГІЧНИМИ ОЗНАКАМИ..... 179

Тимофієва Н.К.

Міжнародний науково-навчальний центр інформаційних технологій та систем НАН та МОН України

**СИТУАЦІЯ НЕВИЗНАЧЕНОСТІ В ЗАДАЧАХ
ШТУЧНОГО ІНТЕЛЕКТУ, ЩО ПОВ'ЯЗАНА З
КОМБІНАТОРНИМИ КОНФІГУРАЦІЯМИ.....** 183

Турчина В.А., Челпанова О. О.

Дніпровський національний університет імені Олеся Гончара

**УЗАГАЛЬНЕННЯ АНОМАЛІЙ В ЗАДАЧАХ
УПОРЯДКУВАННЯ.....** 188

Черноусова Ж. Т.

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

**СТАТИСТИЧНІ МЕТОДИ ПРИЙНЯТТЯ РІШЕНЬ І
СЕРЕДНІ ВТРАТИ.....** 191

Шимко С.В., Башенко Д.В., Мелешко Є.В.,

Маключенко С.І.,

Центральноукраїнський національний технічний університет

**ДОСЛІДЖЕННЯ МЕТОДІВ ПОБУДОВИ СУЧАСНИХ
НЕЙРОННИХ МЕРЕЖ ДЛЯ ЗАДАЧ АВТОМАТИЗАЦІЇ
КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ.....** 195

Яценко А.К., Дубровін В.І., Твердохліб Ю.В.

Національний університет «Запорізька політехніка»

Freshcode

Аналіз мережевого трафіку за допомогою ентропії..... 198

THE PROBLEM OF AUTOMATING THE RUBRICATION IN PRODUCTION OF THE ABSTRACT JOURNAL "MATHEMATICS"

Arkhipova N. A., Voblyi V. A.
VINITI RAS

Abstract. The authors analyze the problem of automating the rubrication of documents in the production of the abstract journal "Mathematics". The system of automatic thematic classification of texts developed at VINITI RAS allows us to partially solve this problem.

VINITI RAS has an automated workstation "Proofreader, Editor, Referent" (KOREF), which provides changes to the descriptions of documents directly in the Unified Technological Database (UTDB) when performing operations of thematic markup, abstracting, scientific editing, indexing, proofreading in the production of the abstract journal "Mathematics" [1]. An important technological stage of document processing is its rubrication, that is, the assignment of the classifier section code to the document. For the abstract journal "Mathematics", documents are categorized according to the UDC (Universal Decimal Classification) and SRSTI (State Rubricator Scientific and Technical Information). Rubrication of documents is necessary to speed up their information search in a large array of documents. For domestic articles and conference materials in accordance with GOST R 7.0.7-2009, the indication of the UDC when publishing is mandatory. However, the share of Russian-language documents submitted to VINITI for the abstract journal "Mathematics" is only about 20% of the total flow, and the categorization of foreign documents is currently performed manually by the editors of the journal sections. Due to the constant growth of the volume of processed information, the task of automating the rubrication of English-language documents is relevant.

It is necessary to distinguish between rubrication (classification) and clustering. For a rubrication, the rubrics are set, but when clustering, the number of rubrics may not be known, and the set of documents is divided into groups (clusters) based on similarity. In our case, the rubricator is set, and it rarely changes.

Rubrication is complicated by the possible ambiguity of assigning a rubric to a document due to the intersection of the thematic areas of the rubrics. For example, a traveling salesman's task is assigned by different experts to the section "Operations Research. Discrete programming. Combinatorial methods" (UDC 519.854.2), or in the section "Algorithmic problems of combinatorial analysis" (UDC 519.16), or in the section "Algorithmic problems of graph theory" (UDC 519.178). The way out of this situation is to rank the resulting categories, that is, to assign several UDC documents with certain ranks or probabilities.

Automatic rubrication methods are divided into two classes: knowledge-based methods and machine learning methods [2]. Knowledge-based methods use the knowledge base to build a categorization model, such as a thesaurus or an ontology. On the basis of this database, experts build rules for assigning documents to rubrics, usually in the form of boolean formulas. These methods require a lot of manual work by experts in the subject area.

Machine learning methods based on the training collection of documents create a model that can do rubrication new documents [3,4]. A number of machine learning methods are known: metric, statistical, and linear classification methods, regression methods, and neural networks. The disadvantage of these methods is the need to have a marked-up array of documents, which is not always possible. VINITI has a huge stock of hand-classified documents that can be used to train the robot program.

The most well-known machine learning method is the Bayes method. This method consists in calculating the probabilities of matching a document to each of the categories and selecting the category with the highest probability. Training is performed by calculating the probabilities of categories on a marked-up collection of documents. The method has a high speed of operation and a simple mathematical model, it is often used as a basic method when comparing different methods of machine learning.

VINITI RAS has developed a system of automatic thematic classification of texts (ATCT) [5]. The main structural and functional components of the ATCT software package are a text preprocessor, a vectorizer, and a classifier program. The text preprocessor module

solves the problem of cleaning the input document from markup elements, service characters, and also performs lemmatization, that is, bringing all words to the initial form. As a result, the text is translated into a universal internal representation, independent of the input format. To take into account the specifics of Russian - and English-language texts, the ATCT provides appropriate tools. The vectorizer converts the text to its vector model of the specified dimension.

To do this, the ATCT uses the Word2Vec technology, which allows to represent the text as a numeric vector, while preserving its semantics in the form of the most significant or averaged thematic features of the words that make up the text. The main parameter of Word2Vec is the dimension of the resulting vector; it determines the detail of the expression of meaning in the thematic space. The classifier program, based on the vector representation of the text, makes a decision about which categories the text belongs to, and evaluates its probability for each category. This program forms a thematic profile of the text-for all variants of classification schemes, which it is trained to work with. This module implements the possibility of applying several of the currently most well-known machine learning algorithms and decision-making models that use natural language processing methods based on computer analysis using artificial intelligence and computational linguistics.

The ATCT system is based on machine learning technology, training samples contain the results of manual classification of documents performed by VINITI specialists for already published issues of the abstract journal. The essential input data for the ATCT system are the titles, annotations, and keywords of the documents. Machine learning methods such as logistic regression and perceptron are mainly used.

Currently, the ATCT system works in VINITI in an experimental mode, it can process 1-3 million documents per day. For each document, the probability of the SRSTI category of the 2nd level is determined [5]. However, the VINITI rubricator is a development of SRSTI up to level 9. Therefore, in the near future, it is impossible to completely abandon the manual rubrication of documents by the editors of the sections of the abstract journal.

References

1. Arkhipova N. A., Voblyi V. A. Modern technology of paperless informatics in the production of the abstract journal "Mathematics". Mater. XXI International. seminar "Combinatorial configurations and their applications", Kropyvnytskyi, 2019, p. 17-20.
2. Peskova O. V. Methods of automatic classification of text electronic documents. Scientific and technical information. Ser. 2, Inform. processes and systems, 2006, No. 3, pp. 13-20.
3. Zhalybin A. A. Automatic rubrication of documents using methods of neural learning. Prospects of science. No. 6[129], 2020, pp. 10-14.
4. Applied and computational linguistics. Edited by I. S. Nikolaev, O. V. Mitrenina, T. M. Lando, M.: LENAND, 2016.
5. Egorov V. S., Kozlova E. S., Lomotin K. E., Fedorets O. V., Filimonov A.V., Shapkin A.V. Automatic text classification system for processing the flow of scientific publications in VINITI RAS. Scientific and technical information. Ser. 2, 2020, No. 5, pp. 1-12.

CONCEPTUAL MODEL OF AUTOMATED SYSTEMS FOR DEVELOPMENT OF A PERSONAL DIET PLAN

Kondruk N. E.

Uzhhorod National University

Abstract. The development of human-machine systems in the field of medicine, which combine the capabilities of modern technical means and knowledge and skills of doctors in decision-making, is becoming increasingly important. The paper describes the features of the conceptual model of automated systems for compiling individualized diets for patients.

Today, more and more attention is paid to the creation of systems that combine the capabilities of modern technology and human knowledge and skills in decision making. In the medical field, in particular, the development of computer information systems is relevant, which is used to support various activities in situations where it is impossible or undesirable to have an automatic system that

fully performs the entire process of finding solutions. In this way, the system does not replace the person, but only helps him in the process of solving the problem. One of such problems is the task of creating a diet plan for therapeutic, therapeutic and prophylactic nutrition and belongs to the field of diet therapy and hygiene. It involves the individualization of therapeutic nutrition, taking into account the possible presence of several diseases in the patient, his sex, age, body weight, intolerance of certain products.

The conceptual model of such systems includes a user-system interface that implements tools for generating and managing dialogue, a product database (PD), and a model database (MD).

The PD should contain a sufficient number of types of products for the preparation of various diets and information about the content of basic nutrients and their maximum allowable doses in the daily diet of the patient.

The interface of the system should provide the user with the opportunity to select the required set of products from the PD, physiological norms of nutrients (set taking into account age, sex, weight of the patient) and determine the purpose of the diet (enrichment of the diet with vitamins and minerals).

The MD should include the formation of flexible models for decision-making. That is, models should be automatically generated based on the emerging needs of the user. To implement this, it is proposed to use a general mathematical model of balanced human nutrition problems [1], which belongs to the class of linear programming problems with a large criterion space and fuzzy parameters. [2-3] describes the information model of the process of solving this problems.

The development of the described machine systems [3] eliminates the technical difficulties of calculating the components of diets with the required content of fats, proteins, carbohydrates, calories, the main nutrients of some diet therapy and balanced nutrition and reduces the time of medical treatment. The diet is economically justified and provides a choice of its components. In addition, the individualization of the diet increases its preventive and curative effectiveness, which leads, in particular, to a shorter period of rehabilitation of the patient.

References

1. Kondruk, N. E. Application of multicriteria models for the problems of a balanced diet / N. E. Kondruk, M. M. Malyar // Bulletin of Cherkasy State Technological University. Series: Engineering Sciences. – 2010. – Issue 1. – P. 3–7.
2. Malyar M. M. Structuring of the criterional space by an angle similarity measure. / M. M. Malyar, N.E. Kondruk // Scientific Bulletin of Uzhhorod University: series: Mathematics and Informatics - Uzhhorod: "Hoverla", 2020. - Issue. 1 (36). - P. 85 - 91. DOI: 10.24144 / 2616-7700.2020.1 (36) .85-91.
3. Malyar, M. M. Algorithm for reducing the number of criteria in a multicriteria linear programming problem / M. M. Malyar, N. E. Tsitsika // Visnyk of Kyiv National University. Series f.–m. Science. - 2004. - Issue. 2. - P. 288–292.
4. Kondruk, N. E. Decision Support Systems for automated diets / N. E. Kondruk // Management of Development of Complex Systems. – 2015. – Issue 23 (1). – P. 110–114.

RIGIDITY OF PLANAR TRUSSES WITH REMOVED FRAGMENTS

Revyakin A.M.

National Research University of Electronic Technology – MIET

Isachenko A. N.

Belarusian State University

Abstract. The authors continued to study the applications of matroids in mathematical modeling of various systems. The problems related to the rigidity of frameworks are considered. It is shown how the rigidity of planar frames with removed fragments is related to the bases of a cyclic matroid of a specially constructed graph.

The study of the rigidity of planar trusses was continued [1-5]. In statics, the equilibrium condition is usually determined by a system of differential or algebraic equations. By constructing an appropriate matroid for a system of equations, we can formulate a criterion for its solvability.

Consider a $k \times l$ square lattice as a planar truss consisting of rigid rods and connecting hinges. If you add additional diagonal rods to some squares, then in some cases the truss becomes rigid, i.e. it does not give in to deformation, and in others it does not. A natural question arises: where should we add the diagonal rods to make the truss rigid? Let's define a bipartite graph $G=(V, E)$ with a set of vertices $V=\{v_1, v_2, \dots, v_k, u_1, u_2, \dots, u_l\}$ and connect the vertices v_i and u_j an edge in it, if and only if a diagonal rod is placed in the square of the grid corresponding to i -th row and j -th column.

Bolker and Krapo proved [6,7] that a square lattice with a certain set of diagonal rods is rigid if and only if the bipartite graph $G=(V, E)$ is connected. Hence, the minimum number of diagonal bars that need to be added for the rigidity $k \times l$ square truss is $k+l-1$. This number is equal to the number of edges in the corresponding spanning tree of the graph G .

For example, the 3×3 -truss in Figure 1 is not a truss, since the corresponding bipartite graph is not connected. The possible deformation of the truss is also shown in Figure 1.

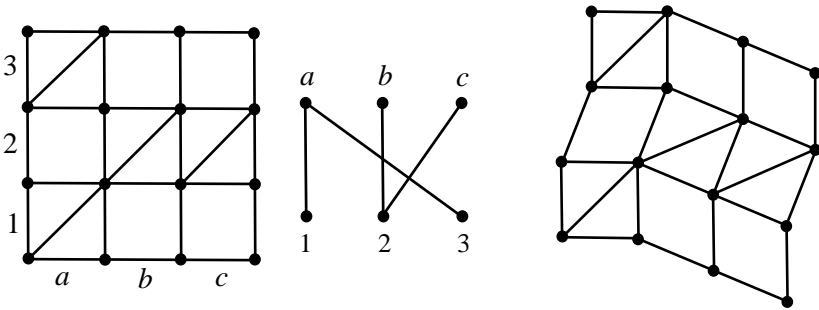


Fig. 1. *Planar truss, its bipartite graph and deformation*

Let us now consider a truss obtained from a square $k \times l$ -lattice by removing a $u \times s$ -fragment from it (see Fig. 2).

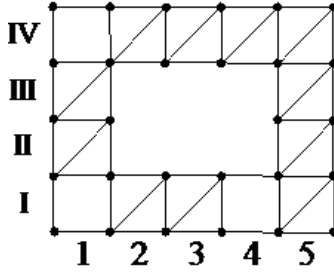


Fig. 2. Planar 5x4-farm with a remote 3×2 -fragment

Now the question: what is the minimum system of diagonal rods that will make the truss rigid? To answer this question, let's update our old bipartite graph.

Replace each vertex corresponding to the line containing the deleted fragment with a pair of vertices, calling them left and right vertices, respectively. Connect the left and right vertexes with a red edge. We will do the same with the vertices corresponding to the columns of the farm. The left and right ends are connected by a green edge.

The resulting "new graph" is shown in Figure 3. Denote it by $G_{k,l,s,u}$.

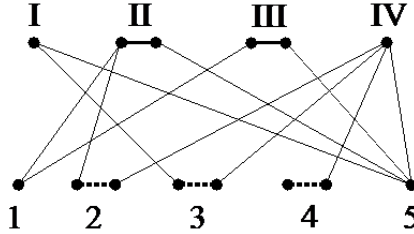


Fig. 3. Graph $G_{k,l,s,u}$

Theorem 1. Suppose that a square $k \times l$ -lattice is removed from its $u \times s$ -fragment. Then adding $k + l + s + u - 3$ diagonal rods to the truss will make it rigid if and only if the set of edges corresponding

to the added diagonal rods forms the base of the cyclic matroid of the graph $G_{k,l,s,u}$ after adding exactly one edge of each color.

In other words, the edges of the graph corresponding to the added diagonal elements form a 3-component forest F and there is one "red" edge a and one "green" edge b such that $F \cup \{a,b\}$ – the tree in $G_{k,l,s,u}$.

For example, for a planar 5×4 -truss with a deleted 3×2 fragment, you need to add 11 diagonal elements for rigidity (Fig. 4).

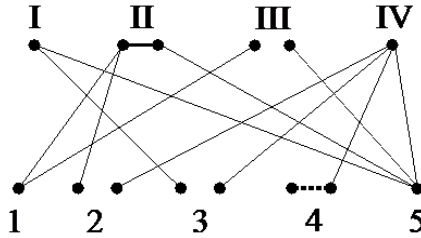


Fig. 4. *Rigid truss graph*

Consider a truss obtained from a square $k \times l$ -lattice by removing two parallel rectangular $u \times s$ -fragments (Fig. 5).

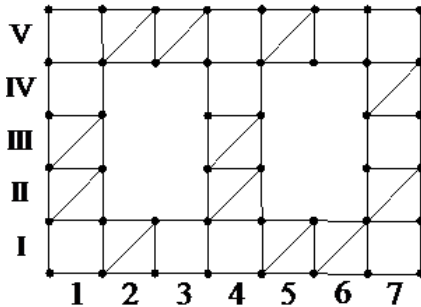


Fig. 5. *A farm with two parallel fragments removed*

Update the bipartite graph. Replace each vertex corresponding to the line of the deleted fragment with three vertices. Connect the first and second copies of the top row vertex with the first color, and

the second and third copies with the second color. Similarly, replace each vertex of the corresponding column that contains the deleted cells with a pair of vertices, connecting them with edges of different (third or fourth) colors, depending on the number of the deleted fragment. The number of edges in the graph remains the same. Only the vertices of the column graph of the farm over which there were no deleted cells are connected to the first copies of the vertexes of the farm rows, the vertexes of the farm columns lying between the deleted fragments are connected to the second, and the remaining vertexes of the farm columns are connected to the third copies of the vertexes of the farm rows. The resulting graph (Fig.6) denote by $G_{k,l,2s,2u}$.

Theorem 2. Suppose that two parallel rectangular $u \times s$ -fragments are removed from a square $k \times l$ -lattice. Then adding $k+l+2s+2u-5$ diagonal rods to the truss will make it rigid if and only if the set of edges corresponding to the added diagonal rods forms the base of the cyclic matroid of the graph $G_{k,l,2s,2u}$ after adding exactly one edge of each color.

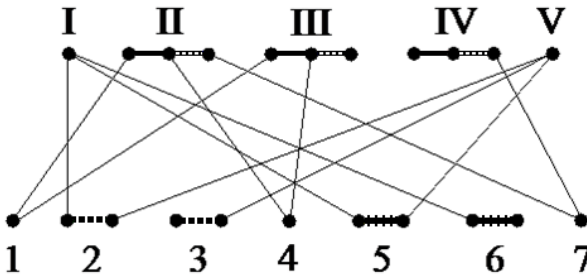


Fig. 6. Graph $G_{k,l,2s,2u}$

For example, for a square 5×7 -lattice in which two parallel 3×2 -fragments are removed, 17 diagonal rods must be added for rigidity.

References

1. Isachenko A. N., Revyakin A. M. Matroids in mathematical modeling of rod systems. // Mater. XXI International. seminar

"Combinatorial configurations and their applications", Kropyvnytskyi, 2019, p. 53-57.

2. Isachenko A. N., Revyakin A.M., Orientability, planning of transport flows and rigidity of farms. // Mater. XIX International. seminar "Combinatorial configurations and their applications", Kropyvnytskyi, 2017, p. 61-67.

3. Revyakin A.M., Rechki A. Rigidity of planar square trusses with remote fragments. //Proceedings of the YIII International Seminar "Discrete Mathematics and its applications" (February 2-6, 2004) / Edited by O. B. Lupanov. - Moscow: MSU, 2004. - p. 222-224.

4. Revyakin A.M. Matroids and rigidity of planar trusses. //Problems of theoretical cybernetics. Abstracts of the XIV International Conference (Penza, May 23-28, 2005) - Moscow: Publishing House of the Faculty of Mechanics and Mathematics of Moscow State University, 2005. - p. 129.

5. Revyakin A.M., Bardushkina I. V. Mathematical methods of modeling in economics. Textbook. - M.: MIET, 2013 – - 328 p.

6. Recski A. Matroid theory and its applications in electric network theory and in statics. – Budapest: Akad. Kiado, 1989.

7. Bolker E.D., Crapo H. How to brace a one-story building? // Environment and Planning B. – 1977. – V. 4, P. 125–152.

MATHEMATICAL MODELS OF DATA PROCESSING UNDER THE MANIFOLD COORDINATE SYSTEMS

Riznyk V.V.

Lviv Polytechnic National University

Abstract. Mathematical models of data processing under the manifold coordinate systems based on multidimensional combinatorial configurations, namely the concept of Ideal Ring Bundles (IRBs) regarded. IRBs are cyclic sequences of positive integers, which form perfect partitions of a finite interval $[1, S]$ of integers. The sums of connected sub-sequences of an IRB enumerate the set of integers $[1, S]$ exactly R -times. Proposed concept involves techniques forming manifold coordinate grid $m_1 \times m_2 \times \dots \times m_t$ over an object that exists in dimension $t+1$ with t disjoint axes and common reference point of the grid grounded on a set of t -stage vectors as basis for data coding and processing under the manifold

coordinate system. A set of these vectors t -stage vectors and their summing modulo $\text{mod } m_1$, $\text{mod } m_2$, ..., $\text{mod } m_t$ enumerate node points grid of the manifold coordinate system with the corresponding sizes and dimensionality exactly R times. Besides, we require each indexed vector data "category-attribute" set mutually uniquely corresponds to the eponymous point set of the coordinate grid. Therefore, a combination of binary code with vector weight discharges of the database is allowed, and the set of all values of indexed vector data sets are the same that a set of numerical values. The model allows develop advanced vector information technologies under manifold coordinate systems.

In recent times, vector data-sets often involve a number of factors, such as national economic, national defense, and other indexes of infrastructure, which have led to difficulties in large-scale data processing a great number of new concepts, parallel algorithms, processing tools, platform, and applications are suggested and developed to improve the value of big vector data. Big spatial vector data management presented in global review [1]. A comprehensive survey of the existing techniques and technologies from both academia and industry given in [2]. Unlike customary coding designs, the vector data puts it in basis of t -dimensional manifold coordinate system, where basis is the smallest of the general number coordinate set, which it creates taking m_i modulo addition, $i=1,2,\dots,t$. This scientific method reduces to the indexing t -dimensional data in the manifold coordinate set basis of t -dimensional reference system, where the basis is a set of less than the total number of the point node coordinates set [3].

The known vector data coding methods [1–12] characterized by limited number of attributes and categories for signal coding and processing information encoded by two or more attributes and categories, as well as are high complexity. Mathematical problem is reduced to establishing a mutually unambiguous display of vector binary code combinations sets according to vector data attribute-categories sets on the coordinate grid of the t -dimensional surface of the t -manifold. The task is to increase the number of code combinations of t -dimensional binary code for the formation of information parameters of signals by the number of attributes and categories in the basis of the outlined t -dimensional coordinate system. We require the code combinations enumerate exactly R -times both vector data attribute-categories sets and node points set of the coordinate grid with

sizes $m_1 \times m_2 \times \dots \times m_i \times \dots \times m_t$, where m_i – number markers for referencing indexed categories on i -th axis, which corresponds to one of t attributes in the manifold coordinate system [13].

Mathematical model of t -dimensional vector data processing system under manifold coordinate system is a set of nodal dots of the t -dimensional spatial coordinate grid of fixed sizes, which one-to-one correspond to a set of t -dimensional vector data attributes and categories.

In the case of two- dimensional ($t=2$) vector data processing system is a coordinate grid of sizes $m_1 \times m_2$, which covers a surface of usual torus with two ($t=2$) concurred axes of the torus with $(0,0)$ reference point. (Fig.1).

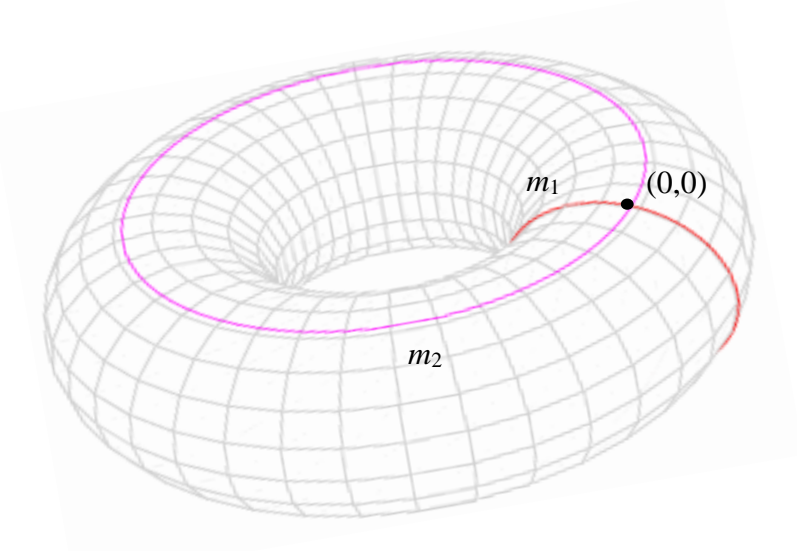


Fig.1. Two-dimensional ($t=2$) vector data processing system as a coordinate grid of sizes $m_1 \times m_2$ covered surface

Let us form toroidal coordinate system for vector data coding two ($t = 2$) attributes with two ($m_1 = 2$) categories of the first, and three ($m_2 = 3$) - the second attribute accordingly under the basis of

three ($n = 3$) two-dimensional ($t=2$) vectors $\{(1,1), (0,2), (0,1)\}$. The basis forms toroidal coordinate system by summing these vectors as a set of all its combinations taking modulo 2, and modulo 3 accordingly: $(0,0) \equiv (0,2) + (0,1) \pmod{2,3}$; $(1,0) \equiv (1,1) + (0,2) \pmod{2,3}$; $(1,2) \equiv (0,1) + (1,1) \pmod{2,3}$. So long as the terms $\{(1,1), (0,2), (0,1)\}$ of the 3-stage ($n=3$) ring sequence themselves are two-dimensional vector sums also, the set of the modular ($m_1=2, m_2=3$) vector sums forms a set of nodal points of annular reference grid over 2×3 exactly once ($R=1$). After arranged of all vectors, we derive the coordinate grid $m_1 \times m_2 = 2 \times 3$:

$(1,0) \equiv (1,1) + (0,2)$	$(1,1)$	$(1,2) \equiv (0,1) + (1,1)$
$(0,0) \equiv (0,2) + (0,1)$	$(0,1)$	$(0,2)$

The binary 2D code under the basis $\{(1,1), (0,2), (0,1)\}$ of toroidal coordinate system $m_1 \times m_2 = 2 \times 3$ given below (Table 1).

Table 1.

The binary 2D vector code under the basis $\{(1,1), (0,2), (0,1)\}$ of toroidal coordinate system $m_1 \times m_2 = 2 \times 3$

n/n	2D vector code combinations under the reference grid $m_1 \times m_2 = 2 \times 3$			
	Vector	Digit weights of the code		
		(0,1)	(0,2)	(1,1)
1	(0,0)	1	1	0
2	(0,1)	1	0	0
3	(0,2)	0	1	0
4	(1,0)	0	1	1
5	(1,1)	0	0	1
6	(1,2)	1	0	1

The underlying code is “perfect” toroidal code because only three ($n=3$) its code combinations make a set of $n (n-1) = 6$ combinations which make it possible to configure a toroidal coordinate system $n \times (n-1)$ over surface of torus, and the set of the code combinations

enumerates the set of node points of the reference grid exactly once ($R=1$).

The following example refers to the use of manifold coding system for storing and efficiently processing vector data indexed arrays in the form of 3D tables. To do this, the basis $\{(0,1,0),(0,2,3),(1,1,2),(0,2,2),(1,0,3),(1,1,1)\}$ of six ($n=6$) code combinations make it possible to shape a manifold coordinate system $m_1 \times m_2 \times m_3 = 2 \times 3 \times 5$ over surface of the manifold, and the set of the code combinations enumerates the set of node points of the reference grid exactly once ($R=1$). In calculations, it is necessary to take into account the values of modules $m_1=2$, $m_2=3$, $m_3=5$ (Table 2).

Table 2.

The binary 3D vector code under the basis $\{(0,1,0),(0,2,3),(1,1,2),(0,2,2),(1,0,3),(1,1,1)\}$ of manifold coordinate system $m_1 \times m_2 \times m_3 = 2 \times 3 \times 5$

n/n	3D vector code combinations under the reference grid $m_1 \times m_2 \times m_3 = 2 \times 3 \times 5$						
	Vector	Digit weights of the code					
		(0,1,0)	(0,2,3)	(1,1,2)	(0,2,2)	(1,0,3)	(1,1,1)
1	(0,0,0)	1	1	1	1	1	0
2	(0,0,1)	0	0	0	1	1	1
3	(0,0,2)	0	0	1	1	1	0
4	(0,0,3)	1	1	0	0	0	0
5	(0,0,4)	1	1	0	1	1	1
.....							
30	(1,2,4)	1	1	1	0	1	1

The result of the calculation forms the $2 \times 3 \times 5$ matrix which enumerates 3-modular three-dimensional vector-sums and each of its meets exactly once ($R=1$).

A rough drawing of manifold reference system about three ($t=3$) concurrent axes m_1 , m_2 , and m_3 with ground coordinate $(0,0,0)$ is given below as a developed projection of these axes on the plane of sketch (Fig.2).

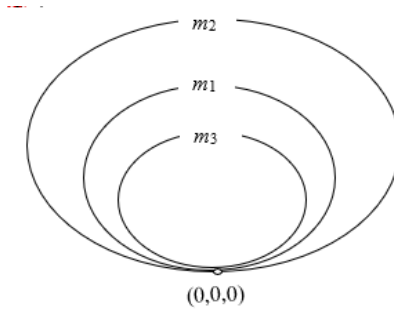


Fig.2 A rough drawing of manifold reference system about three ($t=3$) concurrent axes m_1 , m_2 , and m_3 with ground coordinate $(0,0,0)$

The underlying mathematical models of data processing under the manifold coordinate systems provides opportunities to apply them to configure suitable relation big vector data models, e.g., processing two or more categories of attributes simultaneously [13]. Each indexed vector data "category-attribute" set mutually uniquely corresponds to the eponymous point set of the coordinate grid. The models allow develop advanced vector information technologies under the toroidal or manifold coordinate systems.

There are a priori infinite set of manifold coordinate systems the underlying techniques can be used for easy-to-grasp representation and multidimensional big vector data processing under the systems. Advanced intelligent big vector data information technologies based on concept of multidimensional Ideal Ring Bundles provide competitive advantages with respect to processing speed and storage capacity due to vector IRB-coding. Proposed models of data processing under the manifold coordinate systems make it possible to improve the value of big vector data from both academia and industry.

References

1. Y. Xiaochuang, and Li Guoqing, "Big spatial vector data management: a review," in *Big Earth Data*, vol.2, No1, 108-129, DOI: 10.1080/20964471.2018.1432115
2. N.R.Bose, *Applied Multidimensional Systems Theory*. Springer Int. Publishing AG 2017, Pensilvania State University State College, PA, USA, 192 p.
3. M. Chi, A.Plaza, .A. Benediktsson, Z. Sun, J. Shen, and Y. Zhu, "Big data for remote sensing: Challenges and opportunities," *IEEE*, 104 (11), 2207–2219, DOI:10.1109/Jproc.2016.598228
4. Y.Ma, H. Wu, L. Wang, B.Huang, R. Ranjan, A. Zomaya, and W. Jie, "Remote sensing big data computing: Challenges and opportunities," *Future Generation Computer Systems*, 51, 47–60, 2015 doi:10.1016/j.future.2014.10.029
5. M.K. Pekturk, and M. Unal, "A review on real-time big data analysis in remote sensing applications," *25th Signal Processing and Communications Applications Conference (SIU)*, Antalya, Turkey, May 15–18, 2017
6. A.Eldawy, M.F. Mokbel, S. Alharthi, A. Alzaidy, K. Tarek, and S. Ghani. SHAHED, "A MapReduce-based system for querying and visualizing spatio-temporal satellite data," *IEEE International Conference on Data Engineering*, Seoul, South Korea, April 13–17, 2015
7. S.Ye, T. Yan, Y. Yue, W. Lin, L. Li, X.Yao, D. Zhu, "Developing a reversible rapid coordinate transformation model for the cylindrical projection," *Computers & Geosciences*, 89, 44–56. doi:10.1016/j.cageo.2016.01.007
8. Z.Wu, J.Feng, F. Qiao, and Z. Tan - M. (2016),"Fast multidimensional ensemble empirical mode decomposition for the analysis of big spatio-temporal datasets," *Philos Trans A Math Phys Eng Sci*, 374(2065), 2015.01.97
9. S. Ray, B. Simion, A.D. Brown, and R. Johnson, "A parallel spatial data analysis infrastructure for the cloud," *ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, Orlando, FL, USA, November 5–8, 2013

10. W.Tang, and W. Feng, "Parallel map projection of vector-based big spatial data: Coupling cloud computing with graphics processing units," Computers, Environment and Urban Systems, 61, 187–197. doi:10.1016/j.compenvurbsys.2014.01.001
11. Eldawy, A., Mokbel, M. F., & Jonathan, C. (2016). HadoopViz: A MapReduce framework for extensible visualization of big spatial data. Paper presented at the 32nd IEEE International Conference on Data Engineering, Helsinki, Finland, May 16–20.
12. Delaney, The importance of "Big Data": A Definition. Gartner, 2012.
13. Riznyk V."Method of Big Vector Data Processing under Toroidal Coordinate Systems", (CSIT 2020), pp.105-108. DOI: 10.1109/CSIT49958.2020.9321955

ON THE KAMACHI-ARUMUGAM CONJECTURE ON DISTANCE ANTIMAGIC LABELING OF GRAPHS

Semeniuta M.F., Sherman Z.A.

*National Aviation University
Donetsk National Medical University*

Abstract. Graphs obtained as a result of connection operation on certain types of graphs are investigated for distance antimagic graphs.

Arumugam S. and Kamatchi N. introduced in [1] the concept of remote antimagic labeling of the graph in 2013.

Let $G = (V, E)$ be an undirected graph of order n with neither loops nor multiple edges. Let us consider a function $f: V \rightarrow \{1, 2, \dots, n\}$, given on the set of vertices $V(G)$. The weight $w(x)$ of vertex x is understood as the sum of labels of vertices adjacent to x , then $w_f(x) = \sum_{z \in N(x)} f(z)$, where $N(x)$ is a set of adjacency of vertex x . Function $f: V \rightarrow \{1, 2, \dots, n\}$ is called *distance magic labeling* of graph G , if function f is bijection and

$w_f(x) \neq w_f(y)$ for every two vertices $x, y \in V(G)$. The graphs for which such labeling is possible are called *distance antimagic*. In work [1] it was supposed that the graph G be *distance antimagic* if and only if when $N(x) \neq N(y)$ for any vertices $x, y \in V(G)$. In general, this hypothesis has not yet been proven and refuted, although there are many works devoted to it. More information about it can be found in [2]. Let us study antimagic properties of graphs, obtained by performing a join operation on certain types of graphs and $(1, Q)$ -split graphs.

According to Zykov, *connection of graphs* G and H such as that $V(G) \cap V(H) = \emptyset$, is called a graph $G + H$ with a set of vertices $V = V(G) \cup V(H)$ and set of edges $E = E(G) \cup E(H) \cup \{ax : a \in V(G), x \in V(H)\}$. If $V(G) \cap V(H) \neq \emptyset$, then before the connection operation one remarks the vertices of the operand graphs so that the new sets of their vertices do not have elements in common.

Lemma 1. Graph $K_m + K_n$ is *distance antimagic* for every integer numbers m and n .

Proof. The result of the operation of joining complete graphs is a complete graph, i.e $K_m + K_n = K_{m+n}$. This graph is a distance antimagic for any integer number m i n . The lemma has been proved.

Theorem 1. Graph $K_m + G$ is distance antimagic, if G or $G - x$ is distance antimagic graph and $\Delta(G) < n - 1$, where $m > 1$, x is isolated vertex in G , $\Delta(G)$ is maximum degree of the graph G of order n .

Proof. Suppose, that φ is distance antimagic labeling of graph G of order n . We will denote the vertices of graph K_m as x_1, x_2, \dots, x_m , and vertices of G as y_1, y_2, \dots, y_n . Let us define the labeling f of graph $K_m + G$ as

$$f(x_i) = i + n, \quad f(y_j) = \varphi(y_j),$$

where $i = 1, 2, \dots, m, j = 1, 2, \dots, n$. Function f is a bijection from the set $V(K_m + G)$ on a set of numbers $\{1, 2, \dots, mn\}$. We will define weight of vertices as following:

$$w_f(x_i) = \frac{m(m+1)}{2} + mn - i + \frac{n(n+1)}{2},$$

$$w_f(y_j) = w_\varphi(y_j) + \frac{m(m+1)}{2} + mn.$$

According to the condition $\Delta(G) < n - 1$, then $w_f(x_i) \neq w_f(y_j)$. Thus, function f is distance antimagic labeling of graph $K_m + G$. The theorem has been proved.

Obviously, if equality holds $N(x) = N(y)$ for $x, y \in V(G)$, then G does not allow distance antimagic labeling. This statement is a necessary condition for the existence of the specified labeling and it proves corollary 1 of theorem 2.

Corollary 1. Graph $G + H$ does not allow distance antimagic labeling, if one of the graphs G or H does not hold more than one isolated vertices.

Corollary 2. Graph $K_m + G$ is not distance antimagic, if G is not complete graph, $\Delta(G) = n - 1$ and $m \leq n$, where $m > 1$, $\Delta(G)$ is maximum degree of graph G of order n .

Proof. Suppose, there is distance antimagic labeling f of graph $K_m + G$ and the vertices of graph G are corresponding to numbers from a set $\{1, 2, \dots, n\}$. Let denote the vertices of K_m as x_1, x_2, \dots, x_m , and the vertices of G as y_1, y_2, \dots, y_n . Let vertex y_s

have maximum degree in graph G , then $w_f(y_s) = \frac{n(n+1)}{2} - s$.

Equation $w_f(x_i) = w_f(y_s)$ for weight of vertices x_i and y_s is performed when $i = s$, where $i \leq n$. We have come to a contradiction. The corollary has been proved.

References

1. Arumugam S., Kamatchi N. Distance antimagic graphs. *Journal of Combinatorial Mathematics and Combinatorial Computing*. 2013. Vol.84. P. 61-67.
2. Gallian J. A. A dynamic survey of graph labeling. *The Electronic Journal of Combinatorics*. 2019. DS6: Dec 15. 535 p. DOI: <https://doi.org/10.37236/27>

ANALOGUE OF COMBINATORY KOGANOV IDENTITY FOR ORDERED BELL NUMBERS

Voblyi V.A.

VINITI RAS

Abstract. We prove combinatorial identity with Stirling numbers of the first kind and ordered Bell numbers, which is an analogue of the combinatorial Koganov identity.

We denote by $s(n, k)$ – Stirling numbers of the 1st kind, and by $B(k)$ – Bell numbers. In [1] L. M. Koganov proved the combinatorial identity

$$\sum_k^n s(n, k)B(k) = 1 . \quad (1)$$

Let $S(n, k)$ be Stirling numbers of the 2nd kind, and $O(k)$ be ordered Bell numbers, which are also called Fubini numbers [2, 3, 4]. The ordered Bell numbers were introduced by Cayley in 1859 to enumerate some plane trees with ordered pendant vertices. For $O(k)$, the expression (2) is known.

$$O(k) = \sum_{j=0}^k S(k, j)j! . \quad (2)$$

In the report, we obtaine an analog of the identity (1) for the numbers $O(k)$.

Theorem. For $n \geq 0$, the combinatorial identity is true

$$\sum_{k=0}^n s(n, k) O(k) = n! . \quad (3)$$

Proof. The Stirling numbers $s(n, k)$, $S(n, k)$ satisfy the orthogonality relation [5, p. 47]

$$\sum_{k=j}^n s(n, k) S(k, j) = \delta_{nj} ,$$

where δ_{nj} is the Kronecker symbol.

Substituting (2) in the left part of (3) and changing the order of summation, we get

$$\begin{aligned} \sum_{k=0}^n s(n, k) O(k) &= \sum_{k=0}^n s(n, k) \sum_{j=0}^k S(k, j) j! = \\ &= \sum_{j=0}^n j! \sum_{k=j}^n s(n, k) S(k, j) = \sum_{j=0}^n j! \delta_{nj} = n! . \end{aligned}$$

(Note that $S(k, j) = 0$ for $k < j$).

The proof is completed.

References

1. Koganov L. M. On the number of pairs of independent partitions of a finite set. In "Combinatorial and asymptotic analysis", Krasnoyarsk, 1975, 71-79.
2. Trojovsky P. On a triangular inversion formula, its application and ordered Bell numbers. International Journal of Pure and Applied Mathematics V.2, No.2, 2006, 149-159.

3. Sprugnoli R. Riordan arrays and combinatorial sums. Discrete Mathematics 132(1994), 267-290.
4. Comtet L. Advanced Combinatorics. Dordrecht, Reidel, 1974.
5. Sachkov V.N. Introduction to combinatorial discrete methods mathematics. Moscow: MCCME, 2004.

ЗАСТОСУВАННЯ МЕТОДУ ПАТЕРНІВ ДО АНАЛІЗУ ДИНАМІКИ ПОКАЗНИКІВ РОЗВИТКУ СОЦІАЛЬНО- ЕКОНОМІЧНИХ СИСТЕМ

Бакурова А.В., Сластніков О.П., Кондратов Д.О.
Національний університет «Запорізька політехніка»

THE PATTERN METHOD APPLICATION TO THE ANALYSIS OF THE DYNAMICS INDICATORS OF DEVELOPMENT OF SOCIO-ECONOMIC SYSTEMS

Abstract. The subject of research of this work is the application of the method of patterns to the analysis of structural features of financing and dynamics of development of communities in the South of Ukraine. The advantages of the pattern method are the ability to detect hidden connections, standard patterns and visualize them. Based on the method of patterns and the principles of system dynamics, a method for predicting the type of dynamics of newly formed communities according to their patterns of funding structure is proposed.

1. Вступ

Розвиток соціально-економічних систем визначається рівнем їхнього фінансового забезпечення. Процеси децентралізації що відбуваються в Україні, призводять до складної структури джерел фінансування на рівні об'єднаних територіальних громад (ОТГ): трансферти, субвенції, дотації, міжбюджетні трансферти тощо. Предмет дослідження даної роботи складає застосування методу патернів до аналізу структурних особливостей фінансування та динаміки розвитку ОТГ областей Півдня України. Метод патернів відноситься до методів Data Mining, що дозволяють виявляти приховані

зв'язки, стандартні закономірності та візуалізувати їх. Поняття патерна розглядається як стійка комбінація суттєвих змінних, що пов'язані з характерною моделлю поведінки багатовимірних систем, та є емпіричним аналогом понять «тип», «кластер» [1-2].

2. Результати застосування методу патернів до аналізу розвитку ОТГ

Для аналізу було побудовано патерни двох видів: структури фінансування та динаміки розвитку. Патерни структури фінансування будувались на основі таких показників як:

1. - Власні доходи на 1 особу, грн.;
2. - Видатки розвитку (капітальні видатки) на 1 особу, грн.;

3. - Інфраструктурна субвенція на 1 особу, грн.;

4. - Базова/реверсна дотація на 1 особу, грн.

Патерни динаміки розвитку мали такі базові показники:

C_1 . - Темп зростання власних доходів на 1 особу, %;

C_2 . - Темп зростання видатків розвитку (капітальні видатки) на 1 особу, %;

C_3 . - Темп зростання інфраструктурної субвенції на 1 особу, %;

C_4 . - Дохід на душу населення ОТГ, грн.

Всі базисні показники було нормовано від 0 до 1.

Зазвичай на основі певної мови патернів будуються стандартні підходи до дослідження даних з використанням патернів. В нашому випадку загальна кількість патернів кожного виду визначається комбінацією трьох напрямів в структурі зв'язків між чотирма базисними показниками $\{\rightarrow, \uparrow, \downarrow\}$, тобто загалом маємо $4!=24$ різних упорядкування базисних показників, які можуть утворювати максимально (але не обов'язково) $3^3=27$ типів патернів (кластерів). Далі зупинимось на введеному вище впорядкуванні базисних показників від 1 до 4 (або від C_1 до C_4).

Візуалізація патернів виконана за допомогою методу паралельних координат в чотиримірному просторі, де координати відповідають базовим показникам, для патернів структури фінансування та патернів динаміки розвитку окремо. Метод паралельних координат реалізовано в пакеті MatLab [3] за відкритими даними [4].

Для прогнозування типу динаміки розвитку новоутворених ОТГ запропоновано методику, що ґрунтується на принципах системної динаміки [5]: структура системи визначає її динаміку; та принципи історизму - повторна конфігурація даних в динаміці призводить до аналогічного результату. Основні кроки методики такі.

Крок 1. Побудова діаграм патернів структури фінансування і динаміки розвитку в паралельних координатах. На рис.1 та рис.2 наведено приклади візуалізації патернів за методом паралельних координат.

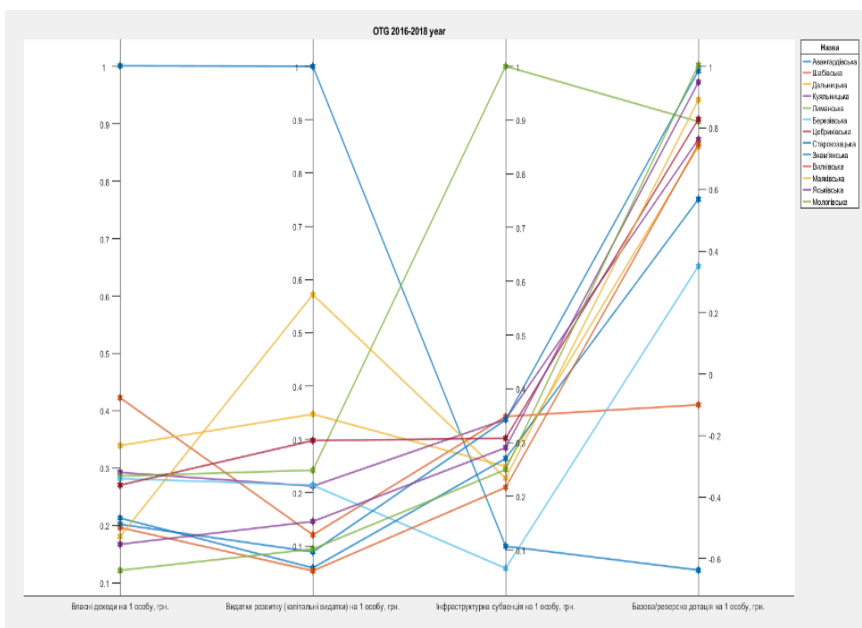
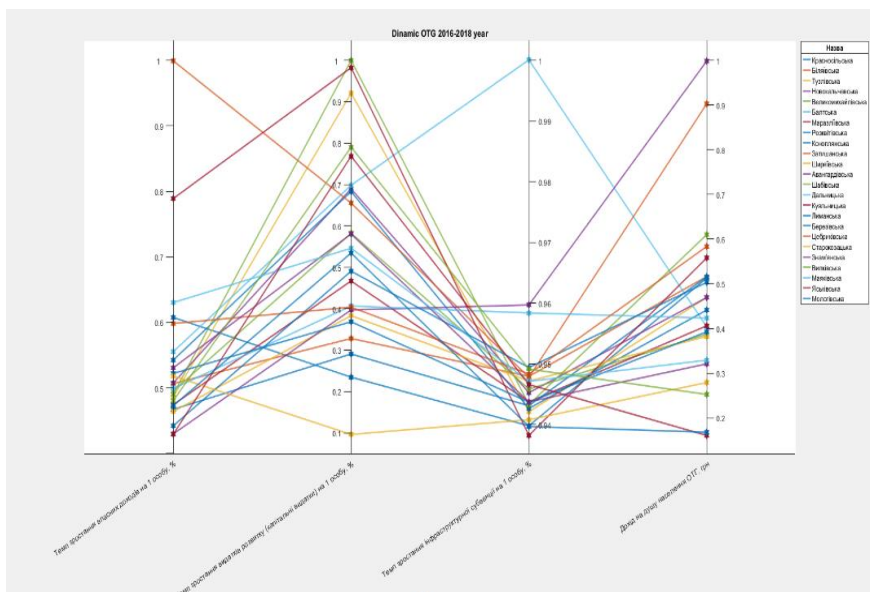


Рис.1. Патерни структури фінансування у 2018 р.
Одеських ОТГ, що були утворені в 2016 р.



Крок 2. Кодування діаграм послідовністю трьох напрямів в структурі зв'язків між чотирма базисними показниками $\{\rightarrow, \uparrow, \downarrow\}$.

Крок 3. Визначення кластерів, що складаються з однакових пар послідовностей зв'язків між чотирма базисними показниками структури фінансування та відповідними показниками динаміки розвитку.

Крок 4. Побудова таблиці співставлення патернів структури фінансування і динаміки розвитку.

Крок 5. Робиться висновок про тип динаміки новоутворених ОТГ за їхньою структурою фінансування та аналогічній структурі раніше утвореної ОТГ.

3. Висновки та обговорення

Запропонована методика прогнозування дозволяє зробити припущення про тип динаміки новоутворених в 2019 та 2020 роках ОТГ півдня України за їх патернами структури

фінансування, що мають аналоги серед ОТГ, утворених в 2015-2018 роках та зробити фінансові планування на наступні періоди.

Робота виконана в рамках договору про співпрацю між Національним університетом «Києво-Могилянська академія» та Національним університетом «Запорізька політехніка», ДБ05038 «Математичне моделювання соціально-економічних процесів та систем» (2018-2021pp).

Література

1. Алескеров Ф.Т., Белоусова В.Ю., Егорова Л.В., Миркин Б.Г. Анализ паттернов в статике и динамике, часть 1: Обзор литературы и уточнение понятия // Бизнес-информатика. – 2013. – № 3 (25). – С. 3-18.
2. Алескеров Ф.Т., Белоусова В.Ю., Егорова Л.В., Миркин Б.Г. Анализ паттернов в статике и динамике, часть 2: Примеры применения к анализу социально-экономических процессов// Бизнес-информатика. – 2013. – № 4 (26). – С. 3-20.
3. Mathworks, 2019. MATLAB*-, 2019. URL: <https://www.mathworks.com/campaigns/products>.
4. Інформаційні джерела відкритих даних: decentralization.gov.ua та openbudget.gov.ua.
5. Forrester J. W. Policies, decisions and information sources for modeling / J. W. Forrester// European Journal of Operational Research. - 1992. - Vol. 59, Issue 1. - P. 42-63.

АРИФМЕТИКА БІТКОІНА У ПРИКЛАДАХ

Бондар О. П.,

Льотна академія Національного авіаційного
університету

Abstract. Examples of numbers and mathematical operations, which are the basis of bitcoin formation, are considered. The examples show the use of arithmetic modulo. Its comparison with arithmetic over the field of real numbers is given. This provides an opportunity, especially for mathematicians, to gain an idea of the principles of formation of cryptocurrencies.

Наша мета – дати користувачам криптовалют, переважно, не математикам, поняття про математичне підґрунтя створення криптовалют на елементарних прикладах засад формування біткоіну.

Математичними аспектами формування і функціонування криптовалют вважається еліптична криптографія – розділ криптографії, який ґрунтується на еліптичних кривих над скінченними полями.

Для формування біткоіну і операцій з ним використовується криптографічний алгоритм ECDSA (Elliptic Curve Digital Signature Algorithm) з еліптичними кривими (elliptic curve) і скінченними полями (finite field).

Протокол біткоіну містить фіксований для всіх користувачів протоколу набір параметрів:

- $y^2 = x^3 + 7$ рівняння еліптичної кривої,
- m - основний модуль поля,
- P – базову точку на кривій,
- k – порядок базової точки.

Перші три параметри задаються у протоколі і не залежать один від одного. Порядок k базової точки є функцією цих параметрів, тобто визначається ними.

Всі програми ECDSA, пов'язані з біткоіном, використовують дуже великі числа для базової точки, простого модуля та порядку.

Вважається, що ці великі значення гарантують безпеку алгоритму. Чому і ким обрано саме ці числа? Однозначної відповіді на ці питання немає [1].

Наразі відомо, що вказана конкретна реалізація під назвою secp256k1 є частиною сімейства рішень еліптичної кривої над скінченними полями, запропонованих для використання в криптографії.

Для наочності уявлення про еліптичну криву, базову точку, простий модуль та порядок базової точки розглянемо елементарні приклади цих параметрів на невеликих числах.

Приклад 1. Поле лишків (рос. вычетов) за модулем 11 є скінченним з 11 елементами:

$$F_{11} = \{0, 1, 2, \dots, 10\}.$$

В цьому полі, наприклад, $27 \pmod{11} = 5$, а $-27 \pmod{11} = 6$.

Перевагами використання модульної арифметики в криптографії є висока швидкість обробки інформації при малому споживанні енергії, а також підвищення стійкості процесорів до можливих помилок.

Приклад 2. Обернений до числа 3 елемент за модулем 11:

$$3^{-1} \pmod{11} = 4,$$

тому що за означенням оберненого елемента

$$3 \pmod{11} * 3^{-1} \pmod{11} = 3 * 4 = 12, \text{ а } 12 \pmod{11} = 1.$$

Канонічне рівняння еліптичної кривої

$$y^2 = x^3 + ax + b$$

при $a=0$ є еліптичною кривою протоколу біткоїна над скінченним полем F_m

$$y^2 = x^3 + 7 \pmod{m}.$$

Зображується крива окремими точками.

Приклад 3. Координати точок еліптичної кривої

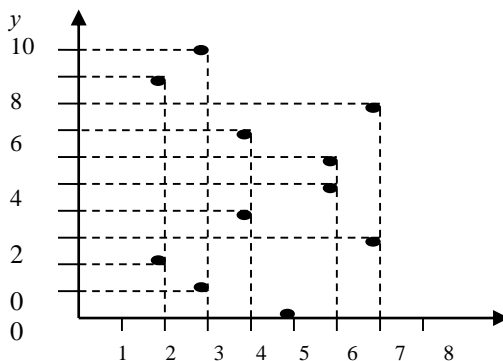
$$y^2 = x^3 + 7 \pmod{11}$$

отримано з розрахунку:

x	0	1	2	3	4	5	6	7	8	9	10
$x^3 + 7 \pmod{11}$	7	8	4	1	5	0	3	9	2	10	6
y	0	1	2	3	4	5	6	7	8	9	10
$y^2 \pmod{11}$	0	1	4	9	5	3	3	5	9	4	1

Відтак, точками даної еліптичної кривої за модулем 11 є точки:

(2, 2), (2, 9), (3, 1), (3, 10), (4, 4), (4, 7), (5, 0), (6, 5), (6, 6), (7, 3), (7, 8). Графіком є множина цих точок:



Не сингулярні криві мають застосування в криптографії завдяки своїм особливостям, які дають можливість побудови групи точок цих кривих з операціями суми і, як наслідок, подвоєння.

Сумою двох точок (x_1, y_1) і (x_2, y_2) є точка (x_3, y_3) , для якої

$$x_3 = t^2 - x_1 - x_2 \pmod{m}, \text{ і } y_3 = t(x_1 - x_3) - y_1 \pmod{m},$$

де $t = (y_2 - y_1) / (x_2 - x_1) \pmod{m}$.

Приклад 5. Сума двох точок за модулем 11:

$$(4, 4) + (2, 9) \pmod{11} = (3, 10),$$

де $t = (9 - 4) / (2 - 4) = -5/2 = 3 \pmod{11}$.

В результаті подвоєння точки (x_1, y_1) отримаємо точку (x_3, y_3) :

$$x_3 = t^2 - 2x_1 \pmod{m}, \quad y_3 = t(x_1 - x_3) - y_1 \pmod{m},$$

де $t = 3x_1^2 / (2y_1) \pmod{m}$.

Приклад 6. Подвоєна точка за модулем 11

$$2(3, 1) \pmod{11} = (3, 10),$$

де $t = 3 \cdot 9 / (2 \cdot 1) = 27/2 = 8 \pmod{11}$.

Поєднання операцій знаходження суми і подвоєння називають скалярним множенням. Наприклад, точку P можна помножити на 6 додаванням її до себе 6 разів: $6P = P + (P + (P + (P + (P + (P + P))))$ або за допомогою комбінації операцій додавання точок та їх подвоєння:

$$6P = 2(P + 2P).$$

Нейтральним елементом групи точок еліптичної кривої беруть так звану «точку на нескінченності» $O(x, \infty)$, для якої

$$P + (-P) = O.$$

Над полем дійсних чисел – це точка «перетину» прямої, що проходить через $(x, 0)$ паралельно осі ординат, з еліптичною кривою. Над полем F_m – це сума точок (x, y_1) і (x, y_2) , симетричних відносно прямої $y = m/2$.

За допомогою зазначених у групі точок еліптичної кривої операцій визначається порядок точки на кривій – це найменше ціле число k , при якому $kP = O(x; \infty)$.

Приклад 7. Точка $(3, 1)$ еліптичної кривої

$$y^2 = x^3 + 7 \pmod{11}$$

має порядок $k = 3$, оскільки

$$3(3, 1) = (3, 1) + 2(3, 1) = (3, 1) + (3, 10) = O(x; \infty).$$

Приклад 8. Точка $(7, 3)$ еліптичної кривої

$$y^2 = x^3 + 7 \pmod{11}$$

має порядок $k = 12$, що можна отримати з результатів скалярного множення точки $(7, 3)$ на 12.

Розглянуті нами приклади математичних основ формування біткоіну мають дати поняття про криптовалюту, показати окремі математичні взаємозв'язки, що існують між деякими її параметрами. А це, зокрема, повинно надати, в першу чергу, не математикам можливість і бажання розуміти закони функціонування сучасного фінансового ринку [2].

Література

1. An Introduction to Bitcoin, Elliptic Curves and the Mathematics of ECDSA. URL: <https://www.slideshare.net/NikeshMistry1/introduction-to-bitcoin-and-ecdsa> (дата звернення: 12.03.2021).
2. Модифікація конкурентних механізмів на глобальному фінансовому ринку в умовах діджиталізації. URL: <https://galicianvisnyk.tntu.edu.ua/?art=743> (дата звернення: 12.03.2021).

ТРЕСНОМІАЛЬНИЙ РОЗПОДІЛ

Волков Ю.І., Войналович Н.М.,

Центральноукраїнський державний педагогічний університет
імені Володимира Винниченка

TRESNOMIAL PROBABILITY

Abstract. We enter the concept of tresnomial distribution that is one generalizations the binomial distribution and we his properties study.

Keywords: distribution, binomial, tressnomial, expectation, variance.

Нехай $w(y) = 1 + 2ay + y^2$, $a > 0$, $y > 0$, і нехай ξ випадкова величина, яка може приймати три значення 0, 1, 2, відповідно з ймовірностями

$$\Pr(\xi = 0) = \frac{1}{w(y)}, \quad \Pr(\xi = 1) = \frac{2ay}{w(y)}, \quad \Pr(\xi = 2) = \frac{y^2}{w(y)}.$$

Генератрисою такої випадкової величини буде функція

$$P(z) = \frac{w(yz)}{w(y)}. \quad \text{Тому математичне сподівання} \quad E\xi = \frac{d}{dz} P(z) \Big|_{z=1} = \frac{yw'(y)}{w(y)}.$$

. Дисперсія $D\xi = y \frac{d}{dy} E\xi$.

Розглянемо іншу параметризацію (х-ову), яка визначається

співвідношенням $E\xi = 2x = \frac{yw'(y)}{w(y)}$, $0 < x < 1$. Звідси

$$y = y(x) = \frac{2ax - a + \sqrt{1 + (a^2 - 1)(2x - 1)^2}}{2(1 - x)}.$$

Дисперсія

$$D\xi = y \frac{d}{dy} E\xi = \frac{y(x)}{dy(x)/dx} = 2v(x, a),$$

де

$$v(x, a) := 2(b(x, a) + x - 2x^2), \quad b(x, a) := x - \frac{2ax(1 - x)}{a + \sqrt{1 + (a^2 - 1)(2x - 1)^2}}.$$

..

Отриманий розподіл називатимемо *елементарним тресноміальним розподілом* з параметрами x і a .

Означення. Будемо говорити, що випадкова величина ξ має тресноміальний розподіл з параметрами n, x, a , якщо вона є сумою n незалежних випадкових величин, кожна з яких має елементарний тресноміальний розподіл з параметром a .

Якщо $a=1$, то отримаємо біноміальний розподіл

Генератрисою тресноміального розподілу буде функція

$$P(z) = \frac{(1 + 2ayz + y^2 z^2)^n}{w(y)^n} =$$

$$\sum_{k=0}^{2n} C(n, k, a) b(x, a)^{k/2} b(1-x, a)^{n-k/2} z^k, k = 0, 1, 2, \dots, 2n, \text{ де}$$

$$C(n, k, a) := \sum_{j=0}^{[k/2]} \frac{n!}{j!(k-2j)!(n+j-k)!} (2a)^{k-2j} = \sum_{j=0}^{[k/2]} \binom{n}{j} \binom{n-j}{k-3j} (2a)^{k-2j}$$

Отже, якщо випадкова величина ξ має тресноміальний розподіл з параметрами (n, x, a) , то

$$\Pr(\xi = k) = C(n, k, a) b(x, a)^{k/2} b(1-x, a)^{n-k/2}, k = 0, 1, 2, \dots, 2n.$$

Це записуватимемо так: $\xi \in \text{Tres}(n, x, a)$.

Встановлено такі результати.

Лема 1. *Нехай*

$$v(x, a) := x(1-2x) + b(x, a) = 2x(1-x) \left(1 - \frac{a}{a + \sqrt{1 + (a^2 - 1)(2x-1)^2}} \right).$$

Тоді

$$v(x, a) \frac{d}{dx} \left(b(x, a)^{k/2} b(1-x, a)^{n-k/2} \right) =$$

$$(k - 2nx) b(x, a)^{k/2} b(1-x, a)^{n-k/2}$$

Теорема 1. *Нехай*

$$s_m(c, x) :=$$

$$\sum_{k=0}^{2n} (k-c)^m C(n, k, a) b(x, a)^{k/2} b(1-x, a)^{n-k/2}, m = 0, 1, 2, \dots$$

Тоді

$$s_{m+1}(c, x) = v(x, a) \frac{d}{dx} s_m(c, x) + (2nx - c) s_m(c, x), s_0(c, x) = 1$$

Лема 2. Рівняння $\frac{b(1-x, a)}{b(x, a)} = c^2, 0 \leq x \leq 1$, має такий

розв'язок $x = \frac{1+ac}{1+2ac+c^2}$.

Це випливає з того, що: $b(1-x, a) = b(x, a)c^2 \Rightarrow$

$$1-x - \frac{2ax(1-x)}{a + \sqrt{1+(a^2-1)(2x-1)^2}} - c^2 \left(x - \frac{2ax(1-x)}{a + \sqrt{1+(a^2-1)(2x-1)^2}} \right) = 0$$

Теорема 2. Нехай $x_k, k = 0, 1, \dots, 2n-1$, розв'язки рівняння

$$\Pr(\xi = k+1) = \Pr(\xi = k).$$

Тоді

$$x_k = \frac{C(n, k, a)(C(n, k, a) + aC(n, k+1, a))}{C(n, k, a)^2 + 2aC(n, k, a)C(n, k+1, a) + C(n, k+1, a)^2}.$$

і якщо $x \in (x_{k-1}, x_k)$, то найбільш ймовірним значенням в.в. ξ буде число k , якщо $x = x_k$, то найбільш ймовірними значеннями в.в. ξ будуть числа k і $k+1$.

Доведення.

Розглянемо

відношення

$$\frac{\Pr(\xi = k+1)}{\Pr(\xi = k)} = \frac{C(n, k+1, a)}{C(n, k, a)} \frac{\sqrt{b(x, a)}}{\sqrt{b(1-x, a)}}.$$

Якщо це відношення менше 1, то послідовність $\Pr(\xi = k)$ зростає, якщо більше 1, то спадає, якщо дорівнює 1, то $\Pr(\xi = k) = \Pr(\xi = k+1)$, тобто, $\frac{b(1-x, a)}{b(x, a)} = \frac{C(n, k+1, a)}{C(n, k, a)}, 0 \leq x \leq 1$.

Теорему 2 тоді отримаємо, якщо в лемі 2 взяти

$$c = \frac{C(n, k+1, a)}{C(n, k, a)}.$$

Наприклад, нехай $\xi \in \text{Tres}(3, 0.7, 1.2)$. Тоді найбільш ймовірним значенням в.в. ξ буде число 5, бо $x_4 = 0.551181, x_5 = 0.714286 \{ \Pr(\xi = 5) = 0.283511 \dots \}$.

Нехай $\xi \in \text{Tres}(4, 40179/70883, 2)$. Тоді найбільш ймовірними значеннями в.в. ξ будуть числа 4 і 5, бо $x_4 = 40179/70883 \{ \Pr(\xi = 4) = \Pr(\xi = 5) = 0.314688 \dots \}$.

Про застосування тресноміального розподілу в теорії наближення функцій див [1].

Література

1. Ю.И.Волков, Обобщенное преобразование Безье, *Матем. заметки*, 2015, том 97, выпуск 5, 655-664.

ПРО ПАРКЕТИ НА ДЕЯКИХ ПОВЕРХНЯХ

Глухов О.Д.

Національний авіаційний університет

ON THE TILLING OF SOME SURFACES

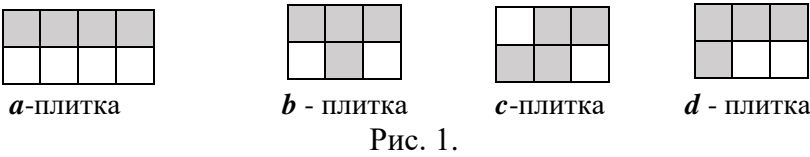
Abstract. The article deals with the tilings of a rectangle, torus, Moebius strip and Klein bottle

Під паркетом (tiling, tessellation) будемо розуміти таке покриття деякої поверхні плитками -многокутниками, в якому будь-які два многокутники мають або спільну сторону, або тільки спільну вершину, або зовсім не мають спільних точок [1].

Нехай n, m -натуральні числа, будемо позначати через $R(n, m)$ прямокутник зі сторонами довжини n і m , який розбитий на $n \times m$ елементарних квадратів (розміру 1×1), а через $T(n, m)$, $M(n, m)$, $K(n, m)$ - склеєні з нього тор, стрічку Мебіуса та пляшку Клейна, відповідно.

В даній роботі розглядається задача замощення поверхонь $T(n, m)$, $M(n, m)$, $K(n, m)$ плитками, які складаються з 4-х елементарних квадратів. Точніше, буде розглядатися питання

існування паркетів, які складаються з однакових плиток наступних чотирьох типів: *a*, *b*, *c*, *d* (рис.1).



Щодо *a*-плитки, то відомо, що паркет на прямокутнику і також на торі існує, тільки у випадку, коли хоча б одне з чисел *n*, *m* кратне 4 [2, 3]. Також очевидно, що жоден з даних чотирьох паркетів *a*, *b*, *c*, *d*, не існує, якщо *n*×*m* не кратне 4. Тому в подальшому будемо розглянути випадок: *n* = 4*s*+2, *m* = 4*t*+2.

Результати для прямокутника (*R*), тора (*T*), стрічки Мебіуса (*M*) та пляшки Клейна (*K*) у випадку *n* ≡ *m* ≡ 2(mod4) зібрані в наступній таблиці.

Тип плитки	Існування паркету			
	R	T	M	K
<i>a</i>	ні	ні	так	так
<i>b</i>	ні	ні	так	так
<i>c</i>	ні	так	ні	так
<i>d</i>	ні	ні	ні	ні

Якщо відповідний паркет існує, його звичайно нескладно побудувати. Так, на рисунках 2 і 3 показані приклади *a*-паркету і *b*-паркету на стрічці Мебіуса, а на рисунку 4 - *c*-паркет на торі при *n*=*m*=10 (стрілки показують орієнтацію при склейці протилежних сторін прямокутника).

Доведення того, що паркет даного типу на заданій поверхні не існує базується на правильному розфарбуванні поверхні на два кольори, тобто такому, що кількість елементарних квадратів, розфарбованих кожним кольором, однакова.

При заданому правильному розфарбуванні поверхні плитку будемо називати збалансованою, якщо число її елементарних квадратів, розфарбованих одним кольором парне (4 і 0 або 2 і 2) і незбалансованою в іншому разі (1 і 3).

Очевидно, що паркет даного типу не існує, якщо можливе таке правильне розфарбування поверхні, при якому число незбалансованих плиток буде непарним. Зауважимо, що оскільки жодне з чисел n , m не кратне 4, то загальне число плиток завжди буде непарним.

На рисунках 5, 6 наведені правильні розфарбування стрічки Мебіуса, які показують неможливість існування c -паркету та d -паркету – відповідно. Наприклад, на рис. 6 одразу видно, що усі 25 d -плиток будуть незбалансованими.

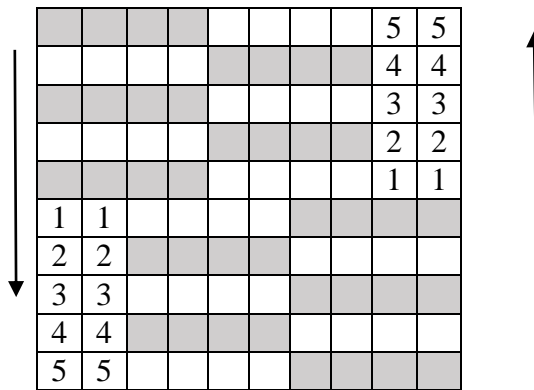


Рис. 2.

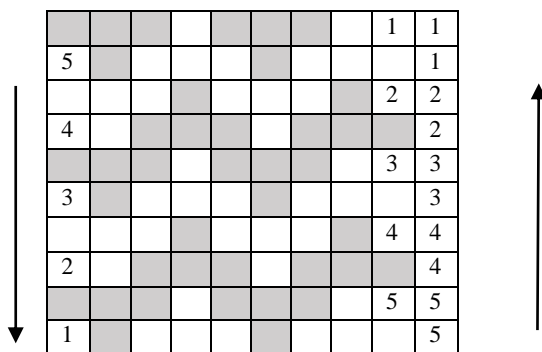


Рис.3.

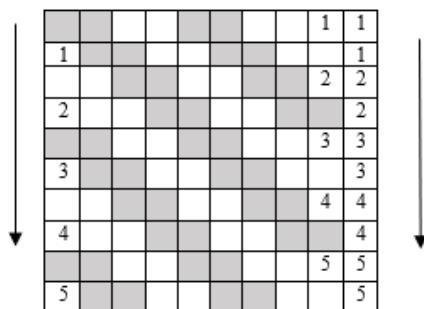


Рис.4.

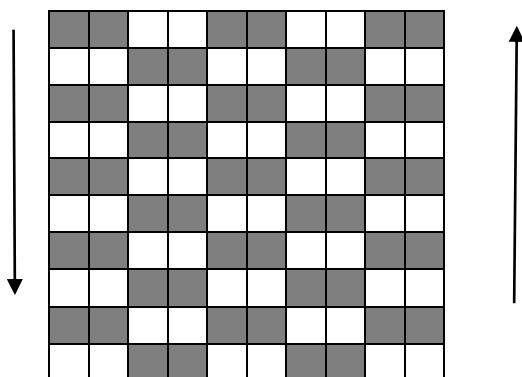


Рис.5.

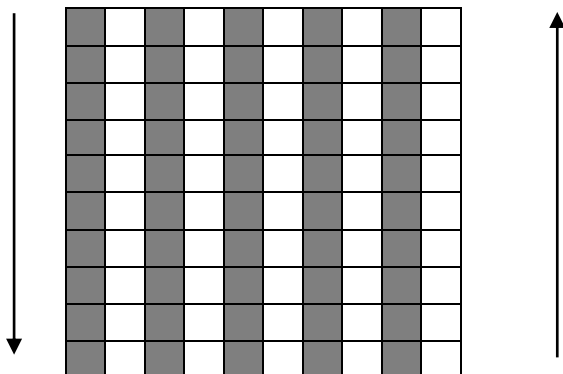


Рис. 6.

Література

1. N.G. de Bruijn, Filling Boxes with Bricks, The American Mathematical Monthly, vol.76, 1969, pp. 37-40.
2. S. Wagon, The American Mathematical Monthly, vol. 94, 1987, pp. 601-617.
3. Глухов О.Д. Одна задача про паркет на торі. - Матеріали XXII Міжнародного науково-практичного семінару «Комбінаторні конфігурації та їх застосування», Запоріжжя-Кропивницький, 2020.- с.38-40.

ДЕСКРИПТИВНІ ЛОГІКИ ТА ПРИРОДНОМОВНІ СИСТЕМИ ЗНАЬ

Гогерчак Г.І.

Київський національний університет імені Тараса Шевченка

DESCRIPTION LOGICS AND NATURAL LANGUAGE KNOWLEDGE SYSTEMS

Abstract. An overview of description logics theory basics is considered. Abstract covers ways of extending the basic attribute language with constructors. Problem of concept executability is stated. Overview of logics application to natural language texts analysis is presented.

Потреба опису знань за допомогою формальних засобів призвела до появи нового класу логік – дескриптивних [1]. Найбільшого поширення наразі набуло сімейство \mathcal{AL} логік, основу яких складає атрибутивна мова.

Нехай $CN = \{A_1, \dots, A_m\}$ та $RN = \{R_1, \dots, R_n\}$ – скінченні непусті множини імен концептів (атомарних концептів) та імен ролей (атомарних ролей). Тоді синтаксис базової логіки \mathcal{AL} визначається наступним чином.

Означення 1. Множина концептів логіки \mathcal{AL} задається індуктивно:

- символи \bullet (універсальний концепт) та \perp (пустий концепт) є концептами;
- довільне ім'я концепту $A \in CN$ є концептом;
- якщо A – ім'я концепту, то $\neg A$ (доповнення до A) – концепт;
- якщо C та D – концепти, то $C \text{ ó } D$ (перетин) – концепт;
- якщо C – концепт, а R – атомарна роль, то $\exists R. \bullet$ (обмежений квантор існування) та $\forall R. C$ (обмеження на значення) – концепти;
- жодних інших концептів не існує.

Семантику логіки \mathcal{AL} можна задати за допомогою поняття інтерпретації з природними теоретико-множинними інтерпретаціями згаданих вище операцій.

Логіка \mathcal{AL} має доволі слабку виразну потужність, проте її можна розширювати за допомогою додаткових операцій над концептами та ролями. Правила розширення зазвичай називають конструкторами логік сімейства \mathcal{AL} .

Таблиця 1.

Конструктори концептів та ролей мов сімейства \mathcal{AL}

	Конструктор	Інтерпретація
\cup	$C \dot{\cup} D$	$(C \dot{\cup} D)^I = C^I \cup D^I$
\exists	$\exists R.C$	$(\exists R.C)^I = \{a \in \Delta^I \mid \exists b[(a,b) \in R^I \wedge b \in C^I]\}$
\mathcal{N}	$\geq nR, \leq nR$	$(\geq nR)^I = \{a \in \Delta^I \mid \{b \mid (a,b) \in R^I\} \geq n\}$ $(\leq nR)^I = \{a \in \Delta^I \mid \{b \mid (a,b) \in R^I\} \leq n\}$
C	$\neg C$	$(\neg C)^I = \Delta^I \setminus C^I$
\mathcal{Q}	$\geq nR.C, \leq nR.C$	$(\geq nR.C)^I = \{a \in \Delta^I \mid \{b \mid (a,b) \in R^I \wedge b \in C^I\} \geq n\}$ $(\leq nR.C)^I = \{a \in \Delta^I \mid \{b \mid (a,b) \in R^I \wedge b \in C^I\} \leq n\}$
I	R^{-1}	$(R^{-1})^I = \{(b,a) \in \Delta^I \times \Delta^I \mid (a,b) \in R^I\}$

Таким чином можна визначити низку мов залежно від того, які конструктори будуть до них включені.

Термінологія \mathcal{T} бази знань формується з аксіом вигляду $C \equiv D$, $C \dot{\cup} D$, $R \equiv S$ та $R \dot{\cup} S$ зі звичними теоретико-множинними інтерпретаціями використаних операторів.

Окрім опису відношень між концептами та ролями, база знань також повинна містити інформацію про окремі факти, про окремі об'єкти предметної області (індивіди) в термінах концептів та ролей.

Введемо додатково нову множину $IN = \{a_1, \dots, a_m\}$ імен індивідів. Частина бази знань, що стосується знань про окремі

індивіди, називається ABox (assertion box) та складається з двох типів фактів:

- $a : A$ (належність індивіда $a \in IN$ до концепту A);
- aRb (зв'язок двох індивідів $a, b \in IN$ роллю R).

Оскільки інтерпретація I ставить кожному атомарному концепту C у відповідність певну підмножину області інтерпретації $C^I \subseteq \Delta^I$, то такому концепту можна поставити у відповідність одномісний предикат $P_C(x)$ належності індивіда x концепту C . Аналогічним чином можна визначити двомісний предикат $P_R(x, y)$ існування відношення R між індивідами x та y .

Це дає змогу говорити про справедливість наступного твердження.

Теорема 1. Логіки сімейства \mathcal{AL} вкладаються у логіку предикатів першого порядку.

Означення 3. Концепт C називається виконуваним в термінології \mathcal{T} , якщо існує інтерпретація I , що задовольняє аксіомам термінології \mathcal{T} , така, що C^I непуста.

Окрім можливості використання методу резолюцій у термінах логіки предикатів, дескриптивні логіки мають власний алгоритмічний апарат, що представлений зокрема алгоритмом семантичного табло [1].

Наявний стан розв'язання проблем обробки природних мов надає якісні вхідні дані для задачі наповнення з текстів природної мови баз знань, що засновані на засадах теорії дескриптивних логік. Так, сучасні моделі машинного навчання для аналізу залежностей, кореферентностей та частин мови дозволяють здійснити перетворення тексту в більш інформативні алгоритмічні структури, а відтак здійснювати видобування та аналіз знань з довільної мови, за умови якісного розв'язання для неї відповідних лінгвістичних задач [2].

До прикладу, речення

Свійські коти рідше займаються виловом мишей та покладаються більше на власних хазяїв.

можна представити за допомогою наступної бази знань:

$$CN = \{ \text{кіт, вилов, вилов _ мишей, хазяїн, свійський, власний} \}$$

$$RN = \{ \text{займатися, покладатися _ на} \}$$

$$IN = \{ x_{\text{свійський_кіт}}, x_{\text{вилов_мишей}}, x_{\text{власний_хазяїн}} \}$$

$$T = \{ \text{вилов_мишей} \hat{=} \text{вилов} \}$$

$$A = \left\{ \begin{array}{l} x_{\text{свійський_кіт}} \text{ займатися } x_{\text{вилов_мишей}}, \\ x_{\text{свійський_кіт}} \text{ покладатися _ на } x_{\text{власний_хазяїн}} \end{array} \right\}$$

де T - термінологія, A - множина фактів.

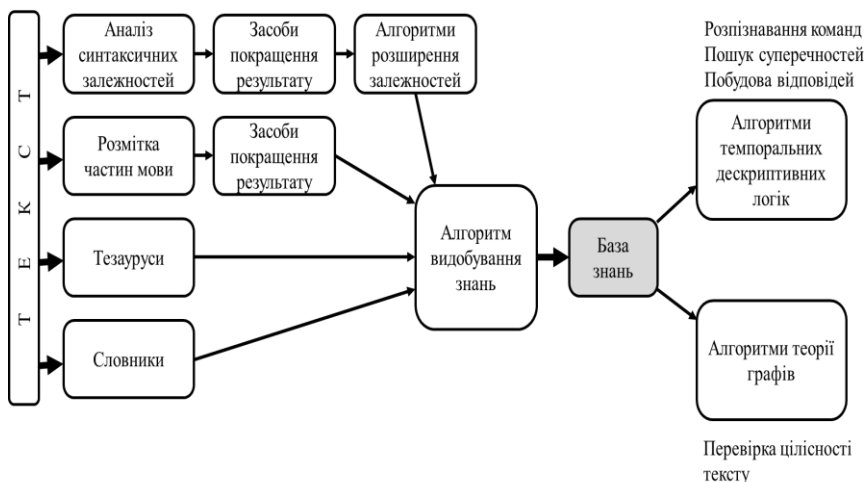


Рис.1. *Схема аналізу текстової інформації на основі баз знань*

Побудована таким чином система знань дозволяє здійснювати аналіз властивостей тексту через аналіз властивостей видобутих з нього знань за допомогою алгоритмів дескриптивних логік, теорії графів тощо.

Зокрема, за допомогою такого представлення можна аналізувати тексти з метою розпізнавання команд, перевірки цілісності, пошуку суперечностей, побудови на основі бази знань діалогової системи для відповідей на запитання користувача, перевірки відповідності твердження поданому тексту тощо.

Література

1. Baader F., Calvanese D., McGuinness D., Nardi D., Patel-Schneider P. The Description Logic Handbook. – Cambridge University Press. – 2007. – P. 578.
2. Gangemi A. Presutti V. Reforgiato Recupero D., Nuzzolese A., Draicchio F., Mongiovi M. Semantic Web machine reading with FRED. – Semantic Web. – 2017. – Volume 8, Issue 6. – P. 873-893.
3. Kryvyi S. L., Darchuk N. P., Hoherchak H. I. Mathematical Logic Problems in Natural Language Processing. – Information Theories and Applications. – Bulgaria, 2020. – Volume 27, Number 2. – C. 121-151.

ПОВЕДІНКОВІ МЕТОДИ ВИЯВЛЕННЯ НЕСАНКЦІОНОВАНИХ ДІЙ ТА АТАК В МЕРЕЖАХ

Горобець В.І., Дубровін В. І., Твердохліб Ю.В.

Національний університет “Запорізька Політехніка”,
FRESHCODE

Abstract. Behavioural methods based on the models of "normal" functioning of an information network. The principle of work is in identifying differences between the current state of the information system functioning and the one that is thought to be exemplary for this network. Any discrepancy is seen as an intrusion or anomaly.

Keywords: wavelet basis, wavelet analysis, wavelet filter, de-noise suppression, network traffic.

Постановка проблеми. Відомі мережеві аномалії настільки різноманітні, що їх не можна категоризувати за допомогою однієї простої класифікації. Швидко зростає кількість атак, їх потужність та складність. Зловмисники шукають принципово нові методи незаконних втручань на мережу і дуже часто існуючі засоби захисту виявляються безсилими перед ними.

Світові лідери з інформаційної безпеки ставлять необхідність виявлення DDoS-атак і протистояння їм як першочергове завдання в своїх дослідженнях і розробках. На даний час не існує якогось універсального засобу для протидії

несанкціонованим вторгненням. Більшість методів мають обмеження в застосуванні до нестаціонарних сигналів.

Методи виявлення мережових атак. Методи виявлення мережових атак, за способом інтерпретації вхідних даних, класифікуються на методи виявлення аномалій і методи виявлення зловживань. До методів виявлення аномалій можна віднести поведінкові методи системи. Вони засновані на використанні інформації про нормальну поведінку системи та її порівнянні з параметрами спостережуваної поведінки [1]. В процесі своєї роботи системи, що використовують даний підхід, порівнюють поточні показники активності з профілем нормальної діяльності, і випадок значних відхилень може розглядатися як свідчення наявності атаки. Дані методи характеризуються наявністю хибнопозитивних спрацьовувань, які пояснюються в першу чергу складністю точного і повного опису безлічі легітимних дій користувачів. Крім того, для більшості подібних систем необхідне проведення етапу попередньої настройки, під час якої система «набирається досвіду» для створення моделі нормального поведінки. Тривалість такого збору даних може займати від кількох тижнів до кількох місяців. До поведінкових методів віднесені наступні методи виявлення атак: вейвлет-аналіз, статистичний аналіз, аналіз ентропії, спектральний аналіз, фрактальний аналіз, кластерний аналіз.

Метод вейвлет-аналізу. Однією з перспективних технологій аналізу даних є вейвлет-перетворення, його інструменти знаходять застосування в самих різних сферах інтелектуальної діяльності. Традиційні методи спектрального аналізу, засновані на перетворенні Фур'є або цифрової фільтрації, надають хороший аналіз стаціонарних і псевдостационарних сигналів. Однак ці методи мають обмеження в застосуванні до нестаціонарних сигналів (тобто сигналів, спектральні характеристики яких змінюються в часі), для функцій з локальними особливостями, зокрема для імпульсних і цифрових сигналів і зображень. Це пов'язано з тим, що базова функція рядів Фур'є - синусоїда визначена на просторі від $-\infty$ до $+\infty$ і за своєю природою є гладкою і строго періодичною функцією.

Така функція на практиці (в умовах обмеження числа членів ряду або спектра розкладання) принципово не здатна описувати довільні сигнали і функції [2].

Вейвлет-аналіз дозволяє виділяти одночасно як частотну, так і часову компоненти мінливості, дає можливість аналізувати часову мінливість частотного спектра процесу. Вейвлет-перетворення володіє самоналагоджувальним рухомим частотно-часовим вікном, однаково добре виявляє як низькочастотні, так і високочастотні характеристики сигналу на різних часових масштабах. Вейвлет-фільтри дозволяють не тільки боротися з шумами, але і витягувати необхідні компоненти сигналу. Оскільки вейвлети мають гарну частотно-часову адаптацію, вони можуть служити зручним інструментом для дослідження частотних характеристик нестационарного сигналу.

Однією з проблем є вибір материнського вейвлета (вейвлет-базису) при проведенні вейвлет-аналізу, який є окремим завданням, але цей вибір нічим не обґрунтований, має випадковий характер і спирається тільки на досвід і думку самого дослідника [2].

Виконання вейвлет-перетворення дозволяє виділити найбільш вагому інформацію як сигнал, відповідних коливань з високою амплітудою, і ігнорувати менш корисну інформацію в коливаннях з низькою амплітудою як шумову складову.

Виявлення аномалій за допомогою вейвлет-аналізу. На практиці в якості вихідних даних використовували агреговані за п'ятихвилинні інтервали середні значення наступних величин: кількість байт за секунду, кількість пакетів за секунду, кількість потоків за секунду, величина середнього розміру TCP-пакета [3].

Наявність різких амплітуд в кожному з представлених сигналів відповідало певним групам аномалій:

- аномалії, викликані помилками в налаштуваннях мережевого обладнання, а також збоєм в роботі обладнання (G1);
- мережеві атаки, представлені класом «відмова в обслуговуванні» (G2);

- перевантаження в мережі (flash crowd), які виникають внаслідок різких сплесків, наприклад, в моменти збільшення легітимних запитів на скачування нових релізів програмного забезпечення (G3);
- інші аномалії, до числа яких відносяться обмін великою кількістю даних, помилки під час запису трафіку на сенсорі або відправки даних колектора NetFlow, що дає можливість аналізу мережевого трафіку на рівні сеансів (G4).

Були виділені три складових компонента первинного сигналу [3]. Низькочастотний компонент сигналу захоплював тривалі за часом мережеві аномалії, які продовжувались від декількох днів. Середньочастотна частина мала нульове математичне сподівання і була призначена для аналізу коливань в межах одного дня. Високочастотна частина відповідала невеликим короткостроковим змінам, які можуть розглядатися як шум.

Після розбиття вихідного сигналу [3] було застосовано до перших двох його компонентів процедуру обчислення локальної дисперсії в рамках ковзаючого вікна розміром 3 години. Далі застосовується метод порогового аналізу. Аномалія ідентифікується в разі, якщо пікова точка останнього сигналу перевищила заданий поріг.

В результаті дослідження було зроблено висновок про те, що представлені типи аномальних подій можуть бути ідентифіковані на конкретних, притаманних їм частотах. Так, грубозернисті аномалії класів G1, G2 і G4 розпізнаються на високих і середніх частотах, в той час як аномалій класу G3 відповідають низькочастотні і середньочастотні сигнали [4].

Недоліками вейвлет-аналізу є неоднозначність вибору базисних функцій, велика обчислювальна складність при розрахунку коефіцієнтів розкладання сигналу. Нетривіальним є завдання правильного завдання розміру вікна. Як було відзначено, якщо розмір ковзаючого вікна набагато перевищує тривалість аномалії [4], то відповідний їй частотний сплеск може бути згладжений, і тим самим атака буде пропущена. В іншому випадку, якщо величина вікна занадто мала, то неминучий потік безглузвих аномалій.

Література

1. Debar H., Dacier M., Wespi A. Towards a taxonomy of intrusion-detection systems //Computer Networks. 1999. vol. 31. Issue 8. pp. 805-822.
2. Твердохліб Ю. В. Методи та інформаційна технологія комплексного оцінювання параметрів вейвлет-перетворення нестационарних сигналів : автореф. дис. ... канд. тех. наук: 05.13.06. Харків. нац. екон. ун-т ім. Семена Кузнеця. Харків, 2018. 20 с.
3. Barford P., Kline J., Plonka D., Ron A. A signal analysis of network traffic anomalies // Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement. 2002. pp. 71–82.
4. Браницкий А.А., Котенко И.В. Анализ и классификация методов обнаружения сетевых атак, СПб, Тр. СПИИРАН, 2016, выпуск 45.-С. 207–244.

СУЧАСНІ АЛГОРИТМИ ВИЯВЛЕННЯ АНОМАЛІЙ ТРАФІКУ МЕТОДАМИ ВЕЙВЛЕТ-АНАЛІЗУ

Гринченко П.В.
САОМ, НУЗІП

Abstract. The work compares two main algorithms for detecting traffic anomalies, based on the processing of squares of wavelet coefficients. The probabilities of errors of the first and second kinds, as well as the size of the analysis window were taken as parameters that will characterize the accuracy of the considered detection algorithms. Sliding windows were used to adapt the algorithms to online detection. The analysis, carried out in this work, allows to establish that the algorithm that based on the sum of the squares of the wavelet coefficients, which were obtained by the decomposition of the original traffic, is more efficient than the algorithm without summation.

В роботі був запропонований огляд сучасних методів вейвлет-аналізу для моделювання мережевого трафіку, що забезпечує підвищення достовірності прийнятих рішень системами захисту в умовах параметричної невизначеності існуючих інформаційних процесів.

Сучасні статистичні методи виявлення порушень у мережі загалом засновані на зіставленні статистичних локальних характеристик потоку пакетів, усереднених за відносно невеликий часовий проміжок, і їх наступному порівнянні із відповідними глобальними характеристиками, що зібрані на тривалому у часі інтервалі.

Одними з перспективних методів виявлення аномалій мережевого трафіку є методи вейвлет-аналізу, що здійснюються на основі вейвлет-декомпозиції аналізованого, в загальному випадку, нестационарного сигналу.

Вейвлет-декомпозиція дозволяє уявити аналізований мережевий трафік у вигляді набору вейвлет-коефіцієнтів, які представляють собою статистичну вибірку, що має свої власні характеристики. У роботі [1] пропонується використання зміни дисперсії як ознаки наявності аномальних викидів. В [2] представлений алгоритм, заснований на дискретному вейвлет-перетворенні та Байєсовому аналізі, що дозволяє виявити і визначити місце розташування будь-яких змін в різних частотних смугах даного часового ряду, однак масштаб розкладання при цьому є обмеженим. В [3] надано WADeS механізм протидії DDoS атакам, що базується на вейвлет-перетвореннях трафіку і наступній оцінці різниці вейвлет-коефіцієнтів на різних часових інтервалах з метою визначення та оцінки моменту атаки. В [4] наведено метод виявлення DDoS атак, що використовує енергетичний розподіл, заснований на вейвлет-аналізі.

Аналіз літературних джерел дозволяє зробити висновок про те, що методи виявлення аномалій, засновані на вейвлет-перетворенні, є ефективним інструментом протидії мережевим атакам. Виходячи з розглянутих методів для вирішення задачі виявлення аномалій трафіку, в роботі проводиться порівняльний аналіз двох основних алгоритмів виявлення аномалій трафіку, що базуються на обробці квадратів вейвлет-коефіцієнтів:

- алгоритм на основі суми квадратів вейвлет-коефіцієнтів;
- алгоритм на основі максимуму квадратів вейвлет-коефіцієнтів.

У якості параметрів, що характеризуватимуть точність розглядаємих алгоритмів виявлення, були взяті ймовірності помилок першого і другого родів (ймовірність пропуску і помилкового виявлення аномалій відповідно), а також розмір вікна аналізу.

Для адаптації алгоритмів до онлайн виявлення була використана техніка ковзаючих вікон, окрема увага приділяється розміру вікна аналізу, що разом із рівнем значимості критерію (α) при перевірці гіпотези, виступають одними із важливих критеріїв порівняння, що мають суттєвий вплив на результати виявлення аномалій різної тривалості.

Проведений в роботі аналіз дозволяє встановити, що алгоритм, заснований на сумі квадратів вейвлет-коефіцієнтів, отриманих в результаті декомпозиції вихідного трафіку, має більшу ефективність у порівнянні з алгоритмом без підсумовування. Найбільший ефект в задачі виявлення аномалій трафіку спостерігається при використанні коефіцієнтів апроксимації для вейвлетів Хаара. При цьому збільшення вікна аналізу призводить до зростання ймовірності правильного виявлення, проте зростання даного показника прямо пропорційне збільшенню ймовірності помилкового виявлення. Порівняльний аналіз характеристик достовірності виявлення аномальних викидів демонструє, що зі збільшенням номера рівня розкладання зростає ймовірність правильного визначення аномалій.

Таким чином, можна зробити висновок про те, що при високому значенні коефіцієнта рівня статистичної значимості ($\alpha = 0,01$) тип вейвлета становить істотний вплив на результат роботи алгоритму. В загальному випадку використання вейвлета Хаара дозволяє отримати найбільш істотний ефект на верхніх рівнях розкладання, в той час як при $\alpha = 0,05$ тип використовуваного вейвлета практично не впливає на загальний результат.

Література

1. Barria, A. Anomaly Detection in Communication Networks Using Wavelets / A. Barria // IEEE Proceedings - Communications. – 2001. – Vol. 148, № 6, P. 26–37.
2. Kemaο, Q. A simple phase unwrapping approach based on filtering by windowed Fourier transform: A note on the threshold selection / Q. A. Kemaο // Opt. Laser Technol. – 2008. – № 8. P. 1091–1098.
3. Atayero, A. A. Detection of DoS and DDoS Attacks in Information Communication Networks with Discrete Wavelet Analysis / A. A. Atayero // International Journal of CSIS. – 2012. – Vol. 10, № 1. P. 38–46.
4. Kim, S. Detecting Traffic Anomalies through Aggregate Analysis of Packet Header Data / S. Kim, A. Reddy // International Conference on Research in Networking : Third International Networking Conference, Athens, 9–14 May 2014 : proceedings. – Berlin: Springer-Verlag, 2014. – P. 879.

НЕЧІТКА КОМБІНАТОРНА МОДЕЛЬ ОДНІЄЇ ЗАДАЧІ ВИБОРУ НА ОСНОВІ НЕЧІТКИХ ВІДНОШЕНЬ

Гуляницький Л.Ф., Рясна І.І.

Інститут кібернетики імені В. М. Глушкова НАН України

FUZZY COMBINATORIAL MODEL OF A CHOICE PROBLEM BASED ON FUZZY RELATIONS

Abstract. A fuzzy model has been developed for the operation of a labor market employment center using fuzzy combinatorial configurations and taking into account the types of S. Stevens' classification scales that can be used to determine the characteristics of labor market actors. Fuzzy relations are built on fuzzy sets under presence of quantitative and qualitative components that determine the prospects of the unemployed – to be aimed at one of the activities of state employment centers when planning their work.

Розглянемо одну з основних проблем, яка виникає у процесі роботи державних центрів зайнятості, а саме – прийняття рішення щодо направлення безробітних, зареєстрованих у

центрі, на заходи з навчання або підвищення кваліфікації. Нехай $U = \{u_1, \dots, u_n\}$ – множина безробітних, які офіційно зареєстровані в державному центрі зайнятості; $V = \{v_1, \dots, v_p\}$ – множина освітніх заходів, які проводяться державним центром зайнятості; n, p – відповідно, їхня кількість; $\Phi = \{\phi_1, \dots, \phi_q\}$ – множина характеристик безробітних, які відповідають вимогам, що ставляться до них при направленні на ці заходи, q – кількість характеристик. Вважаємо, що кожен елемент множин U та V має хоча б одну з характеристик множини Φ . Відповідність характеристик елементів із U елементам множини характеристик Φ визначимо за допомогою нечіткого двомісного відношення $R: U \times \Phi \rightarrow [0, 1]$, функція належності якого f_R задаватиме "кількісні оцінки ступеня володіння характеристиками".

Задамо на множині $V \times \Phi$ нечітке двомісне відношення $S: V \times \Phi \rightarrow [0, 1]$, яке визначає ступінь важливості характеристик множини Φ для елементів із множини V . Значення функції належності f_S задають кількісні оцінки відношення S – "ступінь "важливості" характеристик безробітних щодо заходів". Визначимо $T: U \times V \rightarrow [0, 1]$ – нечітке двомісне відношення, яке характеризує відповідність елементів множини U елементам множини V , тобто враховує ступінь відповідності елементів цих двох множин за характеристиками, які і визначаються множиною Φ . Нечітке відношення T задаємо як композицію нечітких відношень R і S^{-1} , S^{-1} – обернене до S нечітке відношення, $T = R \circ S^{-1}$. Вид композиції визначаємо з урахуванням типу шкали вимірювань значень функцій належності. Нечітке двомісне відношення T задаємо матрицею $\hat{T} = \left(f_T(u_i, v_j) \right)_{i=1, j=1}^{n, p}$, де $f_T: U \times V \rightarrow [0, 1]$ – функція належності нечіткого відношення T . Функція належності f_T визначає нечітке відношення T –

"перспективність" безробітних". В результаті отримаємо нечітку множину

$$\tilde{A}_T = \left\{ \left((u_i, v_j), f_T(u_i, v_j) \right) \right\}, (u_i, v_j) \in U \times V. \quad (1)$$

Функцію належності нечіткого відношення "перспективність", яка визначає "відповідність" безробітного $u_i \in U$ заходу $v_j \in V$ у випадках вимірювань значень функцій f_R та f_S за шкалами відношень (або за абсолютними шкалами)

$$\text{обчислюємо так: } f_T(u_i, v_j) = \frac{\sum_{m=1}^q \min(f_R(u_i, \varphi_m), f_{S^{-1}}(\varphi_m, v_j))}{\sum_{m=1}^q \max(f_R(u_i, \varphi_m), f_{S^{-1}}(\varphi_m, v_j))}.$$

У випадку, коли вимірювання проводяться за шкалами порядку або інтервалів, $\forall \varphi_m \in \Phi$ замість значень функцій $f_R(u_i, \varphi_m)$, $i \in \{1, \dots, n\}$, та $f_{S^{-1}}(\varphi_m, v_j)$, $j \in \{1, \dots, p\}$, застосовуємо нормовані значення рангів з використанням дробових рангів. Тоді значення $f_T(u_i, v_j)$ будуть визначатись за абсолютною шкалою. У випадку, коли значення функції f_S мають сенс обмежень, тобто $\forall \varphi_m \in \Phi$ $f_S(v_j, \varphi_m) = \lambda_j = \text{const}$, а при вимірюваннях цих значень була використана абсолютна шкала, то $\forall \varphi_m \in \Phi$ значення функції належності $f_T(u_i, v_j)$ обчислюємо за формулою

$$f_T(u_i, v_j) = \max_m \left[\min(f_R(u_i, \varphi_m), f_{S^{-1}}(\varphi_m, v_j)) \right], (u_i, v_j) \in U \times V.$$

Тобто, $\min(f_R(u_i, \varphi_m), f_{S^{-1}}(\varphi_m, v_j)) = \lambda_j$, а саме, у випадку, коли $\forall u_i \in U$ $f_R(u_i, \varphi_m) \geq f_S(v_j, \varphi_m)$, то $f_T(u_i, v_j) = f_S(v_j, \varphi_m) = \lambda_j$. Якщо існує $u_i \in U$, для якого $f_R(u_i, \varphi_m) \geq f_S(v_j, \varphi_m)$, а також існують $u_k \in U$, для яких

$f_R(u_k, \Phi_m) < f_S(v_j, \Phi_m)$, то і в цьому випадку $f_T(u_i, v_j) = \lambda_j$; якщо $f_R(u_i, \Phi_m) < f_S(v_j, \Phi_m)$, то $f_T(u_i, v_j) < \lambda_j$.

Позначимо $M_{U \times V}$ – комбінаторний простір комбінацій, який є множиною усіх підмножин множини $U \times V$, нехай X – елемент простору $M_{U \times V}$,
 $X = \left\{ (u_i, v_j)_k \mid k \in \{1, \dots, |X|\}, u_i \in U, v_j \in V \right\}$, $|X|$ – потужність множини X .

Означення 1. Нечіткою комбінацією \tilde{X} назвемо множину упорядкованих пар

$$\tilde{X} = \left\{ \left((u_1, v_1), \theta_{\tilde{X}}(u_1, v_1) \right), \dots, \left((u_n, v_p), \theta_{\tilde{X}}(u_n, v_p) \right) \right\}, \quad (2)$$

$$\begin{aligned} \theta_{\tilde{X}}(u_i, v_j) &= \min \left\{ f_T(u_i, v_j), \chi_X(u_i, v_j) \right\}, \\ \chi_X(u_i, v_j) &= \begin{cases} 1, & (u_i, v_j) \in X, \\ 0, & (u_i, v_j) \notin X, \end{cases} \end{aligned} \quad (3)$$

де $X \in M_{U \times V}$, $(u_i, v_j) \in U \times V$.

Отже, за означенням X – носій нечіткої множини \tilde{X} . Згідно з (2), (3) кожна нечітка комбінація $\tilde{X} \subseteq \tilde{A}_T$, а нечітка множина \tilde{A}_T визначається за формулою (1). Таким чином, кожному елементу X комбінаторного простору комбінацій $M_{U \times V}$ відповідає нечітка комбінація \tilde{X} , кожна комбінація \tilde{X} , є нечіткою комбінаторною комбінацією (об'єктом) першого порядку першого типу [1] – [2] $\mathbb{K}_{(1)}^1 = \langle Y, \Phi, \underline{X}_{(1)}, \Omega \rangle$, де $\Phi: Y \rightarrow \underline{X}_{(1)}$, $\underline{X}_{(1)}$ – базова множина, $Y = ((1, 1), \dots, (1, p), \dots, (n, 1), \dots, (n, p))$, обмеження Ω визначають Φ як строго монотонне відображення.

Означення 2. Комбінаторним простором нечітких комбінацій $\tilde{M}_{(U \times V)}$ назвемо множину всіх нечітких комбінацій \tilde{X} .

Задачу оптимального вибору \tilde{X}^* сформулюємо як нечітку задачу комбінаторної оптимізації:

знайти елемент $\tilde{X}^* \in D \subseteq \tilde{M}_{(U \times V)}$, такий, що

$$F(\tilde{X}^*) = \max \{ F(\tilde{X}) \mid \tilde{X} \in D \subseteq \tilde{M}_{(U \times V)} \},$$

де F – заданий критерій якості, D – допустима область, що визначається обмеженнями задачі.

Висновки. Запропонована нечітка модель управління роботою державного центру зайнятості з використанням поняття нечітких комбінаторних конфігурацій дозволяє урахувувати різні типи шкал за класифікацією С. Стівенса. Застосування теорії нечітких множин забезпечує більш аргументоване вирішення проблем, принциповою особливістю яких є те, що початкові дані можуть бути подані вербально або в термінах формальної чи професійної мови як спосіб опису низки параметрів та характеристик суб'єктів ринку праці. Напрямок подальшої роботи є обґрунтування адекватності побудованої моделі з урахуванням основних положень репрезентативної теорії вимірювань.

Література

1. Гуляницький Л.Ф., Рясна І.І. До формалізації задач комбінаторної оптимізації на нечітких множинах. *Теорія оптимальних рішень*. Київ: Ін-т кібернетики імені В.М. Глушкова НАН України. 2016. С. 17 – 25.

2. Huliannytskyi L. F., Riasna I. I. Formalization and classification of combinatorial optimization problems. *Optimization Methods and Applications* (eds. Butenko S., Pardalos P. M., Shylo V.). Cham: Springer International Publishing AG, 2017. P. 239–250.

ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЕТАЛОННОЇ МОДЕЛІ СУБ'ЄКТА НАВЧАННЯ

Джума Л. М., Серета О.А.

Льотна академія НАУ

THE STUDENT REFERENCE MODEL SIMULATION

Abstract. In this paper we represented a possibility of the estimation of adequacy and efficiency of the student reference model in designing intellectual training systems for operators of the navigation service systems and air traffic control by simulation modeling. Also, foundation of tool for simulation is represented.

Стрімкий розвиток інформаційних технологій суттєво впливає на усі сфери діяльності людини, включаючи освіту. Для авіаційної освіти це є особливо актуальним, оскільки важливу роль в її складі відіграє тренажерна підготовка, специфікою якої є неможливість задовільної реалізації самостійної підготовки. Варіанти впровадження самостійної роботи зводяться до створення різних автоматизованих навчальних систем. Так, в основі інтелектуальної навчальної системи «Диспетчер Tower», що розроблюється на кафедрі інформаційних технологій Льотної академії НАУ, закладено множину моделей, що характеризують предметну область. Отже, виникає необхідність у виборі засобу для оцінки ефективності та адекватності моделей, одним з яких є імітаційне (об'єктно-орієнтоване) моделювання [1].

Інтелектуальна навчальна система «Диспетчер Tower» розробляється відповідно до навчальної методології, що має назву «діяльнісний підхід». Цей підхід являє собою одну з найбільш прогресивних багатообіцяючих концепцій в освіті із застосуванням автоматизованих навчальних систем, за якої засвоєння знань відбувається за результатами діяльності суб'єкта навчання, що формує їхній зміст та якісні характеристики. Особливе значення при цьому надається моделі суб'єкта навчання, як одному з найважливіших елементів системи. Взагалі, можна розглядати *чотири види моделей* суб'єкта навчання: стартову, нормативну (еталонну, цільову),

поточну та модель помилок. *Нормативна (еталонна) модель* є найважливішою з них, оскільки вона задає мету навчання, та з нею пов'язані стандарти освіти. Вона являє собою знання про те, яким ми хочемо бачити суб'єкт навчання, вимоги до його кінцевого стану [2]. Реалізація еталонної моделі суб'єкта навчання інтелектуальної навчальної системи «Диспетчер Tower» є можливою завдяки виявленню закономірностей у діяльності диспетчера АДВ при виконанні злітно-посадкових операцій на підставі аналізу часу виконання ним технологічних операцій (ТО), а також у циркуляції інформаційних потоків на відповідному робочому місці [1; 3].

Окрема увага при використанні діяльнісного підходу приділяється проблемі оцінки адекватності моделей. У загальному розумінні, під **адекватністю моделі** розуміють відповідність результатів, отриманих на моделі, поведінці реального об'єкта. Під **ефективністю моделі** розуміють практичну корисність (оптимальне поєднання якостей адекватності та простоти) [4; 5].

У зв'язку з тим, що професійна діяльність операторів систем навігаційного обслуговування й управління рухом пов'язана з досить складним механізмом формування та обміну інформаційними потоками в системі обслуговування повітряного руху, яка представляє собою відкриту, складну, безперервно-дискретну, стохастичну, динамічну систему управління [6], в якості методу оцінки адекватності та ефективності моделі було обрано **метод імітаційного (об'єктно-орієнтованого) моделювання**. На відміну від аналітичного моделювання, імітаційне моделювання знімає більшість обмежень, пов'язаних з можливістю відображення в моделях реального процесу функціонування досліджуваної системи, динамічної взаємної обумовленості поточних і подальших подій, комплексного взаємозв'язку між параметрами й показниками ефективності системи [1; 6].

Робоче місце оператора систем навігаційного обслуговування й управління рухом має набір властивостей, який дозволяє здійснювати імітаційне моделювання за допомогою таких інструментів:

1. Теорії масового обслуговування.
2. Мережі Петрі.
3. Програмні засоби об'єктно-орієнтованого моделювання.

Особливістю застосування теорії масового обслуговування є можливість опису лише безперервної складової предметної області, а теорії мереж Петрі - лише дискретної складової. Таким чином, для відображення безперервно-дискретної (гібридної) поведінки доцільно використовувати програмні засоби об'єктно-орієнтованого моделювання [6].

Основним засобом імітаційного моделювання, що використовується для оцінки адекватності еталонної моделі суб'єкта навчання при розробці інтелектуальної навчальної системи «Диспетчер Tower», є програма **AnyDynamics**, також відома під назвою **Rand Model Designer**.

AnyDynamics - це система імітаційного моделювання, що дозволяє створювати моделі складних динамічних систем і проводити обчислювальні експерименти з ними. Вибір саме цієї програми для оцінки адекватності та ефективності еталонної моделі суб'єкта навчання ІНС «Диспетчер Tower» був обумовлений такими особливостями системи:

- об'єктно-орієнтована мова моделювання;
- інтуїтивне інтегроване середовище;
- контроль над коректністю моделі на всіх етапах її створення;
- автоматична побудова виконуваної моделі за створюваним в інтегрованому середовищі описом;
- підтримка інтерактивного й автоматизованого експерименту з виконуваною моделлю;
- можливість покрокового налагодження моделі;
- широкий спектр засобів проведення та візуалізації інтерактивного і автоматизованого експерименту;
- 2D- і 3D-анімація;
- можливість моделювання безперервних, дискретних і гібридних (безперервно-дискретних) систем;
- стандартні бібліотеки класів;
- можливість використання виконуваної моделі в якості незалежної програми.

Для запису команд використовуються мови програмування: C# та Ada. Для зберігання внутрішньої інформації використовуються змінні різних видів, найпоширеніші з них: типу *Контакт* і *Поток* (*double*) для змінних величин та типу *Boolean* для позначення істинності/хибності тверджень [7].

На рис. 1 зображено побудовану нами карту поведінки диспетчера TWR при видачі дозволу на зліт при змінних значеннях напрямку вітру та видимості на ЗПС. Напрямок вітру та видимість на ЗПС тут виражається змінними типу *double* (тобто за кожним «прогоном» системи видається випадкове значення вітру від 1 до 360 та видимості – від 1 до 10000). Також тут наявна одна змінна типу *boolean* (*bad_visibility* – погана видимість), за умовою карти поведінки вона активізується при видимості 2000 та менше.

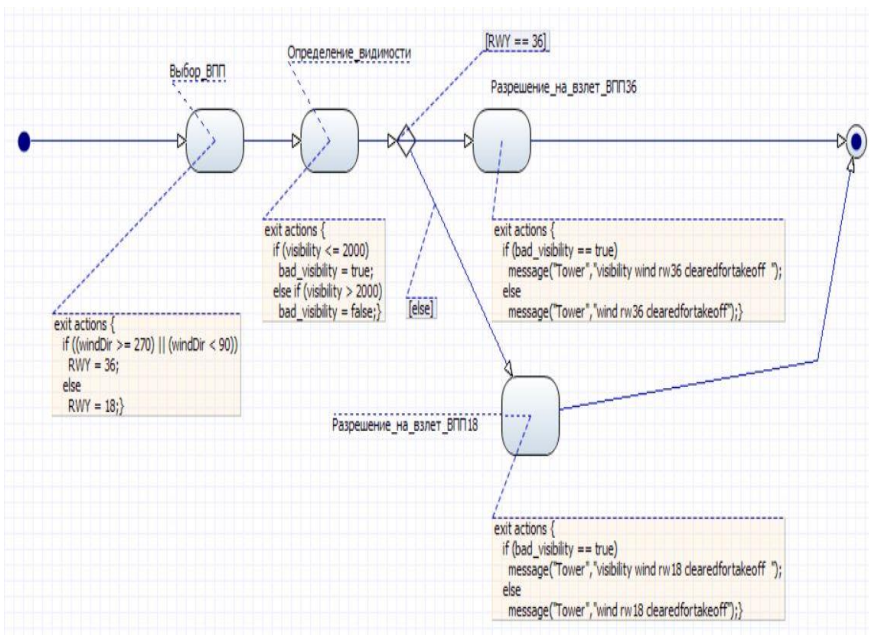


Рис 1. Карта поведінки диспетчера TWR – видача дозволу на зліт при змінних значеннях напрямку вітру та видимості на ЗПС

Таким чином, імітаційне моделювання за допомогою програмних засобів об'єктно-орієнтованого моделювання є найбільш раціональним методом перевірки адекватності та ефективності еталонної моделі суб'єкта навчання інтелектуальної навчальної системи «Диспетчер Tower». На даний момент, актуальним завданням для розробників системи є реалізація еталонної моделі суб'єкта навчання за допомогою комп'ютерних програм з уніфікованою мовою моделювання, а також проведення оцінки її адекватності та ефективності за допомогою системи імітаційного моделювання AnyDynamics.

Література

1. Лавриненко А.С. Модель обучаемого как базовый компонент интеллектуальной обучающей системы. *Політ. Сучасні проблеми науки*: матеріали XV міжнар. наук.-практ. конф., м. Київ, 8-9 квітня 2015 р. Київ, 2015. С. 66.
2. Клюкин В. Э. Web-ориентированные интеллектуальные обучающие системы на основе нечеткого деятельностного подхода в обучении. *Наука и образование: Электронное научное издание*. 2012. № 11. С. 445-476. URL: <http://technomag.edu.ru/doc/489620.html> (дата обращения: 10.03.2021).
3. Лавриненко А. Имитационное моделирование как метод определения эффективности эталонной модели субъекта обучения. *Авіація та космонавтика: стан, досягнення і перспективи*: матеріали XXXVII всеукр. наук.-практ. конф., м. Кропивницький, 12 квітня 2017 р. Кропивницький, 2017. С 230-231.
4. Кубланов М.С. Проверка адекватности математических моделей. *Научный вестник МГТУ ГА*. 2015. № 211. С. 29-36.
5. Адекватность и эффективность математических моделей. URL: <https://ru.calameo.com/read/002696504d9de7849146f> (дата звернення: 24.04.2021).
6. Піліпюнок О.М. Метод підвищення якості управління повітряними суднами операторами систем навігаційного обслуговування й управління рухом: дис ... канд. техн. наук:

05.22.13/ Кіровоградська льотна академія Національного авіаційного університету. Кропивницький, 2017. 265 с.

7. Руководство пользователя AnyDynamics. URL: <https://www.mvstudium.com/downloads/AD8manual-ru.pdf>.
(дата звернення: 24.04.2021).

КОМБІНАТОРНІ СТРУКТУРИ В НАСТІЛЬНИХ ТА АЗАРТНИХ ІГРАХ

Доценко С.І.

*Київський національний університет
імені Тараса Шевченка*

COMBINATORIAL STRUCTURES IS BOARD GAMES AND GAMBLING

Abstract. A short history of combinatorial game theory is considered. Then, some examples of quantitative indicators calculation for several board games are delivered. These examples may be used as good cases in point in combinatorics learning.

Комбінаторна теорія ігор (к.т.і.) виникла як розділ математики, що вивчає послідовні ігри з ідеальною інформацією, але з плином часу границі вивчення к.т.і. постійно розширюється.

Першою статтею з к.т.і. вважається стаття Цермело "Про застосування теорії множин в теорії гри в шахи" (1912), в якій, зокрема, було доведено таке твердження: існує в точності один з трьох алгоритмів: алгоритм виграшу для білих (як би не грали чорні), алгоритм виграшу для чорних або алгоритм, що дозволяє кожному з гравців добитися нічий. Нажаль, по теперішній час не з'ясовано, який саме з трьох алгоритмів існує.

Надалі предметом досліджень стали, зокрема, такі дитячі ігри, як "нім" та "хакембуш", а результати досліджень виявились нетривіальними застосуваннями комбінаторних методів та стали складовою частиною к.т.і.

В 1948 р. Джон Неш придумав гру "Хекс" (шестикутники) та в процесі дослідження цієї гри винайшов концепцію "вкраденої стратегії".

Підрахунки кількості варіантів або кількості позицій, які можуть скластися в настільних іграх можуть бути ілюстративними прикладами у викладенні дисципліни "дискретна математика".

Так, у грі в нарди, на фінальній стадії гри, кожен з гравців заносить 15 своїх шашок у частину дошки, що містить 6 комірок та носить назву "дім". Виявляється, що обчислення кількості варіантів розміщення всіх 15 шашок у домі можна здійснити за допомогою формули кількості сполук з повтореннями: $\bar{C}_6^{15} = C_{6+15-1}^{15} = C_{20}^{15} = C_{20}^5 = 15504$.

Щоб обчислити кількість позицій, враховуючи ті, в яких деякі шашки вже було винесено з дошки, треба додати ще одну комірку, тоді кількість варіантів становитиме: $\bar{C}_7^{15} = C_{7+15-1}^{15} = C_{21}^{15} = C_{21}^6 = 54264$.

Обчислення кількості варіантів розподілу фішок або карт між гравцями на початку гри є вдалою ілюстрацією застосування поліноміальної формули:

$$p(k_1, \dots, k_m) = \frac{(k_1 + \dots + k_m)!}{k_1! \dots \cdot k_m!}.$$

Так, кількість варіантів роздачі карт у грі "преферанс" (гра ведеться колодою у 32 карти, 3 гравці отримують по 10 карт)

$$\text{складає } n = \frac{32!}{(10!)^3 \cdot 2!} \approx 2.75 \cdot 10^{15},$$

у грі "доміно" (гра ведеться набором з 28 фішок (або кісточок), гравці отримують по 7 фішок) у випадку двох гравців

$$\text{складає } \frac{28!}{(7!)^2 \cdot 14!} \approx 1.37 \cdot 10^{11}, \text{ а у випадку трьох або чотирьох}$$

$$\text{гравців ця величина складатиме } \frac{28!}{(7!)^4} \approx 4.73 \cdot 10^{14}.$$

Але найбільша кількість варіантів роздачі має місце у грі в бридж. У грі бере участь 4 гравці, гра ведеться колодою у 52 карти і кількість варіантів роздачі таким чином складає

$$n = \frac{52!}{(13!)^4} \approx 5.36 \cdot 10^{28}.$$

Серед гравців в бридж розповсюдженою є байка про "13 пік". Дехто розповідає, начебто йому в результаті роздачі прийшло 13 карт однієї масті (частіше пікових, бо це найсприятливіший розклад з усіх можливих). Насправді кількість варіантів таких роздач складає

$$k = 4 \cdot 4 \cdot \frac{39!}{(13!)^3} \approx 1.35 \cdot 10^{18},$$
 і таким чином частка розкладів, в

яких дехто отримав всі карти деякої масті складає

$$\frac{k}{n} \approx 2.52 \cdot 10^{-11}.$$
 Отже, комбінаторні підрахунки спростовують

цю байку. Тобто, щоб така подія трапилася, треба, щоб усе доросле населення Землі припинило свої справи та увесь час грало в бридж.

Азартна гра, в яку грали Пушкін та його герой Герман, персонаж «Пікової дами», Лермонтов і герої його творів «Маскарад» і «Штос» називалася «Штос», і виявляється, вона зовсім не цікава і не інтелектуальна, і єдине, що можна зробити цікавого, це обчислити шанси гравців на виграш за допомогою комбінаторних коефіцієнтів.

Гра ведеться колодою з 52 карт (13 номіналів, від двійки до туза, 4 масті). Масть карт у даній грі не має ніякого значення. Організатор гри називається банкометом, він веде гру проти кожного з гравців. Ставки робляться на карти певного номіналу (наприклад, на двійку), масть не має значення.

Банкомет розкладає колоду – першу карту – праворуч, другу – ліворуч, третю – праворуч, четверту – ліворуч і т.д. Якщо карта того номіналу, на яку поставив гравець, з'явиться в лівій (тобто парній) стопці раніше, ніж у правій (непарній), то гравець виграв, і гравець отримує від банкомета виграш, рівний ставці, а якщо у правій (тобто, непарній) стопці раніше, ніж у лівій (парній), то гравець втрачає ставку. На перший погляд гравець і

банкомет знаходяться в однаковому становищі і мають рівні шанси, але насправді це не так, і показати це можна наступним чином.

Будемо в колоді з 52 карт розрізняти карти певного номіналу (наприклад, двійки) і всі інші карти. Чотири двійки можуть розміститися в 52 позиціях $C_{52}^4 = 270725$ способами.

Обчислимо тепер кількість способів, при яких виграє банкомет. Щоб банкомет виграв, потрібно, щоб одна з двійок розташовувалася в позиції i , де i – непарне число, а решта три двійки – в позиціях з номерами, більшими, ніж i , таких позицій буде, очевидно 52- i . При цьому номер першої двійки, що вийшла, змінюється в межах від 1 до 49. Обчислимо кількість способів, при яких одна двійка займає позицію i , а решта три – позиції з номерами, більшими, ніж i . Три карти можуть розташуватися в 52- i позиціях C_{52-i}^3 способами. Отже, кількість способів розкладки карт, при яких виграє банкомет, дорівнює сумі $\sum_{i=1}^{49} C_{52-i}^3$, де i повинно бути непарним.

Позначимо $i = 2k-1$. Якщо i – непарне число, яке змінюється в межах від 1 до 49, то $2i-1$ змінюється в межах від 1 до 25, 52- $i = 53-2k$ і шукана сума набуває вигляду $\sum_{k=1}^{25} C_{53-2k}^3$. Ця сума виявляється рівною 140725, отже, шукана ймовірність виграшу банкомета дорівнює $\frac{140725}{280725} \approx 0.52$, отже, шанси на виграш гравця рівні додатковій імовірності, тобто 0.48.

Література

1. K.G.Binmore. Fun and games: a text on game theory. D.C. Heath, 1992, 602 pp.
2. K.G.Binmore. Playing for real: a text on game theory. Oxford university press, 2007, 639 pp.
3. С. Доценко . Во что играли Пушкин и его герои и при чем здесь комбинаторика. Страна знаний, № 4, 2021 г. www.krainaz.org/2021-04/742-combinatorics.

ВИБІР МІСЦЯ РОЗМІЩЕННЯ КІНОТЕАТРУ НА МАПІ МІСТА

Євдокимов І.В., Ус С.А.

*Національний технічний університет
«Дніпровська політехніка»*

CHOOSING THE CINEMA'S LOCATION ON A CITY MAP

Abstract. The purpose of this work is studying the problem of choosing a cinema's possible location on city map, considering there are several selected potential points and incl. a number of established criteria that directly affect the choosing a locality's area.

Такий бізнес–проект, як відкриття кінотеатру (надалі КТ), є досить вигідним, але у той же час його реалізація передбачає вирішення багатьох проблем – з включенням широкого спектру задач – від визначення місця розташування закладу, його розмірів та оснащення обладнанням для проведення сеансів до складання графіку показів та системи лояльності.

Мета даної роботи – дослідження проблеми вибору можливого місця розміщення КТ на мапі міста з урахуванням наявності декількох обраних потенційних точок та низки встановлених критеріїв і обмежень, які безпосередньо впливають на обрання району населеного пункту.

Змістовна постановка задачі. Нехай у деякому місті інвестором/керівником проекту було обрано декілька точок на мапі для можливого розміщення КТ. Кожна з них характеризується певними показниками щодо можливості відкриття закладу і його потенційної ефективності, які необхідно враховувати при виборі: *ціна оренди/купівлі будівлі/землі, загальна площа будівлі/землі, кількість потенційних клієнтів у обраному районі, середній рівень заробітної плати* навколо обраної точки, можливість для подальшого ймовірного *розширення* зони кінотеатру.

Фрагмент даних для розв'язування задачі представлено у таблиці 1.

Таблиця 1.

Вихідні дані для розв'язування задачі (фрагмент)

точка розміщення	ціна	площа	Клієнти	зарплата	розширення
1	2.5 млн. ум. од.	2500 кв. м.	20 тис. чол.	20 тис. грн.	—
...
9	5 млн. ум. од.	2500 кв. м.	20 тис. чол.	30 тис. грн.	+ 300 кв. м.
10	2.5 млн. ум. од.	1500 кв. м.	5 тис. чол.	50 тис. грн.	+ 150 кв. м.
порогове значення	<i>min</i>	1500 кв. м.	10 тис. чол.	30 тис. грн.	+ 200 кв. м.

Необхідно знайти оптимальну точку для розміщення КТ, враховуючи названі вище характеристики.

Робота проводилася для багатомільйонного міста, поділеного на райони, кожен з яких – зі своїми перевагами і недоліками.

Побудова математичної моделі задачі. Нехай $N = \{1, 2, \dots, 10\}$ – множина можливих місць розміщення КТ.

Введемо такі позначення для i -х точок розміщення:

$c_i, i = \overline{1, 10}$ – ціна,

$s_i, i = \overline{1, 10}$ – площа,

$p_i, i = \overline{1, 10}$ – клієнти,

$b_i, i = \overline{1, 10}$ – зарплата,

$m_i, i = \overline{1, 10}$ – розширення;

$x_i, i = \overline{1, 10} = \begin{cases} 1, & \text{якщо КТ розміщується у } i\text{-й точці,} \\ 0, & \text{якщо КТ не розміщується у } i\text{-й точці.} \end{cases}$

Тоді математичну модель задачі можна записати так:

$$f_1(x_1, \dots, x_{10}) = \sum_{i=1}^{10} c_i x_i \rightarrow \min, \quad (1)$$

$$f_2(x_1, \dots, x_{10}) = \sum_{i=1}^{10} s_i x_i \rightarrow \max, \quad (2)$$

$$\left\{ \begin{array}{l} \sum_{i=1}^{10} p_i x_i \geq 10, \end{array} \right. \quad (3)$$

$$\sum_{i=1}^{10} b_i x_i \geq 30, \quad (4)$$

$$\left\{ \begin{array}{l} \sum_{i=1}^{10} m_i x_i \geq 200, \end{array} \right. \quad (5)$$

$$\sum_{i=1}^{10} x_i = 1, \quad (6)$$

$$x_i \in \{0;1\}, i = \overline{1,10}. \quad (7)$$

Тут: критерій (1) відноситься до загальної вартості оренди/землі, а (2) – до площі потенційних місць розташування КТ; обмеження (3) описує кількість потенційних відвідувачів КТ, яка не може бути нижче за встановлений рівень, (4) – рівень заробітної платні, (5) – існуючу можливість потенційного розширення закладу у майбутньому. Умови (6) і (7) забезпечують знаходження тільки одного місця розміщення.

Розв'язування. Побудована модель є задачею багатокритерійної оптимізації. Для її розв'язування було використано метод головного критерію; згідно з ним, серед усіх представлених обирається тільки один – головний – критерій, тоді як інші перетворюються на обмеження з урахуванням наданих порогових значень [1].

За головний критерій керівник і інвестор проекту обрали вартість оренди/землі (1). Тоді критерій (2) перетворюється на

відповідне йому обмеження; в результаті отримаємо наступне:

$$\sum_{i=1}^{10} s_i x_i \geq 1500, \text{ враховуючи задане порогове значення для}$$

загальної площі КТ, тобто 1500 кв. м. Подальше розв'язування було здійснено для скалярної задачі булевого програмування.

Оптимальним розв'язком отриманої задачі є точка розміщення №9, для якої визначено розмір вкладень інвестицій на рівні 5 млн. ум. од.; її загальна площа становить 2500 кв. м. з можливістю подальшого розширення ще на 300 кв. м.; клієнтський потік передбачено на рівні 20 тис. чол., тоді як середня зарплатня навколо – 30 тис. грн.

Зауважимо, що задача вибору оптимального місця розміщення також може бути вирішена методом аналізу ієрархій [2].

Висновки. Застосування математичного моделювання і методу головного критерію дозволило визначити точку для розміщення кінотеатрального закладу на мапі міста, яка повністю відповідає побажанням керівника/інвестора проекту і дозволяє відкрити відносно недорогий КТ (до 5 млн. ум. од.) з загальною площею не менше 1500 кв. м. у умовному центрі населеного пункту з потенційним потоком клієнтів на рівні 20 тис. чол. (за 1 календарний місяць).

Література

1. Ус С. А. Моделі й методи прийняття рішень: навч. посіб. / С. А. Ус, Л. С. Коряшкіна; М–во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – 2-ге вид. випр. – Дніпро : НТУ «ДП», 2018. – 302 с.

2. Євдокимов І. В. Вибір місця розташування кінотеатру на основі багатьох критеріїв [Електронний ресурс] / І. В. Євдокимов, С. А. Ус // Інформаційні технології: теорія і практика: Матеріали IV Всеукраїнської інтернет-конференції здобувачів вищої освіти і молодих учених. Дніпро, 17 – 19 березня 2021 р. – Дніпро : НТУ «ДП». – 2021. – <http://ir.nmu.org.ua/handle/123456789/157651>.

ПОБУДОВА ТА АНАЛІЗ ГРАФУ ЗВ'ЯЗНОСТІ ДОДАТКІВ У МОБІЛЬНОМУ МАГАЗИНІ GOOGLE PLAY

Ізвалов О.В..

Льотна академія Національного авіаційного університету

BUILDING AND ANALYZING THE CONNECTIVITY GRAPH OF THE APPLICATIONS IN THE GOOGLE PLAY MOBILE STORE

Abstract. One of the main sources of applications visibility in the Google Play mobile store is their presence in the list of “Apps like this” presented on the pages of the other applications in the store. Considering the suchlike presence as a pointed edge of a graph defined on the set of applications, the connectivity graph is built and its properties are analyzed. Corresponding software tool is released as freeware.

Основним джерелом, де користувачі мобільних пристроїв з операційною системою Android знаходять нові додатки та ігри, є мобільний магазин Google Play. Його встановлено на 2 мільярди активних пристроїв у світі. Протягом 2020 року через Google Play було виконано 108 мільярдів завантажень додатків та ігор [1] Всього у магазині представлено 2.9 мільйони додатків, серед яких 400 тисяч - ігри [2].

Користувачі Google Play можуть знаходити цікаві для себе додатки наступними способами:

- ☐ пошуком за ключовими словами;
- ☐ у рейтингах найбільш завантажуваних додатків;
- ☐ у рекомендаціях редакції Google Play на головній сторінці магазину та сторінках категорій;
- ☐ переглядаючи рекламні банери розробників та публішерів додатків, які розміщуються у Google Play;
- ☐ у списку “схожі додатки”, що розташовується під описом кожного додатку у Google Play (рис.1);
- ☐ у списку “інші додатки того ж розробника”, що розташовується під описом кожного додатку.

Для індивідуальних розробників та невеликих команд, знаходження їхніх розробок у категоріях “схожі додатки” є одним з найбільш суттєвих каналів здобуття нових

користувачів. Це підтверджується тим, що коли 21 червня 2018 року Google Play поміняв алгоритми формування списку схожих додатків, обсяги установок для більшості розробників зменшилися у 10-20 разів [3].

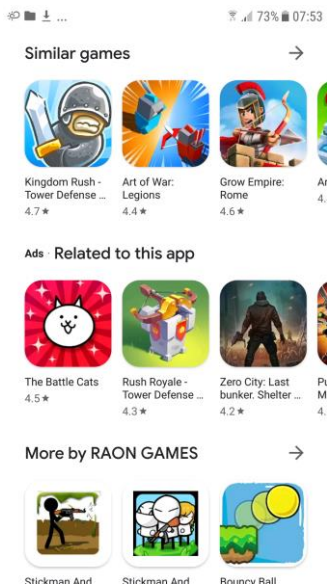


Рис.1. Вигляд списку “схожі додатки”, рекламованого списку та списку інших додатків розробника

Особливість формування списку “схожі додатки” полягає у відсутності симетричності схожості: якщо додаток А вказаний алгоритмом Google Play як “схожий” до додатку В, то не обов’язково додаток В буде вважатися “схожим” до додатку А. Таким чином, можна розглядати орієнтований граф над множиною додатків G , напрямленими ребрами (а, b) якого буде належність додатку b до списку схожих на сторінці додатку a.

Повний розгляд множини G з 2,9 мільйонів наявних додатків у Google Play та побудова усіх ребер схожості є ресурсовимогливою та непрактичною задачею. Натомість, ми розглядаємо підграф такого графу з кожної вершини якого виходять не більше трьох ребер. Це обмеження зумовлено тим, що хоча розділ “схожі додатки” у Google Play може містити

численні додатки, частіше всього користувач взаємодіє із першими трьома позиціями, які видимі без необхідності прокручування сторінки. Підмножина вершин G_1 для побудови графу обиралася наступним чином. Спочатку до неї включили додатки, що вказані у Google Play як “топ категорій” за завантаженістю. Надалі до G_1 рекурсивно включали додатки, що належать до цієї ж категорії та зв’язані з існуючими елементами G_1 вихідними ребрами, побудованими за першими трьома записами у категорії “схожих”.

Запустивши алгоритм побудови графу на категорії “Ігри” та обравши для початкової множини 50 ігор зі списку “Топ безкоштовних”, отримано наступні результати. Множина G_1 наприкінці роботи складається з 7551 гри. Отже, можна вважати, що лише 1,9% з існуючих у Google Play ігор потрапляють до 1-3 позиції списку рекомендацій. Усі вихідні ребра з елементів G_1 приходять у елементи G_1 або вели б до додатків, що не входять у категорію “Ігри”. За кількістю вхідних ребер елементи G_1 розподіляються чином, наведеним у таблиці 1.

Таблиця 1.

$K(N)$ - кількість вершин графу із N вхідними ребрами

N	K(N)	N	K(N)	N	K(N)	N	K(N)
0	27	5	573	10	28	15	0
1	1687	6	349	11	14	16	1
2	1943	7	192	12	5	17	0
3	1559	8	92	13	6	18	3
4	1015	9	55	14	2	19	0

Після обробки граф зв’язності додатків у Google Play можливо візуалізувати (рис.2). Відповідне програмне забезпечення опубліковано за безкоштовною ліцензією [4].

Для розробленого програмного забезпечення можна окреслити два напрями застосування. Інформативне: розробники мобільних додатків, які поширюють їх у Google Play, можуть наочно побачити, які з інших додатків рекомендують їхнє як “схоже”. Аналітичне: при дослідженні ринку для нового мобільного додатку можна отримати уявлення про те, які з існуючих додатків Google Play рекомендує користувачам, та виділити критерії схожості із додатком, що знаходиться у розробці.

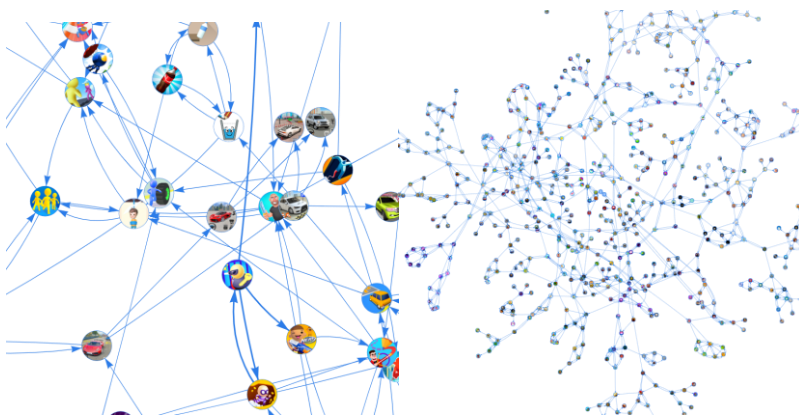


Рис.2. Візуалізація графу зв'язності додатків у Google Play

Серед подальших напрямів дослідження - розгляд графа зв'язності у динаміці, аналіз доданих та вилучених з нього додатків від тижня до тижня. Також планується адаптація розроблених алгоритмів для аналізу AppStore, основного магазину мобільних додатків для операційної системи iOS.

Література

1. Зростання обсягів завантажень додатків з Google Play
<https://www.statista.com/statistics/734332/google-play-app-installs-per-year/>
2. Кількість додатків та ігор у Google Play
<https://www.appbrain.com/stats/android-market-app-categories>
3. Вплив зміни алгоритмів Google Play на видимість додатків

https://www.gamasutra.com/blogs/VladChetrusca/20180626/320734/Thousands_of_indie_android_devs_on_the_brink_of_extinction_after_Play_store_changes_visibility_algorithm_rules.php

4. Розроблене програмне забезпечення для побудови графу
<https://www.airapport.com/p/similar-games-network-builder.html>

ПЕРЕРАХУВАННЯ НЕІЗОМОРФНИХ ДВОКОЛЬРОВИХ ХОРДОВИХ О-ДІАГРАМ РОДУ ТРИ З ОДНИМ СІРИМ (АБО ЧОРНИМ) ЦИКЛОМ

Кадубовський О.А.

ДВНЗ «Донбаський державний педагогічний університет»

ENUMERATION OF 2-COLOR CHORD O-DIAGRAMS OF THE GENUS THREE THAT HAVE ONE GREY (OR BLACK) FACE UNDER ROTATION

Abstract. In this paper we consider 2-color chord O-diagrams (of order n) with one grey and $(n-6)$ black faces under the action of the rotation group (cyclic of the order n). We have established explicit formulas for counting the number of non-isomorphic diagrams from the specified class. In addition, for natural $7 \leq n \leq 30$ we have also listed the exact value of the number of non-isomorphic such diagrams.

Добре відомо, що хордові діаграми ефективно використовують в багатьох галузях науки, зокрема математиці (топології, теорії вузлів), фізиці, біології, генетиці тощо (напр. [2], [5], [6-7], [10]).

Означення 1. Коло з $2n$ точками на ньому (що є вершинами правильного $2n$ -кутника), дуги якого почергово розфарбовані у два кольори (чорний і сірий) та фіксованою нумерацією вершин за годинниковою стрілкою, будемо називати двокольоровим $2n$ -шаблоном – рис. 1 а).

Означення 2. 2-кольоровою хордовою n -діаграмою будемо називати n -діаграму, побудовану на основі двокольорового $2n$ -шаблону; 2-кольорову діаграму, яка не

містить хорд (містить хорди), що сполучають вершини з номерами однакової парності, називатимемо O –діаграмою (N –діаграмою) – рис. 1 *c*), 1 *b*).

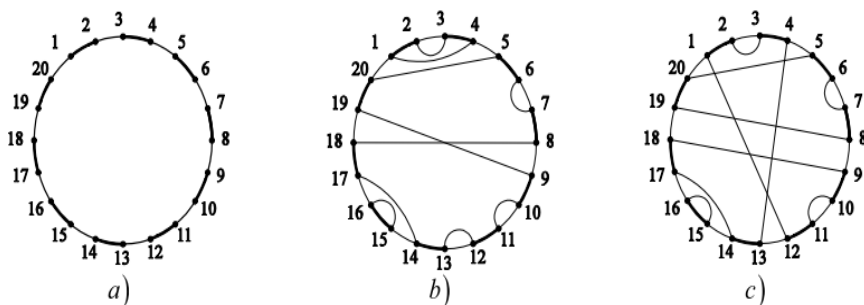


Рис.1. *a*) двокольоровий 20-шаблон; *b*) N –діаграма;
c) O –діаграма.

Означення 3. Чорним (сірим) циклом 2-кольорової O –діаграми з n хордами будемо називати послідовність хорд та чорних (відповідно сірих) дуг, які утворюють гомеоморфний образ (орієнтованого) кола – рис. 1 *b*), *c*).

Означення 4. Родом 2-кольорової O –діаграми з n хордами називають число g , яке визначається рівністю $2g = n + 1 - (k + l)$, де k і l – число чорних та (відповідно) сірих циклів діаграми.

Множину O –діаграм з n хордами, які мають l сірих та k чорних циклів будемо позначати через $\mathfrak{Z}_{k,l}^{n,g}$.

Означення 5. Хордові діаграми називають ізоморфними, якщо одну можна одержати з іншої в результаті повороту.

В роботі [1] вперше встановлено рекурентні формули, за допомогою яких є принципово можливим підрахунок числа діаграм з класу $\mathfrak{Z}_{k,l}^{n,g}$. Крім того, для початкових $g = 0; 1; 2; 3$ в [1] встановлено явні формули, які пізніше були одержані та уточнені в [4], а в [3] – запропоновано іншу рекурентну формулу.

В загальному випадку задача про підрахунок числа нееквівалентних (відносно дії групи діедра порядку $2n$) діаграм з класу $\mathfrak{Z}_{k;l}^{n,g}$ є нерозв'язаною. Задача повністю була розв'язана для класів $\mathfrak{Z}_{k;l}^{n,0}$ та $\mathfrak{Z}_{l;l}^{n,g}$ лише у 2015 р. в [6], [7]. Відомими також є явні формули для підрахунку числа неізоморфних діаграм з класів $\mathfrak{Z}_{n-2;l}^{n,1}$, $\mathfrak{Z}_{n-3;2}^{n,1}$, $\mathfrak{Z}_{n-4;3}^{n,1}$ та $\mathfrak{Z}_{n-4;l}^{n,2}$ [8], [9]. Розв'язанню задачі задачі для класу $\mathfrak{Z}_{n-6;l}^{n,3}$ й присвячено дане повідомлення.

З урахуванням результатів роботи [4], одержано справедливність

Твердження 1. Для довільного натурального $n \geq 7$

$$|\mathfrak{Z}_{n-6;l}^{n,3}| = \frac{1}{72} \cdot C_{n+1}^8 \cdot (9n^4 - 18n^3 - 57n^2 + 34n + 80) = d(n). \quad (1)$$

Використовуючи лему Бернсайда встановлено твердження

Теорема 1 (основна). Число $d^*(n)$ неізоморфних (нееквівалентних відносно дії циклічної групи порядку n) діаграм з класу $\mathfrak{Z}_{n-6;l}^{n,3}$ можна обчислити за формулою

$$d^*(n) = \frac{1}{n} \left(d(n) + \sum_{j|n, j \in \{2;3;4;6;7;8;9;12\}} \phi(j) \rho\left(n; \frac{n}{j}\right) \right), \quad (2)$$

де: $\phi(q)$ – функція Ейлера, $d(n)$ визначається за формулою (1), а величини $\rho\left(n; \frac{n}{j}\right)$ для $j \in \{2;3;4;6;7;8;9;12\}$ визначаються за допомогою наступних співвідношень:

$$\rho\left(n; \frac{n}{12}\right) = \frac{n}{12}; \quad \rho\left(n; \frac{n}{9}\right) = \frac{2n}{9}; \quad \rho\left(n; \frac{n}{8}\right) = \frac{n}{4}; \quad (3)$$

$$\rho\left(n; \frac{n}{7}\right) = \frac{5n}{7}; \quad \rho\left(n; \frac{n}{6}\right) = C_{\frac{n}{6}}^2; \quad \rho\left(n; \frac{n}{4}\right) = C_{\frac{n}{4}}^2 \cdot \frac{n+40}{12}; \quad (4)$$

$$\rho\left(n; \frac{n}{3}\right) = C_{\frac{n}{3}}^3 \cdot \frac{3n+29}{4}; \quad \rho\left(n; \frac{n}{2}\right) = C_{\frac{n}{2}}^4 \cdot \frac{37n^2 + 294n - 2320}{120}; \quad (5)$$

Таблица 1.

Початкові значення величин $d(n)$ та $d^*(n)$

n	$d(n)$	$d^*(n)$	n	$d(n)$	$d^*(n)$
7	180	30	19	1 801 329 010	94 806 790
8	3 044	385	20	3 600 529 450	180 028 084
9	26 060	2 900	21	6 925 187 830	329 770 930
10	152 900	15 308	22	12 869 925 310	584 999 362
11	696 905	63 355	23	23 190 544 696	1 008 284 552
12	2 641 925	220 242	24	40 637 416 600	1 693 230 295
13	8 691 683	668 591	25	69 427 501 000	2 777 100 040
14	25 537 655	1 824 311	26	115 901 728 800	4 457 765 752
15	68 396 900	4 559 818	27	189 426 912 675	7 015 811 753
16	169 537 940	10 596 558	28	303 616 322 295	10 843 450 498
17	393 481 660	23 145 980	29	477 960 911 025	16 481 410 725
18	862 928 092	47 941 370	30	739 984 318 125	24 666 159 267

Література

1. Адрианов Н. М. Аналог формулы Харера-Цагира для одноклеточных двукрашенных карт. *Функциональный анализ и его приложения*. 1997. Том 31, № 3. С. 1–9.

2. Cori R., Marcus M. Counting non-isomorphic chord diagrams. *Theoretical Computer Science*. 1998. Vol. 204. P. 55–73.

3. Chapuy G. A new combinatorial identity for unicellular maps, via a direct bijective approach. *Advances in Applied Mathematics*. 2011. Vol. 47, No. 4. P. 874–893.

4. Goupil A., Schaeffer G. Factoring n -cycles and counting maps of given genus. *European Journal of Combinatorics*. 1998. Vol. 19, No. 7. P. 819–834.

5. Stoimenov A. Enumeration of chord diagrams and an upper bound for Vassiliev invariants. *Journal of Knot and its Ramifications*. 1998. Vol. 7, No. 1. P. 93–114.

6. Кадубовский А. А. О числе топологически неэквивалентных функций с одной вырожденной критической точкой типа седло на двумерной сфере, II. *Труды*

международного геометрического центра. 2015. Том 8, № 1. С. 46–61.

7. Кадубовський О. А. Перерахування топологічно нееквівалентних гладких мінімальних функцій на замкнених поверхнях // *Топологія відображень маловимірних многовидів : Збірник праць Інституту математики НАН України*. 2015. Том 12, № 6. С. 105–145.

8. Кадубовський О. А. Перерахування двокольорових хордових О-діаграм роду 1, які мають три сірих (або чорних) цикли, відносно дії групи діедра. *Збірник наукових праць фізико-математичного факультету ДДПУ*. 2019. Вип. 9. С. 25–41.

9. Кадубовський О.А., Сипчук Є.Ю. Про число неізоморфних двокольорових хордових О-діаграм роду 2, які мають один сірий (або чорний) цикл. *Збірник наукових праць фізико-математичного факультету ДДПУ*. 2020. Вип. 10. С. 21–32.

10. Шарко В.В. Гладкая и топологическая эквивалентность функций на поверхностях. *Український математичний журнал*. 2003. Т. 55, № 5. С. 687–700.

ДЕЯКІ УЗАГАЛЬНЕННЯ ЗАДАЧІ ПАРАЛЕЛЬНОГО УПОРЯДКУВАННЯ

Караваєв К. Д., Турчина В.А.

*Дніпровський національний університет імені Олеся
Гончара*

SOME GENERALIZATIONS OF PARALLEL SEQUENCING PROBLEM

Abstract. The paper considers problems of parallel sequencing, which belong to discrete optimization problems. Generalizations of the classical problem arising from concrete practical applications are presented, namely, problems with sophisticated structure of jobs and executors, with additional conditions and different variants of objective functions. Contemporary challenges related to these problems are formulated: identification of cases solvable in polynomial time, derivation of theoretical estimates of accuracy, the reachability of these estimates,

Першим поштовхом до розвитку задач оптимального упорядкування вершин графів усередині минулого століття стала проблема, пов'язана з оптимізацією порядку виконання робіт на конвеєрах автомобільного виробництва. Ефективний розв'язок формалізованої задачі був знайдений відносно швидко, тому панувала думка, що задача може бути розв'язана у загальному випадку за поліноміальний час.

Впродовж наступних років дослідження математиків були спрямовані на розробку точних методів, заснованих на схемах направленого перебору, та виділенні підкласів задач, для яких існують поліноміальні алгоритми знаходження точних розв'язків. Такі алгоритми знаходять своє застосування в різноманітних сферах: у процесорах обчислювальних пристроїв, приладобудуванні, виробничій та споживчій сферах, тощо.

Розглянемо класичну постановку задачі оптимального упорядкування та її узагальнення, що виникли з необхідності враховувати додаткові обмеження та умови, притаманні конкретним практичним застосуванням.

Нехай маємо n завдань, для яких наявні деякі виробничі зв'язки, що регламентують порядок їх виконання. Математичну модель такої задачі природно подавати у вигляді орієнтовного графу $G(V, U)$, у якому вершини відповідають завданням, а дуги – виробничим зв'язкам. В класичній постановці вважається, що витрати часу на виконання кожного завдання однакові.

Означення 1. Паралельним упорядкуванням вершин орієнтовного графу $G = (V, U)$ називається таке упорядкування його вершин по місцях, розташованих у лінію, при якому з того, що пара $(i, j) \in U$ впливає, що вершина i розташовується в упорядкуванні S лівіше вершини j , тобто з того, що $(i, j) \in U \wedge (i \in S[p], j \in S[q])$ впливає, що $p < q$.

Означення 2. Довжиною l упорядкування S називається число непорожніх місць в ньому: $l(S) = \sum_{i=1}^n \text{sign}|S[i]|$, де $S[i]$ – множини елементів, що знаходяться в упорядкуванні S на місці i .

Означення 3. Шириною h упорядкування S називається величина, що дорівнює найбільшій кількості елементів, що розташовані на одному місці: $h(S) = \max_i |S[i]|$, де $i = 1, \dots, n$.

Класична задача. По заданим графу G і значенню ширини h побудувати паралельне упорядкування мінімальної довжини.

У загальному випадку ця задача є NP-важкою. Точні алгоритми поліноміальної складності відомі лише для деяких спеціальних видів графу G та $h = 2$.

Узагальнення класичної задачі охоплюють усі складові задачі: вводяться додаткові умови на виконання робіт, змінюється структура завдань, властивості виконавців або цільова функція.

До узагальнень із ускладненою структурою завдань можна віднести наступні. Необхідність враховувати різний час виконання завдань стала поштовхом до дослідження задач, в яких одна вершина може займати декілька місць. Окрім цього для виконання деяких завдань одного виконавця може бути недостатньо, чи можуть бути потрібні деякі конкретні виконавці, тоді розглядають задачі, в яких одна вершина може займати декілька позицій на одному місці.

До задач із ускладненою структурою виконавців відносять такі. Для випадків, коли в різні моменти часу доступна різна кількість виконавців, вводяться задачі, де кількість вершин, які можна ставити на i -те місце, не більша ніж h_i . Задачі з різним часом обробки завдань додатково поділяються на задачі із перериваннями (можна зупинити виконання завдання та продовжити його потім) та без, які є імітацією витискальної багатозадачності. Також розглядаються часові затримки між виконанням завдань, що можуть бути пов'язані з необхідністю передачі результатів між виконавцями. Виділяють задачі, де затримки є сталими і виникають лише за необхідності комунікації, та задачі, в яких затримки залежать від виконуваних завдань.

Задачі з додатковими умовами пов'язують з наявністю директивних термінів виконання та часом початку виконання завдань. Перші виникають, коли є необхідність завершити деякі завдання у заданий термін через обмежену в часі наявність того

чи іншого ресурсу. Такі задачі поділяють на задачі зі строгими (не можна порушувати) та нестрогими (можна порушувати) термінами. Відмітимо також, що задача із строгими директивними термінами може не мати допустимих розв'язків. Потреба розглядати другий тип задач виникає в аналогічних умовах, проте регламентується час початку виконання завдання, а не його кінець. Окремо виокремлюють онлайн варіант задачі, де про завдання та структуру графу G дізнаємося протягом виконання завдань, що відповідає розподіленню обчислень між процесорами у обчислювальній техніці.

На ряду з класичною *цільовою функцією* (довжина упорядкування) розглядаються також сума часів виконання всіх завдань та її зважений варіант. Для задачі з нестрогими директивними строками розглядають сумарне запізнення, його зважений варіант та кількість робіт, виконаних вчасно.

Також досліджуються задачі з комбінацією цих узагальнень та цільових функцій.

Наведені задачі у загальному випадку є також NP-важкими. Більшість з них також є NP-важкими для випадків, коли граф G – є лісом або набором ланцюжків, та навіть у випадках з одним виконавцем. Поліноміальні алгоритми для таких задач відомі лише для специфічних умов.

Теоретичні дослідження цих задач, окрім дослідження задач на складність, включають розробку алгоритмів для методів гілок та меж та пошук точних поліноміальних алгоритмів для задач із специфічними умовами для подальшого застосування їх в якості евристики для наближеного розв'язання у загальному випадку. Для оцінки точності алгоритмів найчастіше використовують границю відношення довжини знайденого за алгоритмом упорядкування до довжини оптимального упорядкування. Знаходження найкращого досяжного значення такого відношення пов'язане не з конкретним алгоритмом, а із складністю задачі визначення, чи є довжина оптимального упорядкування меншою, аніж деяке фіксоване число d . Так, якщо задача визначення, чи існує упорядкування довжини менше d , є NP-важкою, тоді не існує поліноміального алгоритму з похибкою меншою за $(d + 1)/d$, якщо $P \neq NP$.

Окремо можна відмітити напрямок, що займається адаптацією алгоритмів локального пошуку для задач теорії розкладів. Через необхідність враховувати технологічні обмеження та додаткові умови, організація процесу побудови околу допустимих розв'язків є складною, тому такі алгоритми, як і наближені поліноміальні алгоритми, мають вузьку область ефективного застосування.

Отже, важливою проблемою теорії розкладів є відсутність універсальних підходів. Так більшість поліноміальних алгоритмів, окрім того, що є наближеними у загальному випадку, ще й в принципі не можуть знайти точний розв'язок через наявні властивості, незалежно від початкових умов. Спробу подолати цю проблему для класичної задачі зроблено у роботі [1], де було показано, що цю задачу можна звести до задачі з упорядкуванням особливого вигляду, для якої існує поліноміальний алгоритм, що завжди може знайти оптимальний розв'язок при правильно обраному початковому наближенні.

Література

1. Turchyna V., Karavaiev K. Analysis of algorithms for constructing dense sequencing of digraphs vertices. – Proceedings of The Third International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020), Zaporizhzhia, Ukraine, April 27-May 1, 2020. CEUR Workshop Proceedings 2608, CEUR-WS.org 2020 – Pp. 690-703.

АНАЛІЗ СТРУКТУРИ ГРАФІВ В ЗАДАЧАХ ПАРАЛЕЛЬНОГО УПОРЯДКУВАННЯ З ПЕРЕРИВАННЯМИ

Коваленко Є. О., Турчина В. А
Дніпровський національний університет
імені Олеся Гончара

GRAPH STRUCTURE ANALYSIS IN PROBLEMS OF PARALLEL ORDERING WITH INTERRUPTIONS

Abstract. The parallel ordering problems, which belong to the scheduling theory are considered as the optimization problems on graphs. The initial conditions imply that a finite set of jobs is needed to be done with a limited amount of resources (workers, machines, processors, etc.). In addition, there are consistent technical restrictions on the jobs sequencing. Also, each worker can perform each job with equal efficiency, but each job may take a different time to be fulfilled. The main questions were about the possibility of interruptions in jobs, can they, and how exactly can they affect the optimal solution? For this reason, the structure of graphs was analyzed, particularly in the case when there are no technological restrictions, which means the graph contains only isolated vertices.

Серед задач теорії розкладів цікавими як з теоретичної, так і з практичної точок зору, є задачі, в яких допускаються переривання при виконанні робіт. За рахунок дозволу переривань інколи доводиться суттєво покращити значення цільової функції. Розглянемо одну із задач, яка пов'язана з оптимальним виконанням скінченної множини робіт, на порядок проведення яких накладаються несуперечливі технологічні обмеження, скінченною множиною ресурсів. Ця задача, яка в [1] визначена як задача паралельного упорядкування вершин орграфів, має наступне формулювання.

Нехай задана скінченна множина робіт $A = \{1, 2, \dots, n\}$. Задано граф $G(V, U)$, $|V| = n$, який моделює технологічні обмеження.

Означення 1. Паралельним упорядкуванням S вершин орграфів є таке розміщення елементів множини V на n місцях, розташованих в лінію (допускаються порожні місця), коли кожен елемент розміщується лише на одному місці. Крім того,

якщо із вершини i йде дуга в вершину j , тобто $(i, j) \in U$, то вершина i знаходиться в S лівіше за вершину j .

Означення 2. Довжиною упорядкування $l(S)$ будемо називати кількість непорожніх місць в S .

Означення 3. Шириною упорядкування $h(S)$ будемо називати величину $\max_{1 \leq i \leq l} |S[i]|$.

Задача 1. За заданим орієнтованим графом G та шириною паралельного упорядкування $h(S)$ треба знайти паралельне упорядкування мінімальної довжини, тобто $l(S) \rightarrow \min_S$.

В класичній постановці [2] вважається, що час виконання кожного завдання дорівнює 1, тому заданий граф — незважений. Розглянемо узагальнення цієї задачі на випадок, коли це обмеження не виконується. Для кожної роботи задано час її виконання t_i , $T = \{t_1, t_2, \dots, t_n\}$, $i = \overline{1, n}$. Вважаємо, що кожен виконавець може виконувати будь-яку роботу. Виникає питання, чи може зменшитися значення цільової функції, якщо дозволити переривання при виконанні завдань, і в разі позитивної відповіді, дослідити як на це впливає структура самого графа? При цьому виграш від переривання будемо оцінювати наступним чином:

$$W = \left(1 - \frac{l_{II}^*}{l^*}\right) \cdot 100\%,$$

де l^* — оптимальна довжина упорядкування без переривань, l_{II}^* — оптимальна довжина упорядкування з перериваннями.

Оцінимо виграш для графів G_1 , G_2 та G_3 .

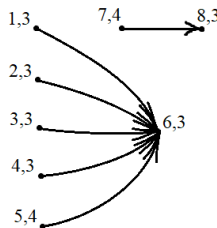


Рис.1. Граф G_1

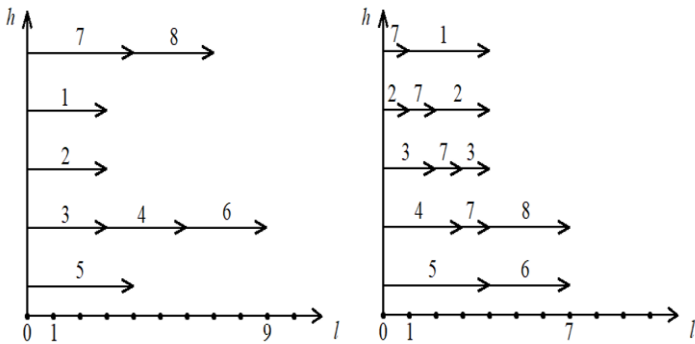


Рис.2. Упорядкування графа G_1 без переривань та з перериваннями при $h = 5$

Для графа G_1 упорядкування з перериваннями дає виграш $W \approx 22\%$. Легко бачити, що оптимальне упорядкування без переривань неефективне з точки зору завантаженості виконавців.

Розглянемо випадок, коли $U = \emptyset$.

1,4 2,3 3,3 4,3 5,3
 • • • • •

Рис.3. Граф G_2

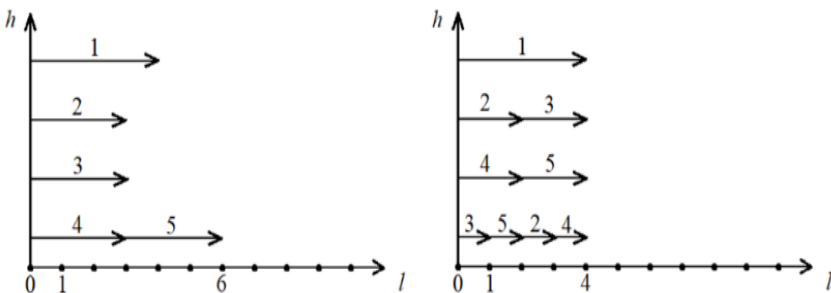


Рис.4. Упорядкування графа G_2 без переривань та з перериваннями при $h = 4$

Видно, що оптимальний розв'язок із використанням переривань дає виграш $W = 33,3\%$ відносно оптимального розв'язку без переривань. Розглянемо приклад для довільного числа $p > 1, p \in N$.

$$\begin{matrix} 1,p & 2,p & 3,p & \dots & p+1,p \\ \bullet & \bullet & \bullet & & \bullet \end{matrix}$$

Рис.5. Граф G_3

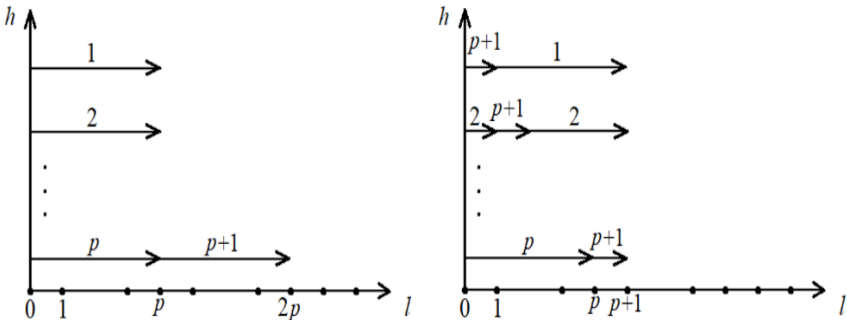


Рис.6. Упорядкування графа G_3 без переривань та з перериваннями при $h = p$

Оптимальна довжина паралельного упорядкування без переривань складає $l^* = 2p$, а довжина з перериваннями $l_{\Pi}^* = p + 1$. Очевидно, що при більших значеннях p відношення довжин упорядкування буде зменшуватися, таким чином при $p \rightarrow \infty$

$$\frac{l_{\Pi}^*}{l^*} = \frac{p+1}{2p} \rightarrow 0,5,$$

з чого випливає, що виграш

$$W = \left(1 - \frac{p+1}{2p}\right) \cdot 100\% \rightarrow 50\%.$$

Як видно з прикладу, у випадку коли $U = \emptyset$ дозвіл переривань може суттєво покращити значення цільової функції. Подальшого вивчення потребує дослідження впливу структури графів, що задають технологічні обмеження, на оптимальність розв'язку. Зокрема паралельно-послідовних графів, графів, що

задают обчислення нерозгалужених арифметичних виразів, змішаних дерев та інших.

Література

1. Бурдюк В.Я. Алгоритмы параллельного упорядочения: учебное пособие / В.Я. Бурдюк, В.А. Турчина. — Днепропетровск: ДГУ, 1985. — 84 с.
2. Гордон В.С. Задачи теории расписаний с заданным частичным порядком / В.С. Гордон, А.Б. Долгий. — Автомат. и телемех., 2010, выпуск 10, с. 15–25

ФРАГМЕНТАРНАЯ МОДЕЛЬ ЗАДАЧИ TREEDEPTH ДЕКОМПОЗИЦИИ ГРАФА

Козин И. В., Землянский А.А., Евсеев О.В.
Запорожский национальный университет

FRAGMENTARY MODEL TREEDEPTH DECOMPOSITION OF GRAPH

Abstract. The paper considers a problem of minimum height Treedepth decomposition of connected graph. Proposes a fragmentary model for this task. Based on a fragmented model constructed metaheuristics for finding suboptimal solutions to the problem of Treedepth decomposition of the graph.

Определение 1. Treedepth декомпозицией (глубинным деревом) связного графа $G = (V, E)$, с множеством вершин V и множеством ребер E называется дерево $T = (V, E_T)$ с тем же множеством вершин, построение которого происходит по нижеизложенной рекурсивной процедуре. Если G имеет одну вершину, тогда $T = G$. В противном случае, выбирается вершина $v \in V$, как корень дерева T . Производится treedepth-декомпозиция каждой компонента связности $G \setminus \{v\}$. в каждом из полученных деревьев выбирается вершина, которая соединяется ребром с вершиной $v[1]$.

Эти выбранные вершины будем называемся потомками вершины v . Соответственно, вершину v будем называть

родительской вершиной для выбранных вершин из нижележащих деревьев. Таким образом, в treedepth декомпозиции $T = (V, E_T)$ связного графа $G = (V, E)$ каждое ребро связывает пару вершин, имеющих отношение родитель-потомок в дереве T .

Каждой вершине $v \in V$ припишем число $L(v)$ -уровень вершины по следующему правилу. Вершина графа v_0 , которая выбирается первой имеет уровень $L(v_0)=0$. Уровень каждой вершины-потомка на единицу больше уровня ее родительской вершины. Максимальный из уровней вершин графа называется высотой построенной Treedepth декомпозиции.

Задача treedepth декомпозиции графа заключается в построении глубинного дерева минимальной высоты.

В общем случае задача treedepth декомпозиции графа является NP -трудной [2]. Поэтому для поиска субоптимальных решений оправдано применение метаэвристик.

Покажем, что задача treedepth декомпозиции графа может рассматриваться как оптимизационная задача на фрагментарной структуре и, следовательно, к этой задаче применимы методы оптимизации для задач на фрагментарной структуре описанные в [3].

В соответствии с [3] фрагментарной структурой на конечном множестве X называется совокупность его подмножеств $E = \{E_1, E_2, \dots, E_n\}, E_i \subseteq X$, такую, что

1. $\emptyset \in E$,
2. $\forall E_i \in E, E_i \neq \emptyset, \exists e \in E_i, E_i \setminus \{e\} \in E$.

Элементы множества E будем называть *допустимыми фрагментами*. Одноэлементные допустимые фрагменты называются *элементарными фрагментами*. Допустимый фрагмент будем называть *максимальным*, если он не является подмножеством никакого допустимого фрагмента.

Построение treedepth декомпозиции T связного графа $G = (V, E)$ на основе фрагментарной модели начинается с определения элементарных фрагментов. В качестве элементарных фрагментов выберем одноэлементные множества, содержащие вершины графа. Опишем алгоритм построения treedepth декомпозиции:

1. Задается порядок(нумерация) элементарных фрагментов (вершин) графа v_0, v_1, \dots, v_n .
2. Выбирается первая вершина v_0 из последовательности. Она становится корнем дерева-treedepth декомпозиции T .
3. Удаляются из графа G все ребра инцидентные вершине v_0 .
4. На очередном шаге алгоритма в каждой компоненте связности оставшейся части графа выбирается одна, первая по заданному порядку вершина. Эти вершины являются потомками вершины v_0 . В дерево T добавляются вершины-потомки и ребра, соединяющие вершину v_0 с ее потомками.
5. Каждая вершина-потомок рассматривается как родительская вершина для соответствующей компоненты связности. Шаги 2-4 алгоритма повторяются для каждой такой вершины.
6. Процесс продолжается пока в T не войдут все вершины.

Приведём пример treedepth декомпозиции графа (рис. 1). Слева изображён исходный граф $G = (V, E)$, справа – его treedepth декомпозиция $T = (V, E_T)$, построенная на основе последовательности $2 - 5 - 4 - 6 - 1 - 0 - 3$

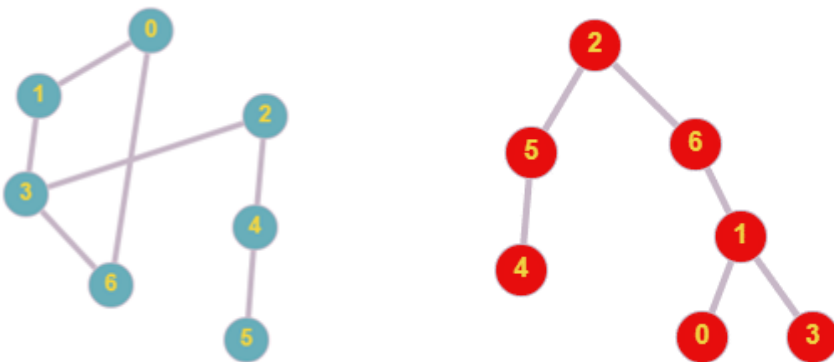


Рис.1 Связный граф и его Treedepth-декомпозиция.

Существует *NP*-полная задача построений *Treedepth* минимальной высоты. Высотой *treedepth* является самый длинный путь от корня дерева к вершине. Для минимизации высоты *Treedepth* и поиска оптимального, или приближенного к оптимальному решению фрагментарный алгоритм, описанный выше, можно совместить с любой метоэвристикой на фрагментарной структуре [3].

В настоящей работе использовался эволюционный алгоритм. Параметры эволюционного алгоритма определялись опытным путём для каждой конкретной задачи и зависят от размеров входных данных, мощности вычислительного устройства и доступного для проведения вычислений времени.

Эффективность нижеописанного эволюционного алгоритма на фрагментарных структурах в задаче минимизации высоты *Treedepth* была доказана при решении задач на международном конкурсе компьютерных экспериментов и параметризованных вычислений РАСЕ-2020 [4].

Для проведения вычислительного эксперимента был применён эволюционный алгоритм с начальной популяцией 16-1024 особи. Расчёты выполнялись на вычислительном устройстве с 16 параллельными потоками. Каждый поток независимый, с собственной уникальной начальной популяцией и ветвью эволюции. Каждую итерацию эволюции в потоке производится сортировка последовательностей (особей) от лучшей к худшей. В получившемся списке проводится попарное скрещивание особей, первая особь скрещивается со второй, третья с четвёртой и т.д. Лучшая полученная особь сохраняется в неизменном виде и переходит в следующую итерацию, если в результате скрещивания не было получено лучшее дерево. Остальные особи вымирают после скрещивания. Таким образом, количество деревьев в каждой итерации уменьшается в два раза.

Мутация представлена инверсией части перестановки случайной длины. Вероятность мутации изменяется от лучших особей к худшим. Для лучших особей вероятность мутации меньше, для худших вероятность больше.

Скрещивание проводится до тех пор, пока в каждом потоке не останется 2 - 4 особи. Скрещивание последних 2 - 4 особей между собой не проводится. В конце происходит сравнение оставшихся особей и выбор лучшей.

Литература

1. [Bodlaender, Hans L.](#); Deogun, Jitender S.; Jansen, Klaus; Kloks, Ton; Kratsch, Dieter; Müller, Haiko; Tuza, Zolt (1998), "[Rankings of graphs](#)" (PDF), [SIAM Journal on Discrete Mathematics](#), 11 (1): 168–181 doi:[10.1137/S0895480195282550](#)
2. Pothén, Alex, The complexity of optimal elimination trees, Tech. Report CS-88-13, [Pennsylvania State University](#)
3. I. V. Kozin, N. K. Maksyshko, V. A. Perepelitsa Fragmentary Structures in Discrete Optimization Problems, [Cybernetics and Systems Analysis](#) November 2017, Volume 53, [Issue 6](#), P 931–936. <https://doi.org/10.1007/s10559-017-9995-6>
4. PACE-challenge. URL: <http://pacechallenge.org/2020> /td/

ФРАГМЕНТАРНЫЙ ПОДХОД В ТРУДНОФОРМАЛИЗУЕМЫХ ЗАДАЧАХ ОПТИМИЗАЦИИ СО МНОГИМИ КРИТЕРИЯМИ

Козин И. В., Перепелица В.А., Нарзуллаев У.Х.

Запорожский национальный университет

*Самаркандский филиал Ташкентского университета
информационных технологий*

FRAGMENT APPROACH TO COMPLEXLY FORMALIZED OPTIMIZATION PROBLEMS FOR MANY CRITERIA

Abstract. A new method is proposed for finding suboptimal solutions to difficult-to-formalize problems with many criteria. The basis of this method is the construction of a fragmentary model of the problem, which allows you to search for feasible solutions using «greedy» algorithms.

Существует ряд практических оптимизационных задач, для которых трудно или практически невозможно формализовать

критерии оптимальности. К числу таких задач можно отнести многие практические задачи размещения, задачи составления расписаний, задачи распределения ресурсов и другие [1]. Одним из основных принципов поддержки принятия решений в таких задачах является взаимодействие «Человек – машина» на всех уровнях решения задачи [2]. Это налагает ряд требований на алгоритмы решения прикладных задач, которые могут использоваться в таких системах.

К числу основных требований относятся:

- а) небольшое число шагов и, соответственно, высокая скорость выполнения алгоритма;
- б) возможность коррекции результата на каждом шаге;
- в) возможность реализаций различных вариантов поиска.

Эффективными алгоритмами такого класса являются «жадные» алгоритмы, которые за малое число шагов приводят к оптимальному решению оптимизационных задач. К сожалению, жадных алгоритмов, дающих точное решение задач дискретной многокритериальной оптимизации, существует немного. Для приближенных алгоритмов с оценками доказано, что их трудоемкость не ниже трудоемкости алгоритмов точных [3]. Таким образом, наиболее приемлемыми алгоритмами диалога «Человек-машина» являются эвристические алгоритмы различных типов.

Одним из перспективных подходов к созданию таких алгоритмов является переход к фрагментарной модели оптимизационной задачи [4].

Определяется множество фрагментов решений и внешняя операция соединения фрагментов. Каждое допустимое решение задачи состоит из конечного числа фрагментов из заданного множества фрагментов, соединенных путем внешней операции.

Задается эффективная процедура упорядочения фрагментов, которая позволяет получать разные порядки в соответствии с выбранными правилами.

Задаются условия присоединения фрагмента. Для уже выбранного подмножества фрагментов задается эффективная процедура проверки возможности присоединения фрагмента, не принадлежащего выбранному подмножеству.

Допустимые решения задачи строятся следующим фрагментарным алгоритмом:

Просматриваются все еще не выбранные фрагменты в заданном порядке. Выбирается первый из них, который удовлетворяет условиям присоединения и добавляется к множеству. Условие остановки алгоритма – не удалось найти очередной фрагмент.

С фрагментарного алгоритма снято основное требование – условие оптимальности. Определение понятия оптимальности переносится на ЛПР (лицо принимающее решение).

Трудоемкость фрагментарного алгоритма оценивается следующим образом. Пусть N – количество фрагментов, m – оценка трудоемкости проверки условий присоединения на каждом шаге, n – верхняя оценка числа фрагментов, составляющих решение задачи. Тогда трудоемкость фрагментарного алгоритма оценивается числом $N \cdot m \cdot n$.

Фрагментарный подход удалось применить для решения некоторых прикладных задач. Одна из них – задача размещения прямоугольных фрагментов в прямоугольнике по критерию симметрии или «красоты» [5]. Задается группа G преобразований прямоугольной решетки. Для каждого множества узлов определяется его степень симметрии как отношение мощности этого множества к мощности его орбиты. Задача состоит в следующем – выбрать в прямоугольной решетке множество подрешеток, каждая из которых изоморфна некоторой типовой решетке, так чтобы степень симметрии множества узлов выбранной системы подрешеток была максимальной.

Другая задача, в которой оказывается эффективным фрагментарный подход – задача составления расписания со многими критериями, среди которых критерий равномерной загрузки оборудования. Постановка этой задачи следующая: требуется произвести обработку N деталей на некотором количестве станков в течение времени T . Каждая деталь требует на обработку время t_i , $i=1,2,\dots,N$. Задача так распределить обработку деталей, чтобы, во-первых, уложиться во время T , во-вторых, использовать по возможности меньшее число станков,

в третьих, загрузить станки наиболее равномерно, то есть, чтобы в любые два момента времени на промежутке $[0, T]$ количества станков, занятых обработкой в эти моменты времени отличались как можно меньше.

Литература

1. А.А.Самарский, А.П.Михайлов Математическое моделирование: Идеи. Методы. Примеры. -М., «Наука.Физматлит», 1997, 320с..
2. М.В.Михалевич, И.В.Сергиенко Моделирование переходной экономики –Киев, Наукова думка, 2005, 670с.
3. Пападимитроу, К.Стайглиц., Комбинаторная оптимизация. Алгоритмы и сложность., -М., «Мир», 1985, 512с.
4. I. V. Kozin, N. K. Maksyshko, V.A.Perepelitsa Fragmentary Structures in Discrete Optimization Problems, Cybernetics and Systems Analysis November 2017, Volume 53, Issue 6, pp 931–936 DOI: <https://doi.org/10.1007/s10559-017-9995-6>
5. И.В.Козин Фрагментарный алгоритм для задачи симметричного размещения, Радиоэлектроника, информатика, управление, 2006 №1, с.76-83.

РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ РІВНЯНЬ В АСОЦІАТИВНИХ КІЛЬЦЯХ З ОДИНИЦЕЮ

Кривий С. Л.

Київський національний університет імені Тараса Шевченка

SOLUTION OF SYSTEM OF LINEAR EQUATIONS IN ASSOCIATIVE RINGS WITH UNIT

Abstract. A method for solution of systems of linear equations over associative non commutative rings with unit is considered.

Розглядається метод розв'язання систем лінійних рівнянь в асоціативних не комутативних кільцях з одиницею. Для асоціативно-комутативних кілець з одиницею алгоритми

розв'язання систем лінійних рівнянь розглядалися в роботах [1,2].

1. Необхідні означення

Асоціативним кільцем з одиницею називається алгебра $K=(A, \{+, \bullet\})$, яка відносно операції додавання $+$ є абелевою (адитивною) групою, відносно операції множення \bullet (мультиплікативним) моноїдом і ці операції зв'язані законами дистрибутивності (знак операції \bullet у виразах не пишеться):

$$(\forall a, b, c \in K) a(c + d) = ac + ad, (c + d)a = ca + da.$$

Нуль 0 адитивної групи називається нулем кільця, а одиниця 1 моноїда – одиницею кільця. Для них справедливі тотожності:

$$(\forall a \in K) \ a0 = 0a = 0, \quad a1 = 1a = a.$$

Якщо для елементів $a, b \in K \setminus \{0\}$ виконується рівність $ab=0$, то a називається лівим дільником нуля, а b – правим дільником нуля. Елемент $x \in K$ називається лівим (правим) дільником одиниці (або оберненим зліва (справа)), якщо в кільці є такий елемент u , що $xu=1$ ($ux=1$). Відомо, що множина дільників одиниці утворюють групу, яка називається мультиплікативною групою кільця [3].

2. Лінійні рівняння над K

Оскільки кільце K не комутативне, то системи лінійних рівнянь (СЛР) можуть бути правими і лівими (невідомі справа (зліва) множаться на коефіцієнти):

[illegible]

– права СЛР, де $a_{ij}, b_j \in K, i=1, \dots, p, j=1, \dots, q$.

Розглянемо одне лінійне рівняння (ЛР):

$$L(x) = a_1x_1 + \dots + a_qx_q = b, \quad (1)$$

де $a_i, b \in K, i=1, \dots, q$. Так як дільники нуля в цьому кільці не складають ідеал, то можлива ситуація, коли сума (різниця) двох або більше дільників одиниці є дільником нуля і навпаки, сума дільників нуля є дільником одиниці. Розглянемо ситуацію коли всі коефіцієнти рівняння (1) дільники одиниці. Тоді, якщо a_i – дільник одиниці, то існує елемент $c_i \in K$ такий, що $a_i c_i = 1$. Звідси знаходимо окремий розв’язок ЛР (1) – $x_i = (0, \dots, 0, c_i b, 0, \dots, 0)$. Розглянемо однорідне ЛР, яке відповідає ЛР (1), тобто

$$L(x) = a_1x_1 + \dots + a_qx_q = 0 \quad (2)$$

Враховуючи умови, розв'язками цього рівняння будуть вектори вигляду:

$x_1^0 = (c_1, -c_2, 0, \dots, 0)$, $x_2^0 = (c_1, 0, -c_3, \dots, 0)$, ..., $x_{q-1}^0 = (c_1, 0, \dots, -c_q)$, де c_i – праві обернені до a_i , тобто $a_i c_i = 1$, $i = 1, 2, \dots, q$.

Зауважимо, що ліве ЛР $L(x) = x_1 a_1 + x_2 a_2 + \dots + x_q a_q = 0$ буде мати ці самі розв'язки. Розглянемо випадок, коли сума декількох коефіцієнтів є дільником нуля. Не обмежуючи загальності, нехай $a_1 + a_3 + a_5 = c$, де c – дільник нуля. Тоді існує $d \in K$ такий, що $cd = 0$ і вектор $x = (d, 0, d, 0, d, 0, \dots, 0)$ – розв'язок ЛР (2). Дійсно, $a_1 d + a_3 d + a_5 d = (a_1 + a_3 + a_5) d = c d = 0$. Покажемо, що цей розв'язок зображується лінійною комбінацією векторів x_i^0 . Побудуємо вектори $s_1 = x_1^0 - x_2^0 = (0, -c_2, c_3, 0, 0, 0, \dots, 0)$, $s_2 = x_1^0 - x_4^0 = (0, -c_2, 0, 0, c_5, 0, \dots, 0)$, тоді нижченаведена лінійна комбінація дає нам шукане зображення вектора x :

$$x_1^0 a_1 d - s_1 a_3 d - s_2 a_5 d = (d, -c_2 a_1 d - c_2 a_3 d - c_2 a_5 d, d, 0, d, 0, \dots, 0) = (d, 0, d, 0, d, 0, \dots, 0) = x,$$

оскільки $-c_2 a_1 d - c_2 a_3 d - c_2 a_5 d = -c_2 (a_1 + a_3 + a_5) d = -c_2 c d = 0$. Таким чином, справедлива

Лема 1. Множина векторів $x_i^0, i = [1, q-1]$ – базис множини розв'язків ЛР (2).

Наслідок. Загальний розв'язок ЛР (1) має вигляд $x = x_1 + \sum x_i^0 d_i$, де x_1 – окремий розв'язок ЛР (1), $d_i \in K$ – довільні, $i = 1, \dots, q-1$.

Якщо всі коефіцієнти a_i дільники одиниці, а b – дільник нуля, то справедлива лема 1 і наслідок з неї, хоча окремий розв'язок тепер буде дільником нуля. Дійсно, довільний розв'язок $x_i = (0, \dots, 0, c_i b, 0, \dots, 0)$ має координату $c_i b$, яка дільник нуля. Отже, загальний розв'язок ЛР (1) має вигляд $x = x_1 + \sum_i x_i^0 d_i$, де x_1 – окремий розв'язок ЛР (1), x_i^0 – базисні розв'язки ЛР (2), $d_i \in K$ – довільні.

Інші можливі випадки в цій роботі не розглядаються.

3. Системи лінійних однорідних рівнянь

Розглянемо систему лінійних однорідних рівнянь (СЛОР) вигляду

$$S_{r_2} = \begin{cases} a_{11}x_1 + \dots + a_{1q}x_q = 0, \\ a_{21}x_1 + \dots + a_{2q}x_q = 0. \end{cases}$$

де коефіцієнти обох рівнянь дільники одиниці. Побудуємо для першого рівняння СЛОР базис множини розв'язків $B_1 = \{e_1, e_2, \dots, e_{q-1}\}$ вищеописаним способом і обчислимо значення $d_i = L_2(e_i)$, $i = 1, 2, \dots, q-1$. Якщо всі d_i дільники одиниці, то розглянемо рівняння вигляду

$$d_1y_1 + d_2y_2 + \dots + d_{q-1}y_{q-1} = 0 \quad (3)$$

і знайдемо його базисні розв'язки g_1, \dots, g_{q-2} . Нехай $g_i = (c_{i1}, c_{i2}, \dots, c_{iq-1})$, побудуємо лінійну комбінацію $u_i = e_{1c_{i1}} + \dots + e_{q-1c_{iq-1}}$.

Лемма 2. Вектори $\{u_i\}$, $i=1,2,\dots,q-2$ є базисом множини розв'язків СЛОП S_{r2} .

Застосовуючи цей спосіб до СЛОР з більшим числом невідомих, де всі її коефіцієнти і проміжні значення рівнянь вигляду (3) дільники одиниці, будуємо таким способом базис множини її розв'язків. Із лем 1 і 2 за допомогою індукції випливає

Теорема. Якщо проблема обчислення обернених елементів в кільці K алгоритмічно розв'язувана, коефіцієнти СЛР і проміжні рівняння задовольняють умові бути дільниками одиниці, то загальний розв'язок таких СЛОР Sr знаходиться описаним вище способом.

4. Системи лінійних неоднорідних рівнянь

Перетворимо неоднорідну СЛР Sr до СЛОР, вводячи додаткову невідому x_0 при вільних членах, які є дільниками одиниці:

[illegible]

Побудуємо базис множини розв'язків цієї СЛОР: $B=\{e_1, e_2, \dots, e_k\}$. В цьому базисі необхідно побудувати розв'язок, у якого остання координата дорівнює 1. Для цього потрібно знайти розв'язок, у якого остання координата є дільником одиниці. Якщо такого розв'язку в множині B немає і ніяка сума дільників нуля не дає дільника одиниці, то початкова неоднорідна СЛР несумісна.

На закінчення відзначимо, що кільця, де *даний метод розв'язання СЛР працює в повному об'ємі*, є кільця з діленням (тіла).

Часова складність методу розв'язання залежить від складності знаходження обернених елементів в кільці і складності реалізації його операцій множення і додавання. Наприклад, відомо, що в кільці матриць над полем раціональних чисел ці елементи обчислюються в поліноміальному часі і тоді побудова загального розв'язку СЛР теж виконується в поліноміальному часі [2].

Література

1. Кривий С. Л. Лінійні діофантові обмеження та їх застосування (видання 2). – К: Інтерсервіс. – 2021. – 260 с.
2. Bockmayr A., Weispfenning V. Solving Numerical Constraints.– Handbook of Automated Reasoning.–2001. – chap. 12. – P. 753-842
3. Кострикин А. И. Введение в алгебру (часть 2). – М.:ФИЗМАТЛИТ. – 2004. - 272 с.

ІНФОРМАЦІЙНА МОДЕЛЬ ФІНАНСОВОЇ БЕЗПЕКИ СУБ'ЄКТІВ ГОСПОДАРЮВАННЯ

Мазютинець Г.В., Маляр М.М., Шаркаді М.М.
Ужгородський національний університет

INFORMATION MODEL OF BUSINESS ENTITIES FINANCIAL SECURITY

Abstract. Financial security of business entities: ensures balanced and stable financial condition; promotes the effective operation of the business entity; allows in the early stages to identify problem areas in the organization; neutralizes crises and prevents bankruptcy. This publication presents an information model for determining the level of financial security, which is the state of enterprises protection from unforeseen negative changes. The author proposes modeling the level of the enterprises financial security using the theory of fuzzy sets. The activity of economic entities in a market economy is determined primarily by the state of their finances. Forecasting the level of financial security is a complex analytical

and calculation process and requires a detailed study of development trends and prediction of the studied factor components impact on the level of the company's economic security. Performed study was devoted to the current problem of developing a model of a multilevel neural network for information technology on the example of determining the level of financial security of the company, which occurs in the functioning of socio-economic systems and is based on the application of fuzzy logic, fuzzy sets and neural phase.

Глобальна фінансова криза 2020 року супроводжується зменшенням ділової активності у багатьох галузях та відповідним падінням ВВП у багатьох країнах. Це свідчить про те, що існуючі теорії фінансового аналізу і методи прогнозування розвитку фінансово-економічних систем практично себе вичерпали. Тому постає потреба істотного переосмислення використовуваних методів математично-статистичного моделювання. Вчені схилиються до використання нейронних мереж, які вважаються одним з найбільш відомих та ефективних інструментів інтелектуального аналізу даних.

Для аналізу стану рівня фінансової безпеки і прогнозування загроз сьогодні будуються моделі із застосуванням апарату нечітких множин [1]. Оскільки фінансова безпека підприємства є досить індивідуальною характеристикою, яка залежить від значної кількості факторів зовнішнього та внутрішнього походження, доцільно використати саме нечіткі множини. Вплив даних факторів і наслідки цього впливу окремо для кожного підприємства можливо оцінити лише з долею суб'єктивної ймовірності, що задана у якості інтервалу значень.

Нехай для певного підприємства відома множина кількісних і якісних показників його функціонування, а також відома історія цих показників за певні періоди часу. Виникає завдання передбачити оцінку рівня фінансової безпеки даного підприємства. Для вирішення даної проблеми пропонується наступна інформаційна модель, яка потребує вирішення низки задач (завдань), у вигляді нейро-фазі мережі (рис. 1), структура якої складається із сукупності послідовних етапів, на кожному із яких розв'язується конкретний клас задач [2].

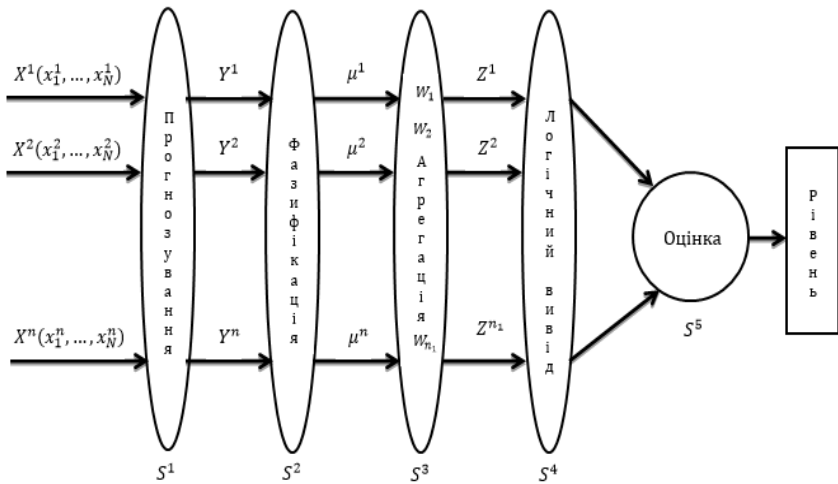


Рис.1. Структурна схема нейро-фазі мережі

На етапі прогнозування передбачається на основі історії показників за певні періоди часу спрогнозувати значення цих показників на майбутні періоди з використанням моделей та методів регресійного аналізу і машинного навчання.

Етап фазифікації включає представлення показників (критеріїв) ефективності за допомогою апарату нечіткої математики.

На етапі агрегація розглядається задача об'єднання показників, представлених у вигляді нечітких чисел, у певні групи (кластери). Кластеризація даних є процесом розподілу елементів на класи або групи так, щоб елементи в одному класі були якомога близькими, а елементи різних класів є настільки різномірними, наскільки це можливо. Для цього використовуються різні міри подібності, причому міра подібності визначає самі кластери.

Кластерний аналіз займає одне з центральних місць серед методів аналізу даних і є сукупністю підходів, методів і алгоритмів, призначених для знаходження деякого розбиття досліджуваної сукупності об'єктів на підмножини схожих між собою об'єктів. При цьому вихідним припущенням для виділення таких підмножин, що отримали спеціальну

назву кластерів, служить лише неформальне припущення про те, що об'єкти, які відносяться до одного кластера, повинні мати більшу схожість між собою, чим з об'єктами з інших кластерів.

Наступний етап передбачає, за допомогою методів логічного виведення, визначати нечітку оцінку, яка є інтегрованим показником. Для цього необхідно формування нечіткої бази знань, яка грубо відображає нелінійний взаємозв'язок «входи - вихід» за допомогою лінгвістичних правил <Якщо-то>. Ці правила генеруються експертом або отримуються в результаті екстракції нечітких знань з експериментальних даних. Параметрами, що настроюються, є ваги правил і параметри функцій належності нечітких термів.

На кінцевому етапі отримана нечітка оцінка дефазифікується у чітке значення і визначається її рівень.

Даний підхід дозволяє визначити оцінку рівня фінансової безпеки компанії через суб'єктивну ймовірність.

Література

1. Зайченко Ю.П. Нечеткие модели и методы в интеллектуальных системах [Текст]: учеб. пособие / Ю. П. Зайченко. – К.: Слово, 2008. – 341с.

2. Шаркаді М.М. Нечітке моделювання показників фінансової безпеки підприємства / Шаркаді М. М., Маляр М.М., Мазютинець Г.В. // Вісник Ужгородського університету, серія «Математика і інформатика». – 2020. – № 2 (37). – С. 176-183.

НЕЧІТКЕ МОДЕЛЮВАННЯ В МЕДИЧНІЙ ДІАГНОСТИЦІ

Маляр-Газда Н.М., Шаркаді М.М.
Ужгородський національний університет

FUZZY MODELING IN MEDICAL DIAGNOSIS

Abstract. To diagnose a patient, the doctor must thoroughly and systematically examine the patient, that is, collect medical history, objectively examine the patient's condition, analyze clinical, laboratory, X-ray and other types

of examinations. Processing large amounts of information, often of a non-specific nature, over a limited period of time is a difficult task for a physician in which can help a pre-diagnosis support system. The article proposes an approach for establishing a preliminary diagnosis based on the theory of fuzzy sets, which will allow for a given set of symptoms to calculate the probability of a particular diagnosis by fuzzy logic..

На сьогоднішній день доцільним є впровадження систем інтелектуального аналізу даних у сферу медичної діагностики та надання медичних послуг пацієнтам. Сучасна медицина володіє великою кількістю лікувальних і діагностичних методів, що призводить до необхідності врахування багатьох факторів різного роду при встановленні діагнозу. При цьому об'єм знань неперервно збільшується, а час, відведений для прийняття рішень лікарем, зазвичай, залишається незмінним. Такий процес веде до росту лікарських помилок, що призводить до зниження ефективності системи охорони здоров'я, а також збільшення рівня смертності. На практиці це проявляється у зменшенні тривалості життя хворих та погіршенні його якості.

Для встановлення діагнозу пацієнту, лікар повинен всебічно і системно обстежити хворого, тобто, зібрати анамнез, об'єктивно дослідити стан хворого, провести аналіз клінічних, лабораторних, рентгенівських та інших видів обстеження, а також врахувати наявність супутніх захворювань у пацієнта. Все це призводить до появи задачі обробки великого об'єму інформації, яку можна вирішити використовуючи сучасні інформаційні технології. Сучасні інформаційні технології - це моделі, методи і засоби інтелектуальної обробки даних. Очевидною, при встановленні попереднього діагнозу, є проблема проведення обробки великого об'єму нечіткої інформації, вирішення якої не можливо без використання комп'ютерного моделювання. Процес оброблення медичних даних потребує залучення різного математичного апарату, який дозволить адекватно моделювати ймовірність певного діагнозу пацієнта.

Проблему встановлення попереднього діагнозу представимо у вигляді наступної математичної моделі. Припустимо, що загальні показники виражаються через n

показників симптомів $S = \{S_i, i=\overline{1, n}\}$, за якими визначається діагноз із скінченної множини хвороб $D = \{d_j, j=\overline{1, m}\}$.

Часткові показники симптомів залежать від різних параметрів ознак, тобто $S_i(X_1, X_2, \dots, X_k)$. Наприклад, вік, стать, клінічні прояви, лабораторні, біохімічні показники і т. п. Взаємозв'язок між параметрами і показниками симптомів може задаватись не тільки в числовому або аналітичному вигляді, а і у вигляді різних моделей та алгоритмів функціонування. Симптоми, як правило, поділяються на специфічні, які притаманні одній хворобі і неспецифічні, які супроводжують цілий ряд хвороб. Значення самих параметрів носять неточний і невизначений характер, тобто їм властива невизначеність інтенсивності у межах даного симптому.

Пропонується для реалізації можливості параметрів використати нечіткі множини[1] у вигляді $X_l = \{x_l, \mu_l(X_l)\}$, де $\mu_l(X_l)$ - функція належності, параметри можуть приймати значення x_l .

Враховуючи вище сказане, впливає, що для кожного варіанту-діагнозу $d \in D$ показник $S_i(d)$ є нечіткою множиною $S_i = \{s_i, \mu_{S_i(d)}(S_i)\}$, де $\mu_{S_i(d)}(S_i)$ - функція належності, яка інтерпретується як можливість того, що показник симптоматики S_i прийме значення s_i для діагнозу $d \in D$ [2]. Таким чином, поняття діагнозу представляється у вигляді нечіткої множини $d_l = \{S_i, \mu_{S_i}(d)\}$.

В загальному випадку модель задачі діагностування захворювання можна представити у вигляді таблиці 1.

Таблиця 1.

База даних належності симптомів конкретному захворюванню

S/d	d ₁	d ₂	d _m
S ₁	$\mu_{S_1}(d_1)$	$\mu_{S_1}(d_2)$	$\mu_{S_{11}}(d_m)$
.....
S _n	$\mu_{S_n}(d_1)$	$\mu_{S_n}(d_2)$	$\mu_{S_n}(d_m)$

Задача отримання узагальненого показника симптомів представляє собою деяку операцію над нечіткими подіями $S(d) = F(S_1(d), S_2(d), \dots, S_n(d))$ для будь-якого діагнозу $d \in D$.

Точність встановлення попереднього діагнозу залежить від побудови вхідної бази даних симптомів хвороб із врахуванням частоти їх проявів при діагностуванні конкретної хвороби.

Література

1. Зайченко Ю.П. Нечеткие модели и методы в интеллектуальных системах [Текст]: учеб. пособие / Ю. П. Зайченко. - К.: Слово, 2008. - 341с.

2. Коробова Л. А., Гладких Т. В. Разработка модели принятия решения для постановки диагноза заболеваний на основе нечеткой логики // Вестн. ВГУИТ. 2018. Т. 80. № 4. С. 80–89. <https://doi.org/10.20914/2310-1202-2018-4-80-8>

МЕТОД ПРОГРАМНОГО ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ КОРИСТУВАЧІВ ТА БОТІВ У РЕКОМЕНДАЦІЙНІЙ СИСТЕМІ

Мелешко Є.В.

*Центральноукраїнський національний технічний
університет*

METHOD OF SOFTWARE SIMULATION OF USERS AND BOTS IN THE RECOMMENDATION SYSTEM

Abstract. In this work, a computer model of a social network with a recommendation system was created. The proposed method of a software simulation of users, bots, and objects in a recommendation system consists of the following parts: generation of the structure of a social graph of a recommendation system and simulation of the behavior of users and bots of a recommendation system. The generation of the structure of a social graph of a recommendation system was carried out on the basis of a modified Barabashi-Albert algorithm. Modeling of user behavior was based on a generation of a dynamic graph Users-Ratings-Items.

У даній роботі було створено програмну модель соціальної мережі з рекомендаційною системою [1, 2] за допомогою графової бази даних Neo4j [3] та мови програмування Python.

Запропонований метод програмного імітаційного моделювання користувачів, ботів та об'єктів рекомендаційної системи складається з наступних частин:

- Генерація структури соціального графу рекомендаційної системи.

- Та симуляція поведінки користувачів і ботів рекомендаційної системи.

Генерація структури соціального графу рекомендаційної системи здійснювалася на основі алгоритму Барабаши-Альберт (АБА), що створює безмасштабні мережі на основі 2 умов:

Ріст. Починаючи з невеликого числа n_0 вузлів, на кожній новій ітерації додається один новий вузол з n зв'язками (де $n \leq n_0$), які з'єднують новий вузол з n різними уже існуючими вузлами.

Бажане приєднання. Ймовірність P , з якою новий вузол утворить зв'язок з деяким уже існуючим вузлом i , тим вища, чим більше зв'язків у i -го вузла:

$$P_i = \frac{k_i}{\sum_j k_j}, \quad (4)$$

де k_i – степінь i -го вузла, а в знаменнику підраховується сума всіх степенів існуючих у мережі вузлів.

У розробленому методі використовується генерація наступних підграфів: 1) Users-Friends, Users-Followers, Posts-Published, Posts-Viewed та Posts-Liked - створюються генератором графу соціальної мережі, 2) Users-Similarity, Posts-Similarity та Posts-Recommended - створюються рекомендаційною системою.

Етапи розробленого методу:

Етап 1. Генерується неорієнтований підграф Users-Friends на основі АБА.

Етап 2. Генерується неорієнтований підграф Users-Followers на основі АБА.

Етап 3. Підграфи Users-Friends та Users-Followers об'єднуються в один граф.

Етап 4. Генерується орієнтований підграф Posts-Published на основі модифікованого АБА. На першій ітерації випадковим чином обираються n_0 користувачів, які "створюють" m_0 постів. Ймовірність опублікування посту деяким користувачем:

$$P_i = \frac{k_{1i} + k_{2i} + k_{3i}}{\sum_j (k_{1j} + k_{2j} + k_{3j})} \quad (5)$$

де k_{1i} – кількість друзів у i -го вузла, k_{2i} – кількість підписників у i -го вузла, k_{3i} – кількість постів у i -го вузла, а в знаменнику підраховується сума всіх цих значень для усіх існуючих вузлів.

Етап 5. Підграф Posts-Published приєднується до загального графу.

Етап 6. Генерується орієнтований підграф Posts-Viewed на основі модифікованого АБА. Ймовірність, що пост буде переглянутий:

$$P_i = \frac{q_{1i} + q_{2i} + q_{3i}}{\sum_j (q_{1j} + q_{2j} + q_{3j})} \quad (6)$$

де q_{1i} – кількість друзів у автора i -го поста, q_{2i} – кількість підписників у автора i -го посту, q_{3i} – кількість переглядів у i -го посту, а в знаменнику підраховується сума всіх цих значень для усіх існуючих у мережі вузлів.

Етап 7. Підграф Posts-Viewed приєднується до загального графу.

Етап 8. Генерується підграф Posts-Liked. Ймовірність виставлення лайку:

$$P_i = \frac{q_{1i} + q_{2i} + q_{3i} + q_{4i}}{\sum_j (q_{1j} + q_{2j} + q_{3j} + q_{4j})} \quad (7)$$

де q_{1i} – кількість друзів у автора i -го поста, q_{2i} – кількість підписників у автора i -го поста, q_{3i} – кількість переглядів у i -го поста, q_{4i} – кількість лайків у i -го поста, а в знаменнику підраховується сума всіх цих значень для усіх існуючих вузлів.

Етап 9. Підграф Posts-Liked приєднується до загального графу.

Етап 10. Підграфи Users-Similarity , Posts-Similarity та Posts-Recommended генеруються алгоритмами рекомендаційної системи.

Були проведені дослідження [1], які показали, що параметри згенерованих соціальних графів, зокрема, такі як щільність, діаметр мережі, коефіцієнт кластеризації тощо [4], відповідають властивостям реальних соціальних мереж.

Моделювання поведінки користувачів та ботів відбувалося на основі генерації динамічного графу $\text{Users-Ratings-Items}$.

Етапи розробленого методу:

Етап 1. Ініціалізація параметрів користувачів та об'єктів системи.

Етап 2. Створення «Зерна» соціального графу.

Етап 3. Моделювання часових ітерацій. На кожній часовій ітерації до моделі приєднується деяка кількість користувачів, об'єктів, ботів. Користувачі та боти виставляють оцінки об'єктам.

Етап 4. Зупинка роботи моделі та збереження результатів.

Генерація оцінки для звичайних користувачів:

$$P_i = \frac{q_{1i} + q_{2i} + q_{3i} + q_{4i}}{\sum_j (q_{1j} + q_{2j} + q_{3j} + q_{4j})} \quad (8)$$

$$d_{u,m} = \frac{\sum_{i=0}^n |f_{u,i} - f_{m,i}|}{n}, \quad (9)$$

де $d_{u,m}$ – дистанція між користувачем u та об'єктом m у багатомірному просторі прихованих факторів; n – кількість прихованих факторів; $f_{u,i}$ – i -тий прихований фактор користувача u ; $f_{m,i}$ – i -тий прихований фактор об'єкту m ; b_u – зсув користувача в оцінюванні об'єктів (рівень вимогливості до контенту); b_m – зсув об'єкту у одержанні оцінок (рівень якості контенту); $\Psi()$ – функція, що перетворює одержане дробове

число у дискретне число з набору оцінок [0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0].

Серед існуючих моделей атак для програмного моделювання було обрано випадкову, середню та популярну атаки, – так як ці моделі найбільш вживані, універсальні та ілюструють різні стратегії атаки. Формула для генерації оцінок від ботів:

$$r_{u,m} = \begin{cases} \text{attackPattern()}, \text{якщо об'єкт} - \text{"звичайний"} \\ 5.0, \text{якщо об'єкт} - \text{"цільовий"} \end{cases}, \quad (10)$$

де `attackPattern()` – функція, що генерує оцінки для нецільових об'єктів відповідно до типу обраної моделі атаки.

Література

1. Meleshko Ye. Computer model of virtual social network with recommendation system // Innovative technologies and scientific solutions for industries. – 2019. – №2(8). – P. 80-85.

2. Мелешко Є.В., Босько В.В., Резніченко В.А. Розробка рекомендаційної системи на базі СУБД Neo4j // Збірник тез V Міжнародної науково-практичної конференції «Інформаційні технології та взаємодії (IT&I – 2018)», м. Київ, 20-21 листопада 2018 р. – Київ: КНУ. – 2018. – С. 351-352.

3. Neo4j Documentation // Official website of the graph database Neo4j. – [Electronic resource] – Access mode: <https://neo4j.com/docs/>

4. Barabási L.-A., Albert R., Jeong H. Diameter of the world-wide web // Nature, Vol. 401. – 1999. – P. 130-131.

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ГРАФОВОЇ БАЗИ ДАНИХ NEO4J ДЛЯ ЗАСТОСУВАННЯ У РОЗРОБЦІ РЕКОМЕНДАЦІЙНИХ СИСТЕМ СОЦІАЛЬНИХ МЕРЕЖ

Мелешко Є.В., Якименко М.С., Міхав В.В.

*Центральноукраїнський національний технічний
університет*

RESEARCH OF POSSIBILITIES OF GRAPH DATABASE NEO4J FOR APPLICATION IN DEVELOPMENT OF RECOMMENDATION SYSTEMS OF SOCIAL NETWORKS

Abstract. In this work, the possibility and main advantages of using the NEO4J graph database for the development of recommendation systems of social networks were researched.

Рекомендаційні системи значним чином впливають на те, яким користувачі сприймають інформаційний простір. Вибір методу представлення даних, якими оперує рекомендаційна система, має вагомий вплив, оскільки ефективний спосіб представлення даних, необхідних для роботи такої системи, може зменшити кількість потрібних ресурсів та збільшити кількість доступних алгоритмів для формування списків рекомендацій. Отже, вибір СУБД для зберігання даних рекомендаційної системи є важливою задачею.

В наш час все більшої популярності здобувають нереляційні бази даних типу NoSQL [1], зокрема графові бази даних. Часто вони бувають більш зручними, ніж традиційні бази даних, для використання у розподілених системах та хмарних сховищах. У графових базах даних інформація зберігається у вузлах графу, а взаємозв'язки між різними інформаційними об'єктами зберігаються у ребрах.

Neo4j – графова СУБД з відкритим сирцевим кодом, створена у 2003 році американською компанією Neo Technology [2]. Для здійснення запитів до цієї бази даних можна використовувати її вбудовану мову Cypher або мову обходів та зміни графів Gremlin. Також існують бібліотеки для роботи з Neo4j для багатьох мов програмування, зокрема, для Java, Python, Clojure, Ruby та PHP.

Крім API для можливості роботи з СУБД Neo4j з використанням різних мов програмування, вона також має зручний та зрозумілий візуальний інтерфейс користувача. Для доступу до нього достатньо завантажити та встановити з офіційного сайту бази даних додаток Neo4j Desktop. У даному додатку можна безпосередньо робити запити до СУБД Neo4j та одержувати результат у вигляді таблиці, графу або тексту. Цей функціонал дозволяє швидко опанувати основи роботи з даною СУБД перед тим як створювати додаток, що буде її використовувати.

СУБД Neo4j надає багато можливостей для полегшення розробки рекомендаційних систем [2-4], зокрема, слід згадати наступні з них:

- Наглядність – моделі даних у вигляді графів більш наглядні та зрозумілі, ніж реляційні моделі даних, що полегшує програмісту процес розробки.

- Швидке масштабування – структуру даних та наповнення СУБД Neo4j можна швидко редагувати та адаптувати навіть при суттєвих змінах у структурі та об'ємах вхідних даних без великих змін та перебудов, в той час як реляційні моделі потребують значної перебудови при масштабуванні.

- Продуктивність – внутрішній механізм Index-free adjacensy у Neo4j усуває необхідність у складних об'єднаннях і забезпечує обхід в реальному часі при аналізі великої кількості даних.

- Зручність – за допомогою запитів внутрішньої мови запитів Cypher у Neo4j можна реалізовувати алгоритми рекомендаційних систем значно швидше, а їх запис буде значно компактнішим, ніж у об'єктно-орієнтованих мовах програмування.

- Вбудовані алгоритми – наявні вбудовані алгоритми для роботи з графами та статистичної обробки даних, які є досить корисними при побудові рекомендаційних систем.

Розглянемо останній пункт детальніше. При розробці рекомендаційних систем часто використовуються алгоритми кластеризації даних, що дозволяють формувати персоналізовані рекомендації на основі вподобань певних груп користувачів, до

яких відноситься цільовий користувач. База даних Neo4j має вбудовану бібліотеку Graph algorithms з розпаралеленими алгоритмами для роботи з графами. Для кластеризації графів дана бібліотека містить реалізації наступних алгоритмів [2, 3]:

- Louvain (функція `algo.louvain`) – алгоритм кластеризації графів, заснований на оптимізації модулярності. Є досить популярним алгоритмом і добре працює з великими графами.

- Label Propagation (функція `algo.labelPropagation`) – алгоритм кластеризації, заснований на розмітці графу. Також є популярним та легким у використанні.

- Triangle Counting / Clustering Coefficient (функція `algo.triangleCount`) – визначає кількість трикутників, що проходять через кожен вузол у графі. Трикутник являє собою набір з трьох вузлів, в якому кожен вузол має зв'язки з усіма іншими вузлами. На основі одержаних даних визначає коефіцієнт кластеризації. Слід зазначити, що цей алгоритм, знаходить тільки трикутники у графі, а трикутник лише частковий випадок кластеру.

Усі перераховані алгоритми можуть використовуватися у процесі формування рекомендацій користувачам соціальної мережі.

Таким чином використання графової бази Neo4j як інструменту для побудови рекомендаційних систем є перспективним та може значно полегшити процес їх розробки.

Література

1. Meier A., Kaufmann M. SQL & NoSQL Databases // Springer Vieweg, Wiesbaden. – 2019. – 229 p. – [Electronic resource] – Access mode: <https://doi.org/10.1007/978-3-658-24549-8>

2. Neo4j Documentation // Official website of the graph database Neo4j. – [Electronic resource] – Access mode: <https://neo4j.com/docs/>

3. Мелешко Є.В. Дослідження засобів кластеризації графів у графовій СУБД Neo4j для виявлення співтовариств у соціальних мережах // Збірник тез VIII Міжнародної наукової конференції «Інформація. Комунікація. Суспільство», смт.

Чинадієво, 16-18 травня 2019 р. – Львів: Видавництво Львівської політехніки. – 2019. – С. 19-20.

4. Мелешко Є.В., Босько В.В., Резніченко В.А. Розробка рекомендаційної системи на базі СУБД Neo4j // Збірник тез V Міжнародної науково-практичної конференції «Інформаційні технології та взаємодії (IT&I – 2018)», м. Київ, 20-21 листопада 2018 р. – Київ: КНУ. – 2018. – С. 351-352.

ПРО ДЕЯКІ ЗАДАЧІ ВЕКТОРНОЇ ОПТИМІЗАЦІЇ

Пасенченко Ю. А.

*Військовий інститут телекомунікацій та інформатизації
ім. Героїв Крут*

ABOUT SOME PROBLEMS OF VECTOR OPTIMIZATION

Abstract. This post discusses a statistical vector optimization problems. The formulation of two tasks for product quality management and influence management is considered. The solution of problems with many criteria is reduced to problems of mathematical programming, for which there are known solution methods.

В повідомленні розглянуто статистичні задачі векторної оптимізації, які знаходять застосування в управлінні якістю продукції [3] і в управлінні впливами.

Нехай A, B, \dots – випадкові події в просторі елементарних подій Ω , $H=(H_1, H_2, \dots, H_n)$ ($P(H_i)>0$) – повна група гіпотез:

$$\sum_{i=1}^n P(H_i) = 1, H_i \cdot H_j = \emptyset, \quad \text{при } i \neq j, P(A|H_i), P(B|H_i) -$$

умовні ймовірності, вирази:

$$\sum_{i=1}^n P(A|H_i)P(H_i) = P(A),$$

$$\sum_{i=1}^n P(B|H_i)P(H_i) = P(B) \quad (1)$$

є ймовірностями випадкових подій A, B згідно формули повної ймовірності [1]. Ці вирази можна розглядати як середні значення умовних ймовірностей $P(A|H_i), P(B|H_i)$ на повній групі гіпотез H . Дисперсія випадкових подій A , на H має вигляд:

$$\begin{aligned} D(A) &= \sum_{i=1}^n (P(A|H_i) - P(A))^2 P(H_i) = \\ &= \sum_{i=1}^n P^2(A|H_i) P(H_i) - P^2(A), \end{aligned} \quad (2)$$

а середньоквадратичне відхилення $\sigma(A) = \sqrt{D(A)}$. Можна ввести також і коваріацію випадкових подій A, B на групі гіпотез H :

$$\text{cov}(A, B) = \sum_{i=1}^n (P(A|H_i) - P(A))(P(B|H_i) - P(B))P(H_i), \quad (3)$$

що дозволяє враховувати корельованість подій.

В цих припущеннях отримуємо різні оптимізаційні задачі в залежності від призначення керованих змінних, параметрів моделі, обмежень і прагнень ОПР (осіб, які приймають рішення).

І. Розглянемо задачу управління якістю. Нехай A – випадкова подія означає, що виріб задовольняє певним вимогам якості, H_i ($i=1, \dots, n$) – випадкова подія означає, що виріб виготовлений в i -му підрозділі підприємства, яке виробляє однорідну продукцію. При цьому (1) надає середню якість по усьому виробництву, а $P(A|H_i)$ – є кількісною мірою якості продукції в i -му підрозділі. Розглянемо задачу оптимального розподілу продукції якості A по n підрозділах виробництва з метою максимізації якості і мінімізації розсіювання її рівня. В цьому випадку частки виробництва $P(H_i)$ ($i=1, \dots, n$) є керованими змінними, а $P(A|H_i)$ – фіксованими параметрами моделі. Двокритеріальна оптимізаційна задача набуває вигляду:

$$P(A) \rightarrow \max, \quad \sigma^2(A) \rightarrow \min$$

$$\sum_{i=1}^n P(H_i) = 1, \quad 0 \leq a_i \leq P(H_i) \leq b_i$$

Частки $P(H_i)$ можуть задовольняти і іншим обмеженням.

Один із методів розв'язання багатокритеріальних задач полягає в їх заміні однокритеріальними. Зокрема задачу (4) можна замінити задачею нелінійного математичного програмування:

$$P(A) - k \cdot \sigma(A) \rightarrow \max \quad (5)$$

$$\sum_{i=1}^n P(H_i) = 1, \quad 0 \leq a_i \leq P(H_i) \leq b_i,$$

де параметр $k > 0$ обирається так, щоб співмірними ставали складові двох частин виразу (5) і враховувалися прагнення ОПР. Задачу (5) можливо розв'язувати засобами MS Excel [2]. Якщо виріб має кілька якостей, то отримуємо задачу векторної оптимізації з більшою кількістю критеріїв, причому $\text{cov}(A, B)$ дозволяє враховувати залежність якостей.

II. Розглянемо задачу управління впливами. Нехай випадкова подія A означає наявність впливу на певне середовище Ω , $H = (H_1, H_2, \dots, H_n)$ – розділення Ω на частини, що не перетинаються: $\Omega = \bigcup_{i=1}^n H_i$ (ймовірнісна інтерпретація: Ω – простір елементарних подій, H – повна група гіпотез), $P(A|H_i)$ означає рівень впливу в H_i . Тоді $P(A)$ згідно (1) це середній рівень впливу по усьому Ω , а $(P(H_1), P(H_2), \dots, P(H_n))$ $\sum_{i=1}^n P(H_i) = 1$, $P(H_i) > 0$ – структура розбиття Ω . Нехай ОПР оптимізує вплив. Тоді $P(H_i)$ – є фіксованими параметрами моделі, а $P(A|H_i)$ – керованими змінними.

Якщо ОПР прагне збільшити вплив, то отримує задачу:

$$P(A) \rightarrow \max, \quad \sigma^2(A) \rightarrow \min, \quad 0 \leq P(A|H_i) \leq 1 \quad (6)$$

$$0 \leq \sum_{i=1}^n P(A|H_i)P(H_i) \leq 1.$$

Якщо ОПР прагне зменшити вплив, то отримує задачу:

$$P(A) \rightarrow \min, \quad \sigma^2(A) \rightarrow \min, \quad 0 \leq P(A|H_i) \leq 1 \quad (7)$$

$$0 \leq \sum_{i=1}^n P(A|H_i)P(H_i) \leq 1.$$

Задачі (6), (7) також є векторними, можуть замінюватися задачами нелінійного математичного програмування виду (5) і розв'язуватися засобами MS Excel. Якщо є кілька впливів A, B, \dots , то їх зв'язок враховується за допомогою $\text{cov}(A, B)$ і також можливо сформулювати задачі векторної оптимізації. Причому, якщо впливи A, B, \dots є конкуруючими, то відповідні оптимізаційні задачі стають задачами теорії ігор.

Література

1. Коваленко І.Н., Гнеденко Б.В. Теория вероятностей. Учебник для университетов и вузов. – Киев: Вища школа, 1990. – 328 с.
2. Леоненков А.М. Решение задач оптимизации в среде MS Excel. – СПб : BHV, 2005. – 704 с.
3. Окрепилов В.В. Управление качеством. Учебник для вузов /2-е изд./– СПб: АОО “Издательство Наука “, 2000. – 912 с.

ПРОТОТИПИ ГРАФІВ ОБСТРУКЦІЙ ЗАДАНОГО НЕОРІЄНТОВАНОГО РОДУ

Петренюк В. І., Петренюк Д. А.

*Центральноукраїнський національний технічний
університет*

PROTOTYPES OF GRAPHS OF OBSTRUCTIONS OF A GIVEN NON-ORIENTABLE GENUS

Abstract. Прототипом (моделлю) графа-обструкції G неорієнтованого роду 2 будемо називати граф заданого більшого неорієнтованого роду, отриманий шляхом приклеювання в доступних частинах частини чи підграфа, гомеоморфно го K_5 чи $K_{3,3}$, принаймні однієї копії площинного підграфа H графа-обструкції G , чи аналогічному цьому підграфу H . Основний результат- твердження 1.

Поняття та позначення узяті з [1,2]. *Задача* полягає в побудові графів-обструкцій неорієнтованого роду на основі множини графів-обструкцій на не більш ніж 10 вершинах неорієнтованого роду 2, чи 3, а саме, повної для проективної площини чи неповної для інших поверхонь, зокрема поверхні Клейна. Прототипом (моделлю) графа-обструкції G неорієнтованого роду 2 будемо називати граф заданого неорієнтованого роду, отриманий шляхом ототождження пар точок чи частин підграфа-основи K , гомеоморфного K_5 чи $K_{3,3}$, та, принаймні, однієї копії іншого підграфа H графа-обструкції із [3,4]. Для побудови прототипів графів-обструкцій заданого обмеженого роду застосуємо комбінацію методу ϕ -перетворень та методу рекурсії, приклади використання цих методів наведено на рис. 1 та 2, відповідно. Інший спосіб - розділення на площині основної клітки графа-основи кінцевих точок приєднання підграфа H до границі основної клітки графа-основи схрещених пар графів меншого роду як аналога схрещених ребер, що розділятимуть пари кінцевих вершин пари ребер, схрещених на евклідовій площині.

Використаємо наступні визначення.

Визначення 1. Будемо називати клітковою довжиною $d_G(s_i, s_j, f)$ між границями кліток s_i, s_j із заданими на них підмножинами L_i, L_j вершин графа G , де $L_i \subseteq G^0 \cap \partial s_i$, $L_j \subseteq G^0 \cap \partial s_j$, $\{s_i, s_j\} \subseteq S_G(N, f)$, (при заданому мінімальному вкладенні f графа G до неорієнтованої поверхні N , яке реалізує число досяжності t , $t_G(M, \Sigma_0) = t$), потужність $|J|$ найменшої по включенню впорядкованої множини J , $J = \{s_i, s_{i+1}, \dots, s_{i+k}, s_j\}$, $J \subseteq S_G(N, f)$, де послідовні пари кліток мають на своїх границях, принаймні одне, спільне ребро графа G , причому підмножину множини J , складену із двох чи більше кліток зі спільною вершиною, будемо рахувати як одну клітку. Також будемо говорити, що на множині J задано особливий

простий клітковий ланцюг L_{ij} , $L_{ij} = L(s_i, s_j)$, який з'єднує клітки s_i, s_j кліткового графа для графа G .

Визначення 2. Будемо позначати через T , $T = T_G(M, \Sigma_0)$, та називати деревом досяжності множини точок M , $M = \{a_i\}_{i=1}^m$, площинного графа G пару множин $(\{s_i\}_1^t, \{L_{ij}\})$, де $\{L_{ij}\}$ - найменша по включенню множина особливих кліткових ланцюгів $L(s_i, s_j)$, що або безпосередньо, або як об'єднання кількох (зі спільним кінцем одного та початком іншого), з'єднують всі пари кліток s_i, s_j та мають найменшу суму довжин взяту по всім з $\{L_{ij}\}$.

Визначення 3. Для множини точок M , $M = \{a_i\}_{i=1}^m$, площинного графа G з числом досяжності $t_G(M, \Sigma_0)$, де $t_G(M, \Sigma_0) = t$, $t > 2$, будемо називати клітковою довжиною $d_G(M, t)$, де $d_G(M, t) = \min_{\forall f \in fGN} \sum_{\forall s_i, s_j \in T} d_G(s_i, s_j, f)$, між підмножинами L_i, L_j множини M , $M = L_i \cup L_j$, $L_i \cap L_j = \emptyset$, розташованими на границях довільних кліток $s_i, s_j, \{s_i, s_j\} \subset S_G(\Sigma_0, f)$, що реалізують число досяжності $t_G(M, \Sigma_0)$, де fGN - множина всіх неізоморфних вкладень графа G до Σ_0 , що реалізують число досяжності $t_G(M, \Sigma_0)$ множини точок M .

Визначення 4. Будемо називати множину M , $M = \{a_i\}_{i=1}^m$, точок графа G з числом досяжності t , де $t_G(M, \Sigma_0) = t$, критичною відносно кліткової довжини $d_G(M)$ при операції видалення довільного елемента a_i , якщо має місце нерівність $d_G(M \setminus a_i) < d_G(M)$, чи відносно операції стискання ребра в точку, якщо має місце нерівність $d_{Gu}(M) < d_G(M)$.

Визначення 5. Будемо називати граф G мінімальним відносно $d_G(M)$ при операції видалення або стискання в точку довільного ребра u , якщо $d_{G_u}(M) < d_G(M)$, або $d_{G_u}(M) + 1 = d_G(M)$, де G_u - граф зі стиснутим в точку ребром u

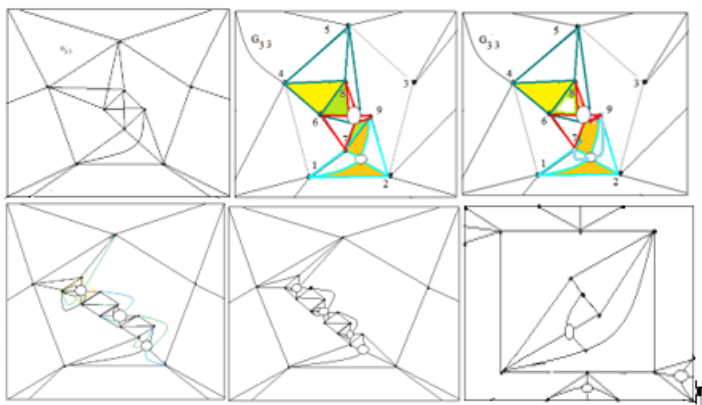


Рис. 1. Прототип граф-обструкція G_{33} мінімально вкладений на N_3 , де клітки з границями $(1,7,9,2)$ та $(4,6,8)$ мають кліткову відстань 1.

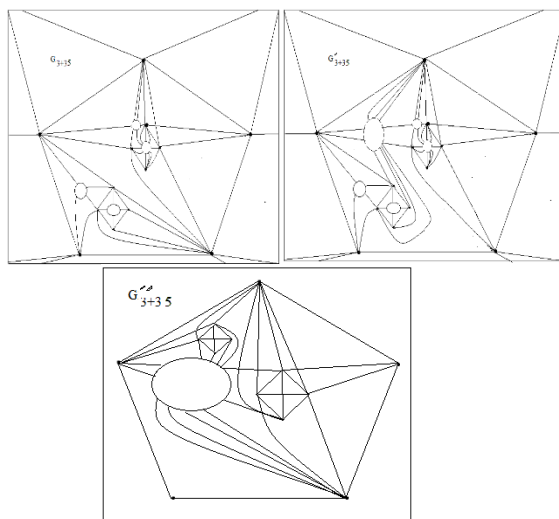


Рис 2. Графи-обструкції неорієнтованого роду 6, отримані шляхом склейки графів G_3 та G_{35} по підграфу K_5 .

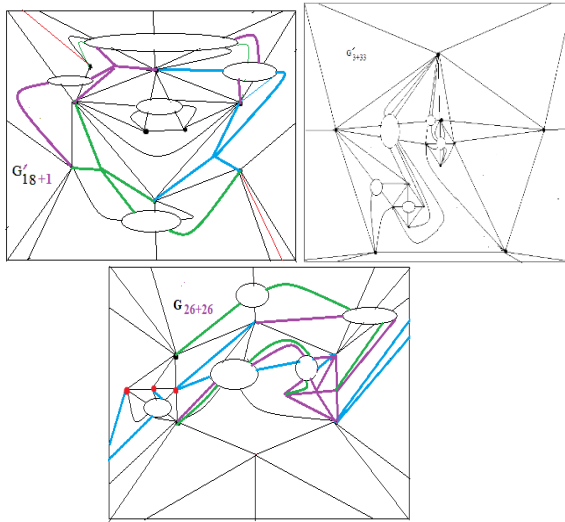


Рис 3. Графи-обструкції неорієнтованого роду 6, отримані шляхом склейки графів G_1 та G_{18} по підграфу K_5 (перша карта), графів G_3 та G_{33} по підграфу K_5 (друга карта), двох копій графа G_{26} по підграфу K_6 , (третья карта) зліва направо.

Лема 1. Мають місце наступні твердження:

1. Дві зірки, що перетинаються на площині елементарного диску висячими ребрами, можливо вкласти на приклеєній до диску ленті Мебіуса без перетину у внутрішніх точках;

2. Нехай ребро e перетинає на евклідовій площині елементарного диска ребра трикутника K_3 .

а). Ребра графа K_4 можливо вкласти на ленті Мебіуса, приклеєній до елементарного диска так, щоб пара зхрещених на евклідовій площині ребер та одне з двох паралельних ребер розташувалися на ленті Мебіуса:

б). Якщо ребро e перетинає ребро-основу трикутника, то на приклеєну ленту Мебіуса можливо вкласти як ребро e так і всі ребра трикутника, в іншому випадку можливо вкласти на приклеєну ленту Мебіуса ребро e та два суміжних з ним ребра трикутника, а ребро-основу трикутника ні;

3. Нехай два графи G_1 , G_2 мають n_1, n_2 неізоморфних мінімальних вкладень до неорієнтованої поверхні. Якщо граф

G - φ -образ графів G_1, G_2 при склейці по вершинах повних підграфів одного порядку, то граф G матиме $n_1 \cdot n_2$ неізоморфних вкладень до неорієнтованої поверхні.

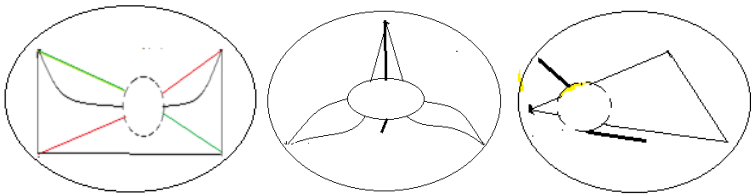


Рис 4. На першій карті граф K_4 вкладено на ленту Мебіуса, на інших проілюстровано співвідношення б) твердження 2 леми 1.

Лема 2. Нехай є простий ланцюг $L_n(H_1, H_2)$ із n графів $H_1, H_3, \dots, H_{n-1}, H_2$, гомеоморфних K_4 , приєднаний до графа H , гомеоморфного K_5 чи $K_{3,3}$ та вкладеного до проективної площини, наступним φ -перетворенням

$$\varphi(H + L_2(H_1, H_2), \sum_{i=s}^2 (h_i + e_i)) \rightarrow (G, \{h_i^*\}_{i=1}^2),$$

де пара несуміжних ребер

$e_1, e_2, e_i \in H_i^1$, ототожнюється з парою ребер h_i , що належить до границі найбільшої клітки s графа H з ребрами $h_i, i=1,2$. Мають місце наступні твердження:

1. Якщо перетини $H_1^1 \cap H_3^1, H_4^1 \cap H_3^1, \dots, H_{n-1}^1 \cap H_2^1$ пар $(H_1, H_3), (H_4, H_3), (H_{n-1}, H_2)$ відповідно, матимуть множину спільних ребер K_2^1 , то матимемо наступні співвідношення:

а). Якщо $n=2$ та ребра h_i несуміжні, то граф G є графом-обструкцією рода 2 та при видаленні спільного ребра графів H_1, H_2 одна з двох зірок зі зхрещеними ребрами вкладається до псевдоклітки s' із спільним ребром з s ;

б). Якщо $n=2$ та ребра h_i суміжні, то при видаленні спільного ребра графів H_1, H_2 одна з двох зірок зі зхрещеними ребрами вкладається до псевдоклітки s' із спільним ребром з 2-кліткою s та граф G - граф-обструкція роду 2;

в). Якщо ланцюг $L_n(H_1, H_2)$ має, принаймні, три графи, то несуттєвими ребрами графа G відносно роду при видаленні будуть ребра із H_3^1 ;

2. Якщо перетини $H_1^1 \cap H_3^1, H_4^1 \cap H_3^1, \dots, H_{n-1}^1 \cap H_2^1$ пар $(H_1, H_3), (H_4, H_3), (H_{n-1}, H_2)$, відповідно, матимуть множину спільних ребер K_3^1 , то матимемо наступні співвідношення:

а). Якщо до графа H приклеїти по простим реберним ланцюгам L_1, L_2 два різні ланцюги $L_2(H_1, H_2), L_2(H_1', H_2')$, де L_1, L_2 без спільних ребер графа H , або $L_1 = L_2$, то граф $G \in$ графом-обструкцією роду 3.

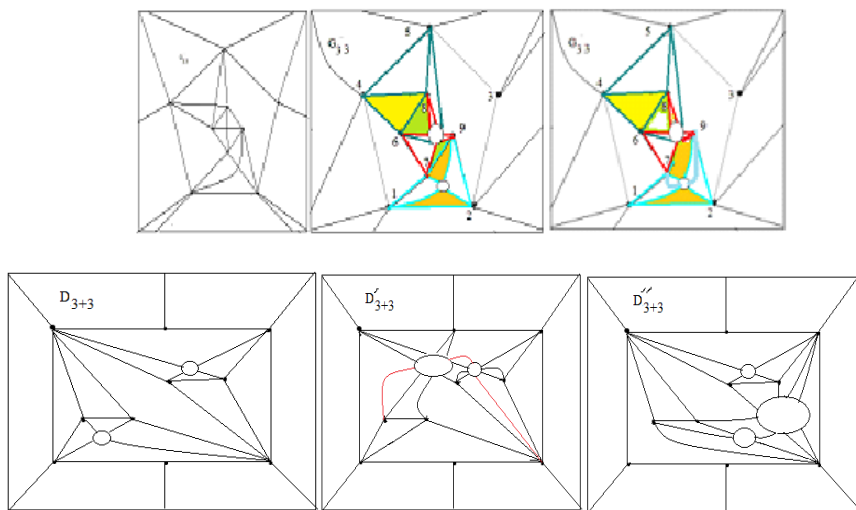


Рис 5. Графи та граф-обструкції неорієнтованого роду 3 та 4, отримані шляхом склейки по підграфу $K_{3,3}$ двох графів G_3 .

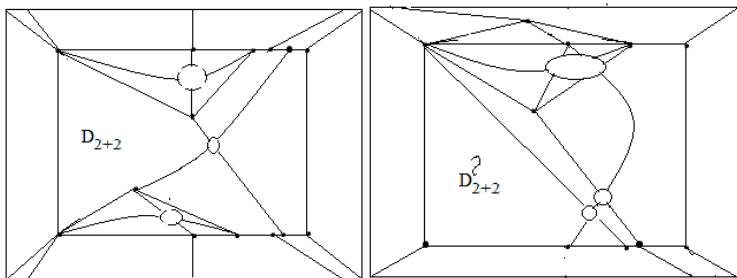


Рис 6. Прототип граф-обструкції та граф-обструкції неорієнтованого роду 3 і 4, отримані шляхом склейки графів G_2

Твердження 1. Кожен граф-обструкція G для N_k -неорієнтованої поверхні рода $k, k \geq 1$, поданий перетворенням: $\varphi(H + St_n(g_0), \sum_{i=s}^m (a_i + g_i)) \rightarrow (G, \{a_i^*\}_{i=1}^m)$, як φ -образ графа H та зірки $St_n(g_0)$ приєднаної висячими вершинами g_i до точки a_i , де $M = \{a_i\}_{i=1}^m, M \subseteq (\partial s_1 \cup \partial s_2) \cap H^0$ -задана множина точок площинного графа H розміщується на границях t кліток s_1, s_2, \dots, s_t з множини $\Sigma_0 \setminus f(H)$, де $t \geq 2, m \geq 2, i = 1, 2, \dots, m$, мінімальним вкладенням $f, f: H \rightarrow \Sigma_0$, та задовольняє наступним співвідношенням:

1. Якщо $t_H(M, \Sigma_0) = 2$, та граф H є площинним підграфом графа-обструкції G для N_1 , то для s_1, s_2 маємо нерівність $1 \leq d_H(s_1, s_2) \leq 2$;

Доведення. Доведемо співвідношення 1. Нехай G -граф-обструкція для N_1 -, поданий як φ -образ площинного графа H та зірки $St_n(g_0)$ наступним перетворенням:

$$\varphi(H + St_n(g_0), \sum_{i=s}^m (a_i + g_i)) \rightarrow (G, \{a_i^*\}_{i=1}^m), \quad \text{тобто до}$$

площинного графа H приєднано зірку $St_n(g_0)$ шляхом

ототоження пар (a_i, g_i) , утворених точками a_i графа H , та висячими вершинами g_i , де $i=1,2,\dots,m$. Будемо вважати, що $M = \{a_i\}_{i=1}^m$ - задана множина точок графа H має число досяжності 2, тобто мінімальним вкладенням f графа H до евклідової площини Σ_0 розміщується на границях двох кліток s_1, s_2 із множини кліток $\Sigma_0 \setminus f(H)$. Припустимо, що $d_H(s_1, s_2) \neq 1$. Згідно визначення 1 матимемо два наступних випадки, що вичерпують всі можливі варіанти:

- а) $d_H(s_1, s_2) = 0$, тобто границі кліток s_1, s_2 мають спільну вершину a ;
- б) $d_H(s_1, s_2) > 2$, тобто між клітками s_1, s_2 є клітковий ланцюг довжиною не менше 3.

Розглянемо випадок а). У випадку а) існуватимуть ребра e_1, e_2 , $f(e_1) \subset \partial s_1, f(e_2) \subset \partial s_2$, що мають спільну вершину a . Видалимо з цих ребер частини e_1', e_2' , $e_1' \cap e_2' \cap G^0 = a$, розщепимо вершину a на дві вершини a', a'' , які є центрами двох зірок $St_G(a')$, $St_G(a'')$, одна з яких містить $f(e_1' \cup e_2' \setminus \{a\}) \cup \{a'\}$ та розглянемо нову клітку s , яка утворилася з двох кліток s_1, s_2 при розщепленні вершини a . До клітки s приклеїмо ленту Мебіуса L на яку розмістимо вкладенням f' зірку $St_G(a')$, $f'(St_G(a')) \subset L$, тобто перехрестимо висячі ребра зірки e_1', e_2' , e_1, e_2 та ототожнимо вершини a', a'' в вершину a . Таким чином побудуємо вкладення $f': G \rightarrow N_1$, де $f'|H = f|H$, $f'|St_G^1(a'') = f|St_G^1(a'')$. Тим самим отримаємо суперечність умові про неорієнтовний рід $\gamma(G) = 2$, де G граф-обструкція для проективної площини. Припущення невірне. Випадок а) неможливий.

Розглянемо випадок б). У випадку б) немає ребер e_1, e_2 , $f(e_1) \subset \partial s_1, f(e_2) \subset \partial s_2$, зі спільною вершиною a , але матимемо клітковий ланцюг довжиною не менше 3, якій з'єднує клітки s_1, s_2 . Оскільки графа-обструкцію G для поверхні N_1 , подано як φ -образ площинного графа H та зірки $St_n(g_0)$ наступним перетворенням: $\varphi(H + St_n(g_0), \sum_{i=s}^m (a_i + g_i)) \rightarrow (G, \{a_i^*\}_{i=1}^m)$, то існуватиме вкладення f графа H в N_2 яке розміщує на границі деякої клітки s множину точок $f(M) = f(\{a_i\}_{i=1}^m)$. Побудуємо це вкладення як продовження мінімального вкладення f_0 графа H в Σ_0 при якому множина M розміщується на границі двох кліток s_1, s_2 , що з'єднані клітковим ланцюгом L , $L = L_{f(G)}(s_1, s_2)$, довжини більше 2. Для цього потрібно приклеїти до евклідової площини три ленти Мебіуса до кліткового ланцюга L , як це показано на першій карті рис.7. Розмістимо кожну пару перехрещених ребер, що утворюють кут та належать до границь послідовних кліток ланцюга L , на ленті Мебіуса наступним чином. Виріжимо на поверхні елементарний диск з центром в x -точці перехрещення двох ребер та заклеїмо цей отвір лентою Мебіуса на яку вкладемо діаметрально протилежні пари точок частин ребер, що утворилися при видаленні точки перетину. Оскільки таких кутів матимемо не менше 3, а род графа-обструкції G для N_1 дорівнює 2, то побудоване вкладення не є мінімальним. Тобто достатньо двох лент, приклеєних до евклідової площини, та у випадку вкладення G в N_2 -поверхню Клейна та розміщення множини $f(M) = f(\{a_i\}_{i=1}^m)$ на половинках границь кліток s_1, s_2 із клітковою довжиною більше 2. Тоді можливий вибір іншої пари кліток зі спільним ребром на границях, як це показано на 2-й та 3-й картах рис.7. Тобто матимемо суперечність умові, що G -

граф-обструкція для N_1 . Випадок б) неможливий. Відмітимо, що в результаті такого вкладення отримаємо клітку, на границі якої будуть розміщуватися границі всіх кліток ланцюга $L = L_{f(G)}(s_1, s_2)$, окрім тієї, що має на границі спільне ребро з границею однієї з кліток s_1, s_2 . Доведення співвідношення 1 закінчене.

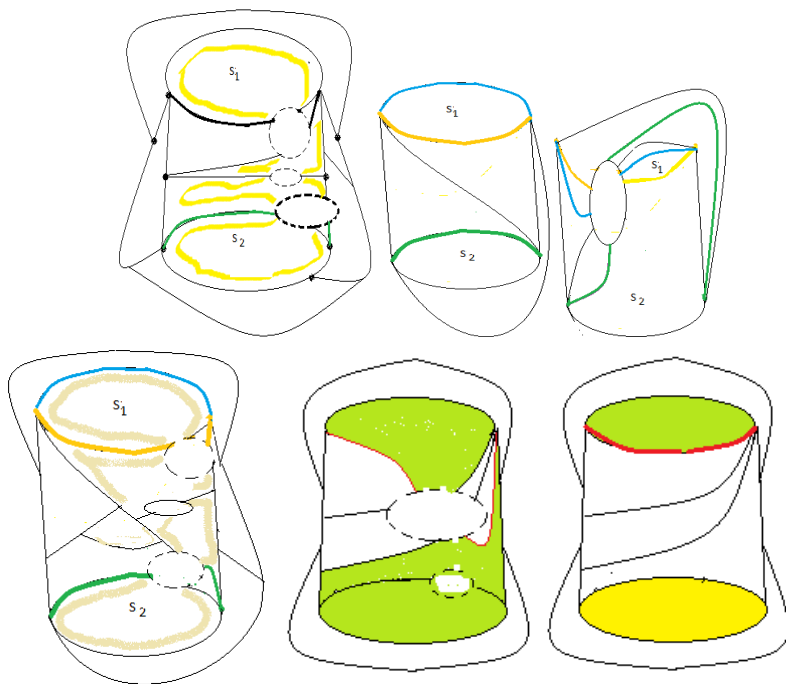


Рис. 7. Можливий вибір іншої пари кліток зі спільним ребром на границях, показано на 2-й та 3-й картах.

Література

1. Хоменко М. П. ф-перетворення графів. препринт ИМ НАНУ, Киев, 1973, 383 с.
2. Хоменко М. П. Топологические аспекты теории графов. препринт ИМ АНУ, Киев, 1970, 299 с.
3. Mohar B., Thomassen C., Graphs on Surfaces, Johns Hopkins University Press, 2001- 412 p.

4. Hur S. The Kuratowski covering conjecture for graphs of order less than 10. Phd, Ohio State University, (2008).

МЕРЕЖІ ПЕТРІ ДЛЯ МОДЕЛЮВАННЯ РОЗПОДІЛУ МАТЕРІАЛІВ

Самарай В.П.

*Відкритий міжнародний університет розвитку людини
“Україна”*

Abstracts. The research is devoted to the application of Petri nets for modeling the processes of material distribution in different modes. The designed and tested workable Petri net for sequential material distribution mode, which allows to simulate, visualize, diagnose, predict, optimize the behavior of equipment and systems for the implementation and maintenance of SCADA-systems.

Правильний оптимальний розподіл формувальної і стрижневої суміші (ФС і СС), вихідних матеріалів для ФС, шихтових і інших матеріалів у ливарних цехах представляє великий інтерес для підвищення якості виливків, коефіцієнту використання обладнання і продуктивності праці, зменшення часу виготовлення і собівартості виробів.

Науковий і практичний інтерес до таких процесів розподілу базується на поглядах з різних точок зору: моделювання процесів, оптимізація, аналіз, прогнозування і діагностування поведінки реальних систем розподілу з різними алгоритмами. Системи розподілу можуть собою представляти стрічкові і пластинчасті конвеєри з горизонтальним і вертикальним контуром.

Можна розглядати різні реалізації моделей систем розподілу матеріалів:

- а) схемо-технічні віртуальні електронні моделі;
- б) схемо-технічні реальні електронні моделі;
- в) моделі на мові GPSS (на підходах і принципах подій і ймовірності);
- г) моделі в програмі Any Logic; (на підходах і принципах подій і ймовірності);

- д) імітаційні програмні моделі на різних мовах програмування;
- є) когнітивні моделі;
- ж) моделі на принципах ланцюгів Маркова;
- з) в термінах та на принципах і підходах мереж Петрі;
- і) інші моделі.

Являє великий інтерес будувати моделі систем розподілу саме мережами Петрі.

Апарат мереж Петрі надає можливість провести ґрунтовний аналіз поведінки систем розподілу матеріалів і в статичному, і у візуальному динамічному режимах. Мережі Петрі є дводольними графами з можливістю моделювати передавання і пересування інформації, речовин, матеріалів, енергії і навіть здійснення математичних і хімічних розрахунків. Мережі Петрі реалізовані у вигляді множини багаточисельних різноманітних інтерактивних програмних продуктів, що написані на різних мовах програмування і які дозволяють у інтерактивному режимі запрограмувати і моделювати динамічну поведінку різних систем. В залежності від мети таким чином можна змодельовати будь-які принципи розподілу матеріалів і в першу чергу найбільш популярні і застосовані на виробництвах послідовний і незалежний режими і системи розподілу.

Результатом роботи стало створення схем мереж Петрі, що можуть бути реалізовані в програмних середовищах різної складності і різної потужності: в програмах малої, середньої, великої і надвеликої складності. Програми мереж Петрі малої складності дозволяють віртуально відтворити динамічну поведінку невеликих і не дуже складних систем без матричного аналізу всіх можливих станів системи і без аналізу критичних тупікових ситуацій.

Програми мереж Петрі середньої складності дозволяють віртуально відтворити динамічну поведінку більш великих і складних систем з додатковим матричним аналізом переборним методом всіх можливих станів системи і критичних тупікових ситуацій з представленням результатів у матричному вигляді.

Програми мереж Петрі великої і надвеликої складності дозволяють віртуально відтворити динамічну поведінку великих і дуже складних систем з матричним аналізом всіх

можливих станів системи і критичних тупікових ситуацій але при цьому моделювання і розрахунки можуть відбуватися для ускладнених початкових і межних умов:

- кольорові мережі (задіяна не одна змінна, а багато незалежних змінних);
- ймовірнісні мережі (у позиціях і на переходах можна організувати появи випадкових подій);
- обмежувальні багатовимірні мережі (можуть мати місце різні обмеження у різних вимірах);
- просторові багатовимірні мережі.

Побудовані мережі Петрі є модельним аналогом послідовного і незалежного розподілу матеріалів з реалізацією наступних ключових елементів таких систем:

- горизонтальна ділянка стрічкового або пластинчастого транспортерів (конвесрів);
- вертикальні ділянки бункерів (як правило 4 і більше);
- вертикальні системи датчиків рівнів;
- перемикачі підключення бункерів, формувальних і стрижневих машин;
- елементи пам'яті стану бункерів;
- ланцюги керування плужками;
- зворотні зв'язки межних верхніх і нижніх рівнів завантаження бункерів.

Мережі Петрі дозволяють здійснювати не тільки якісний але і кількісний аналіз поведінки систем розподілу матеріалів з фіксацією накопичених матеріалів. Розроблені моделі систем розподілу можуть бути базою для інших моделей теорії графів – для когнітивних моделей (когнітивного аналізу), для ланцюгів Маркова і інших і можуть бути базою для розробки систем автоматичного управління і регулювання.

Наявність таких моделей в мережах Петрі дозволяє на простих принципах теорії графів і на динамічних принципах руху і розподілу “фішок” (“маркерів”) саме в мережах Петрі не тільки моделювати, оптимізувати, діагностувати і прогнозувати процеси розподілу але і проводити обчислювальні наукові експерименти, підвищувати рівень представлення навчальних

матеріалів, навчання і вдосконалення студентів і практичних фахівців.

Резюме. Дослідження присвячене застосуванню мереж Петрі для моделювання процесів розподілу матеріалів за різними режимами. Наведено спроектовану і перевірену працеспроможну мережу Петрі для послідовного режиму розподілу матеріалів, яка дозволяє імітувати, візуалізувати, діагностувати, прогнозувати, оптимізувати поведінку обладнання і систем для впровадження і підтримки SCADA-систем.

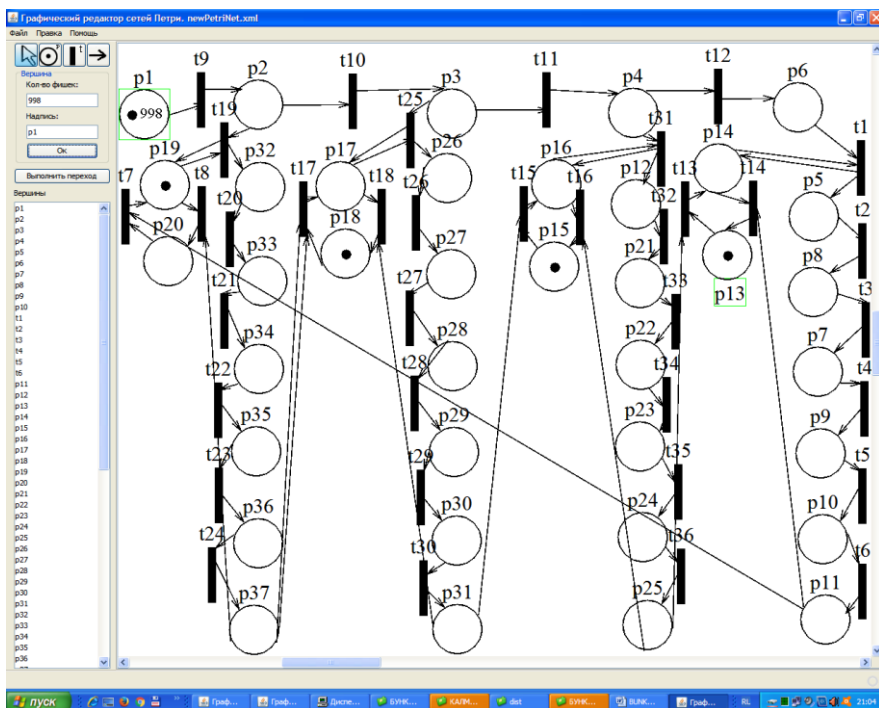


Рис. 1 Схема послідовного режиму розподілу матеріалів.

ПІДХІД ДО РОЗВ'ЯЗАННЯ ЗАДАЧІ ВЕКТОРНОЇ ОПТИМІЗАЦІЇ НА КОМБІНАТОРНІЙ КОНФІГУРАЦІЇ ПЕРЕСТАНОВОК З НЕЧІТКО ЗАДАНИМИ ДАНИМИ

Наталія Семенова, Віктор Колєчкін

*Інститут кібернетики імені В.М. Глушкова НАН
України, Київ*

THE APPROACH TO SOLVING THE VECTOR OPTIMIZATION PROBLEM ON A COMBINATORIAL CONFIGURATION OF PERMUTATIONS WITH FUZZY DATA

Abstract. The paper formulates the formulation of the vector optimization problem on the combinatorial configuration of permutations as a problem with vaguely given data. The vagueness in this problem is given in the first case as a function of the goal, in the second - in additional constraints. The properties of the domain of admissible solutions of the vector optimization problem are described. An approach to solving a problem with fuzzy data on a combinatorial permutation configuration is proposed.

Вступ. Задачі векторної оптимізації досить поширені як задачі багатокритерійного вибору зі скінченної множини альтернатив та є моделями актуальних прикладних задач, які можуть оцінюватися як кількісно, так і якісно [1-3]. Особливість таких задач, як способу математичного моделювання різних прикладних проблем, полягає у тому, що багатокритерійний вибір найбільш доцільного розв'язку здійснюється із множини непокрещуваних альтернатив. Виключне значення для розв'язання таких задач відіграє принцип Парето [2-4].

Слід зазначити, що у більшості прикладних задач формальна постановка векторної оптимізаційної моделі не просто утруднена, а й у цілому ряді випадків основні параметри можуть бути нечіткими. Методи прийняття рішень на нечітких моделях дозволяють зручно і якісно проводити оцінку альтернатив за окремими критеріями. Дослідження проблеми прийняття рішень в нечіткому середовищі стало можливим завдяки публікації статті Р. Беллмана і Л. Заде [6].

У даній роботі сформульована постановка задачі векторної оптимізації на комбінаторній конфігурації перестановок як задачі з нечітко заданими даними. Нечіткість у даній задачі задана у першому випадку у функції цілі, в другому – в додаткових обмеженнях. Таким чином, принцип Еджворта–Парето поширюється на більш широкий клас багатокритерійних задач, у яких множина допустимих розв’язків є нечіткою, або функція цілі має нечітко задані параметри. У працях [3, 7] досліджено векторні задачі з нечіткими цільовими функціями, а у [1, 4] задачі на комбінаторних множинах. Очевидно, доцільно розглянути задачу, що поєднує вищезазначені.

1. Постановка задачі векторної оптимізації на комбінаторній конфігурації перестановок з нечітко заданими даними

При формулюванні задачі багатокритерійної оптимізації як вимоги до оптимальності рішення вводиться умова обов’язкового задоволення всіх часткових критеріїв і обмежень, а саме: у точці оптимуму всі функції належності до множини оптимальних розв’язків мають бути відмінні від нуля, а критерії в оптимумі мають задовольнятися з максимально можливою мірою. При цьому збільшення значення узагальненого критерію не повиненно відбуватися при поліпшенні значення одних показників якості за рахунок погіршення інших. У термінології теорії прийняття рішень остання вимога еквівалентна умові належності точки оптимуму множині Парето [1, 3, 4].

Розглядається векторна задача комбінаторної оптимізації

$$Z(F, X) : \max \left\{ F(x) \mid x \in X \subset R^n \right\}, F(x) = (f_1(x), \dots, f_l(x)),$$

$$f_i : R^n \rightarrow R, i \in N_l, X = \text{vert } \Pi_{nk}(A) \cap D \neq \emptyset, \Pi_{nk}(A) = \text{conv } P_{nk}(A),$$

де $P_{nk}(A)$ – комбінаторна множина перестановок, $D \subset R^n$ – опуклий многогранник.

На множині X задана нечітка підмножина $\tilde{X} = \{x, \mu_{\tilde{X}}(x)\}$, де $x \in X$, а $\mu_{\tilde{X}}(x) : X \rightarrow [0, 1]$ – функція належності множині \tilde{X} . Під максимізацією будемо розуміти вибір нечіткої підмножини \tilde{R} з нечіткої множини \tilde{X} , якій відповідає найбільше значення, як

векторної функції F , так і функції належності $\mu_{\tilde{X}}(x)$ нечіткій множині альтернатив. Ці альтернативи в задачах багатокритерійної оптимізації називаються ефективними (оптимальними за Парето).

Цікавим є випадок, коли задача векторної оптимізації є задачею з нечіткою функцією цілі. Така задача передбачає знаходження розв'язків $x \in X$, які задовольняють наступне обмеження вигляду:

$$f_i(x) \leq \tilde{g}_i, f_i : R^n \rightarrow R, i \in N_l,$$

де $\tilde{g}_i, i \in N_l$, нечітка множина з функцією належності

$$\mu_{\tilde{X}}(f_k(x)) = \begin{cases} 1, f_k(x) \leq g_k, \\ 1 - \frac{f_k(x) - g_k}{t_k}, g_k \leq f_k(x) \leq g_k + t_k, \\ 0, f_k(x) \geq g_k + t_k. \end{cases}$$

Розв'язання нечіткої задачі багатокритерійної оптимізації може бути зведене до розв'язання чіткої задачі шляхом перетворення обмежень до такого вигляду

$$\alpha \rightarrow \max, \mu_k(f_k(x)) \geq \alpha, x \in X,$$

де α – рівень (зріз) нечіткої множини $\tilde{g}_i, i \in N_l$.

2. Підхід до розв'язання задачі на основі властивостей множини розв'язків

Нехай задана нечітка мультимножина

$$\tilde{A} = \{a_1, \mu_{\tilde{A}}(a_1), a_2, \mu_{\tilde{A}}(a_2), \dots, a_q, \mu_{\tilde{A}}(a_q)\}, \text{ її основа}$$

$$S(\tilde{A}) = \{e_1, \mu_{\tilde{A}}(e_1), e_2, \mu_{\tilde{A}}(e_2), \dots, e_k, \mu_{\tilde{A}}(e_k)\}, \text{ де}$$

$$\mu_{\tilde{A}}(e_i) = \min \left\{ \mu_{\tilde{A}}(a_{ij}) \mid a_{ij} = a_i, j \neq t, \forall i, j, t \in N_q \right\},$$

$e_j \in R_1 \forall j \in N_k = \{1, \dots, k\}$ і кратність елементів $k(e_j) = r_j, j \in N_k$,

$$r_1 + r_2 + \dots + r_k = q.$$

Як відомо, нечітка підмножина $P(\tilde{A})$, елементами якої є нечіткі n -вибірки з нечіткої мультимножини \tilde{A} , називається нечіткою евклідовою комбінаторною множиною, якщо для довільної пари її елементів $a = (a_1, \mu_{\tilde{A}}(a_1), a_2, \mu_{\tilde{A}}(a_2), \dots, a_n, \mu_{\tilde{A}}(a_n))$ і

$b = (b_1, \mu_{\tilde{A}}(b_1), b_2, \mu_{\tilde{A}}(b_2), \dots, b_n, \mu_{\tilde{A}}(b_n))$ виконуються умови:
 $(a \neq b) \Leftrightarrow (\exists j \in N_n : (a_j, \mu_{\tilde{A}}(a_j)) \neq (b_j, \mu_{\tilde{A}}(b_j)))$, тобто множина $P(\tilde{A})$

має таку властивість: два елементи множини $P(\tilde{A})$ відмінні один від одного, якщо вони незалежно від інших відмінностей відрізняються порядком розміщення символів, що їх утворюють і ступенем належності нечіткій підмножині $P(\tilde{A})$.

Нечітка множина перестановок з n дійсних чисел, серед яких k різних, називається загальною нечіткою множиною перестановок і позначається $P_{nk}(\tilde{A})$.

Для розв'язування вище сформульованої задачі, розглядається **метод гарантованого результату**, який дає гарний результат навіть для самого найменшого із критеріїв, тобто, компромісний розв'язок отримується шляхом розв'язання наступної задачі оптимізації: $z = \min_k f_i(x) \rightarrow \max, i=1,2,\dots,l$
 $x \in X$.

Як відомо, з урахуванням нормалізації критеріїв методи гарантованого результату складають найбільш перспективний напрям у розв'язанні задач багатокритерійної оптимізації.

Література

1. Семенова Н.В., Колечкина Л.Н., Нагорная А.Н. Векторные задачи оптимизации с линейными критериями на нечетко заданном комбинаторном множестве альтернатив. Кибернетика и системный анализ. 2011. № 2. С. 77 – 87.
2. Бекмуратов Т. Ф., Мухамедиева Д. Т. Подходы к решению задачи многокритериальной оптимизации с нечеткой целью. Проблемы информатики. 2014. № 1. С. 3–9
3. Ногин В.Д. Принцип Эджворта-Парето и относительная важность критериев в случае нечеткого отношения предпочтения. Журн. вычислит, математики и мат. физики. 2003. Т. 43. № 11. С. 1666–1676.

4. Семенова Н.В., Колечкина Л.М. Векторні задачі дискретної оптимізації на комбінаторних множинах: методи дослідження та розв'язання. Київ: Наук. думка, 2009. 266 с.

5. Ємець О.О., Ємець Ол-ра О. Деякі операції та відношення над нечіткими числами. Наукові вісті НТУУ «КПІ». 2008. № 5. С. 39–46.

6. Zadeh L.A. Fuzzy sets. Inform. and Control. 1965. Vol. 8. P. 338–353

7. Машенко С.О. Принятие решений при нечетком множестве состояний природы. Теория и методы. ГмбН: LAP Lambert Academic Publishing, 2013. 108 с.

ПРО ДОСТАТНЮ УМОВУ ІСНУВАННЯ ДИСТАНЦІЙНОЇ МАГІЧНОЇ РОЗМІТКИ ГРАФІВ

Семенюта М.Ф., Дмітрієв О.М., Сорока М.Ю.

Льотна академія НАУ

ON A SUFFICIENT CONDITION FOR THE EXISTENCE OF DISTANCE MAGIC LABELING OF GRAPHS

Abstract. We have proved one of the sufficient conditions for the existence of distance magic labeling of graphs.

Будемо розглядати зв'язні скінчені неорієнтовані графи без кратних ребер та петель. Граф вважатимемо порожнім, якщо степені всіх його вершини дорівнюють нулю. Сукупність підмножин множини S називається *розбиттям* S , якщо об'єднання цих підмножин співпадає з S і ніякі дві з них не перетинаються. Для будь-якої множини $A \subseteq V$ вершин графа $G = (V, E)$, *породженням* підграфом $G(A)$ називають максимальний підграф графа G , множиною вершин якого є A . Дві вершини з A суміжні в $G(A)$ тоді і тільки тоді, коли вони суміжні в G [1]. Підмножину $B \subseteq V$ вершин графа $G = (V, E)$ називають *клікою*, якщо породжений нею підграф $G(B)$ є

повним. Граф $G(B)$ також називають клікою. Підмножину вершин графа називають незалежною, якщо породжений нею підграф – порожній.

Поняття дистанційної магічної розмітки введено незалежно кількома авторами [2-5]. Під *дистанційною магічною розміткою* графа $G = (V, E)$ порядку n розуміють бієкцію $f: V(G) \rightarrow \{1, 2, \dots, n\}$, для якої існує таке натуральне число k , що для кожної вершини u справедлива рівність $k = \sum_{v \in N(u)} f(v)$, де $N(u)$

– множина вершин графа G суміжних з u . Число k називається *магічною сталою розмітки f* , а граф, що допускає таку розмітку, – *дистанційним магічним*. Задача знаходження дистанційних магічних графів еквівалентна задачі пошуку врівноважених неповних турнірів [6].

В даній роботі доведено наступну гіпотезу.

Гіпотеза [7]. Якщо $G = (V, E)$ – дистанційний магічний граф, відмінний від $K_{1,2,2,\dots,2}$, то множину його вершин V можна так розбити на підмножини V_1, V_2, \dots, V_p , що $|V_i| > 1$ і $G(V_i)$ є порожнім графом для кожного $i = 1, 2, \dots, p$.

Використано той факт, що кожен граф $G = (V, E)$ можна розкласти на підграфи, кожен з яких є клікою або порожнім графом, тобто існує розбиття множини вершин V на кліки та незалежні множини.

Література

1. Харари Ф. Теория графов. – М.: Мир, 1973. – 300 с.
2. Vilfred V. Σ -labelled graph and circulant graphs. // Ph. D. Thesis, University of Kerala, Trivandrum, India. – 1994.
3. Miller M., Rodger C., Simanjuntak R. Distance magic labelings of graphs // Australian Journal of combinatorics. 2003. Vol. 28. P. 305-315.
4. Sugeng K. A., Fronček D., Miller M., Ryan J., Walker J. On distance magic labeling of graphs// JCMCC – 2009. Vol. 71. P. 39-48.

5. Jinnah M.I. On Σ -labelled graphs. // In technical Proceedings of Group Discussion on Graph Labeling Problems. / B.D. Acharya, S.M. Hedge, 1999. – P. 71–77.

6. Froncek D., Kovar P., Kovarova T. Fair incomplete tournaments. // Bull. of ICA – 2006 Vol. 48, P.31-33.

7. Sugeng K.A., Froncek D., Miller M., Ryan J., Walker J. On distance magic labeling of graphs. Journal of Combinatorial Mathematics and Combinatorial Computing. – 2009. Vol.71. P. 39-48.

ЗАДАЧІ, ЩО ПРИВЕЛИ ДО ПОНЯТТЯ РОЗМІТКИ ГРАФА

Семенюта М.Ф., Донець Г.П., Марков К.

*Летная академия НАУ,
Институт кібернетики НАН України
Институт інформаційних теорій і застосувань, Софія,
Болгарія*

PROBLEMS THAT LEAD TO THE CONCEPT OF LABELINGS OF GRAPHS

Abstract. We have presented the types of problems that influenced the appearance and development of graph labeling theory.

Засновником теорії розміток вважається А. Роса, який в 1967 році запропонував декілька типів розміток, як інструмент для розкладу повного графа на ізоморфні підграфи. Зупинимось на ряді задач, які вплинули на виникнення і розвиток теорії розміток.

Задача 1. Існує особливий клас задач комбінаторної оптимізації, метою яких є призначення міток елементам графа таким чином, щоб оптимізувати певну цільову функцію. Перші задачі такого виду пов'язані з проблемою пропускної здатності. В 1950-х роках пропускна здатність вивчалася по відношенню до матриць з ненульовими елементами, що лежать у вузькій

смугі навколо головної діагоналі. Необхідно знайти таку перестановку рядків і стовпців розрядженої матриці, щоб утримати ненульові елементи у смугі, максимально наближеної до головної діагоналі. Ця задача виникла коли вперше проаналізували сталі конструкції за допомогою комп'ютерів. Оптимізація матриць за пропускну здатністю, середньою пропускну здатністю, діапазоном, профілем та антидіапазоном викликають значний інтерес протягом багатьох років через їх практичну актуальність для значного кола глобальних оптимізаційних додатків. В 1964 році Л. Харпер залучивши комбінаторні методи до знаходження мінімальної нумерації гіперкуба, одержує нові підходи до проблеми пошуку пропускну здатності матриці при кодуванні інформації [1]. Після цього було виконано чимало роботи по вирішенню вказаного питання, конкретизованої в алгоритмах та їх реальних програмах.

Задача 2. В 1962 році досліджується проблема аналогічна тій, що представлена в задачі 1. Ця проблема визначення ширини графа при пошуку відповідей на питання про мінімізацію максимальних абсолютних похибок 6-бітних кодів зображень, представлених різницями ребер в гіперкубі [2]. В загальному вигляді для лінійної цільової функції задачу 2 можна сформулювати наступним чином. Нехай $G = (V, E)$ – неорієнтований граф порядку p . Розміткою (або нумерацією) графа G називають бієкцію $\varphi: V \rightarrow \{1, 2, \dots, p\}$, а довжиною ребра $uv \in E$ – величину $|\varphi(u) - \varphi(v)|$. Найбільша довжина ребра вважається шириною розмітки φ і позначається $B(G, \varphi)$. Необхідно знайти розмітку φ^* , для якої значення $B(G, \varphi^*)$ мінімальне. Величина $B(G) = B(G, \varphi^*)$ є шириною графа G , а φ^* – оптимальною розміткою. Ця задача є NP-важкою [3] і залишається такою для деяких окремих випадків.

Задача 3. Комбінаторна теорія схем та теорія чисел також мали суттєвий вплив на появу досягнень в такому розділі теорії графів, як теорія розміток. Комбінаторна теорія схем шукає відповідь на питання, чи можна впорядкувати елементи

скінченної множини в підмножини, щоб виконувалися певні властивості «балансу». Під схемою розуміють таку пару (X, \mathcal{A}) , яка задовольняє наступним властивостям: 1) X – це множина елементів, що називаються точками; 2) \mathcal{A} представляє собою сукупність (мультимножину) непорожніх підмножин X , які називаються блоками. Схема (X, \mathcal{A}) називається симетричною, якщо кожна пара блоків перетинається рівно по λ елементам. При вивченні симетричних блок-схем з'явилися різницеві множини. (v, k, λ) -Різницевою множиною за модулем v (або циклічною (v, k, λ) -різницевою множиною) називають таку множину $D = \{d_1, \dots, d_k\}$ різних елементів \mathbf{Z}_v , що кожен ненульовий елемент $z \in \mathbf{Z}_v$ можна представити у вигляді $z \equiv d_i - d_j \pmod{v}$ точно λ різними способами. Якщо D являється (v, k, λ) -різницевою множиною за модулем v , то можна виконати перетворення: $D+a = \{d_1+a, \dots, d_k+a\} \pmod{v}$, кожне з яких є також різницевою множиною. Сукупність перетворень $\{D, D+1, \dots, D+(v-1)\}$ складається з блоків симетричної (v, k, λ) -схеми.

Поняття різницевої множини узагальнено двома способами. Перший полягає у використанні груп, відмінних від \mathbf{Z}_v . Другий спосіб приводить до поняття різницевої системи. (v, k, λ) -Різницева система породжується сукупністю D_1, \dots, D_t k -підмножин адитивної абелевої групи X розміру v , таким чином, що кожен ненульовий елемент X зустрічається точно λ раз. Досконала система різницевих множин відрізняється тим, що значення різниць за абсолютною величиною охоплює всі значення від k до $v+k-1$. Не важко помітити її зв'язок з граціозною розміткою графів. Такі системи застосовувалися для побудови граціозних розміток графів, які складаються з декількох копій повних графів різних розмірів або для доведення не існування граціозної розмітки.

Задача 4. Схожою за змістом, хоча декілька відмінною за стилем є частина теорії чисел, а саме адитивна теорія чисел. Вона займається вивченням задач про покриття цілих чисел сумами або різницями заданого виду.

Різницевим базисом по n вважають множину цілих чисел d_1, \dots, d_k для якої кожне додатне число не більше за n може бути

представлено як $d_i - d_j$ для деякої пари $i, j \leq k$. Зауважимо, що не виключається можливість використання числа більшого за n або того факту, що різниця з'являється більше одного разу. Різницевий базис з обмеженням є множиною, яка має додаткове обмеження: $0 = d_1 < \dots < d_k = n$. Основна проблема полягає у встановленні мінімальної кількості елементів базису. В 1949 році Л. Реді та А. Рені показують існування границі для $\frac{k^2(n)}{n}$, де $k(n)$ – мінімальне значення для обмеженого різницевого базису. Ці результати С. Голомб в 1970-х роках застосував при обчисленні обмежень на розмір найбільшого граціозного підграфа повного графа K_n [4-5]. Також вони пов'язані з побудовою розкладів графа на певні підграфи і сприяли одержанню такого безнадлишкового вимірювального приладу як лінійка Голомба.

Задача 5. Велика кількість комбінаторних задач може бути описано в термінах розкладів графів в заздалегідь задані підграфи.

Коли H – повний граф порядку k , то задачі існування, побудови, переліку H -розкладу графа λK_n еквівалентні аналогічним задачам для (n, k, λ) -збалансованих неповних блоксхем. Через λK_n позначають неорієнтований мультиграф на n вершинах, у якого кожна пара вершин з'єднана рівно по λ ребрам. H -розклад λK_n називається H -схемою порядку n і індексу λ або $(\lambda K_n, H)$ -схемою. Якщо G і H – довільні неорієнтовані графи, то замість $(\lambda K_n, H)$ -схеми одержимо (G, H) -розклад.

Наприкінці 1960-х років набула популярності задача, яка полягає в розробці методів оптимального розміщення елементів антенної решітки в зв'язку з явищем інтерференції. До її розв'язку має відношення еквівалентність між напівграціозною розміткою графа та лінійкою Голомба. Проблеми проектування радіоантен привели до математичної задачі побудови розкладу повного графа на копії, ізоморфні деревам. Вперше в 1963 році

на симпозіумі, що проходив в Смоленіце (Smolence), ця задача сформульована Г. Рінгелем і відома як гіпотеза Рінгеля. Більш строга її версія запропонована А. Коцігом в 1966 році.

Гіпотеза 1. (Рінгеля) [6]. Для кожного натурального числа q існує розклад повного графа K_{2q+1} на $2q+1$ ізоморфні дерева з q ребрами.

Гіпотеза 2. (Коціга) [6]. Для кожного дерева з q ребрами існує циклічний розклад повного графа K_{2q+1} на дерева, ізоморфні даному.

Ще одна гіпотеза також приписується Г. Рінгелю, це більш загальне твердження, ніж його знаменита гіпотеза 1.1.

Гіпотеза 3 (Рінгеля). [7] Для кожного натурального числа q існує розклад повного графа K_{mq+1} на ізоморфні дерева з q ребрами за умови, що m і $q+1$ не є оби два непарними.

Перспективи розвитку теоретичного фундаменту теорії розміток закладено цими гіпотезами. Перші розмітки запропоновані А. Роса в роботі «On certain valuations of the vertices of graph» 1967 року, як інструмент для розкладу повного графу на ізоморфні підграфи. З їх допомогою знайдено розв'язки гіпотез 1.1-1.3 для окремих випадків дерев.

Література

1. Harper L.H. Optimal assignments of numbers to vertices. Journal of the Society for Industrial and Applied Mathematics. 1964. Vol. 12, Issue 1 P. 131-135.
2. Chinn P. Z., Chvátalová J., Dewdney A. K., Gibbs N. E. The bandwidth problem for graphs and matrices – a survey. J. Graph Theory. 1982. – Vol. 6. – P. 223-254.
3. Papadimitriou Ch. H. The NP-completeness of the bandwidth minimization problem// Computing. – 1976. – Vol. 16. – P. 263-270.
4. Golomb S.W. How to number a graph // Graph Theory and Computing (ed. R.C. Read), Academic Press, New York, 1972. – P. 23-37.
5. G.S. Bloom and S.W. Golomb, Numbered complete graphs, unusual rulers, and assorted applications, in Theory and Applications of Graphs, Lecture Notes in Math., 642, Springer-Verlag (1978) 53-65.

6. A. Rosa On certain valuations of the vertices of a graph, Theory of Graphs (Internat. Symposium, Rome, July 1966), Gordon and Breach, N. Y. and Dunod Paris (1967) 349-355.

7. Häggkvist R. Decompositions of complete bipartite graphs. // Surveys in combinatorics, 1989. Cambridge Univ. Press. – 1989. – P. 115-147.

ЗАСТОСУВАННЯ МЕТОДУ ПАРЕТО ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧІ ОПТИМІЗАЦІЇ ВЗАЄМОДІЇ ПІЛОТА ТА АВІАДИСПЕТЧЕРА В ПОЗАШТАТНИХ СИТУАЦІЯХ

Сікірда Ю.В., Шмельова Т.Ф., Касаткін М.В.

Льотна академія Національного авіаційного університету

Національний авіаційний університет

*Харківський національний університет Повітряних Сил
ім. І. Кожедуба*

APPLICATION OF THE PARETO METHOD FOR SOLVING THE PROBLEM OF OPTIMIZATION OF INTERACTION BETWEEN PILOT AND AIR TRAFFIC CONTROLLER IN EMERGENCIES

Abstract. The possibility of using the Pareto method to solve the problem of optimizing the interaction between the pilot and air traffic controller in the process of collaborative decision making in flight emergencies is considered.

Вступ. Ефективна взаємодія пілот-авіадиспетчер є обов'язковою умовою для забезпечення безпеки польотів в стандартних умовах та позаштатних ситуаціях (ПШС). На сьогоднішній день одним з перспективних напрямків підвищення безпеки польотів ІКАО вважає можливість організації оперативного сумісного прийняття рішень (CDM – Collaborative Decision Making) пілотом та авіадиспетчером на основі діалогу між ними та оцінки інформації в реальному масштабі часу на всіх етапах польоту повітряного судна та в будь-якій польотній ситуації. Тому актуальним завданням є

розробка сучасних методів та адекватних моделей оптимізації взаємодії пілота та авіадиспетчера в процесі CDM в ПШС, застосування яких у складі інтелектуальних систем дозволить підвищити ефективність діяльності операторів в умовах неповноти та невизначеності інформації.

Основна частина. Задача оптимізації взаємодії пілота та авіадиспетчера в процесі CDM при виникненні ПШС вирішується за двома показниками ефективності: часом виконання дій з парировання ПШС оператором T , $T \rightarrow \min$; безпекою польотів, яка вимірюється часом експлуатації повітряного судна без загрози для життя та здоров'я людей S , $S \rightarrow \max$. Тому для розв'язання даної задачі можна розглянути застосування методів векторної оптимізації, зокрема, методу Парето, який дозволяє знайти такий розв'язок, за якого значення цільових функцій будуть прийнятними [1]. Зобразимо всі можливі рішення задачі оптимізації взаємодії пілота та авіадиспетчера в процесі CDM при виникненні ПШС у вигляді замкнутих множин A і B , розглянутих на площині з осями координат T , S (рис. 1).

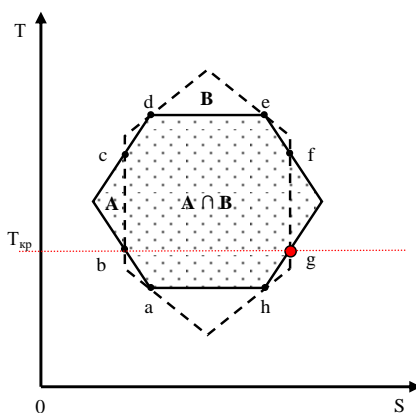


Рис. 1. Множина Парето: A – множина дій з парировання ПШС пілотом $\{a_i\}$; B – множина дій з парировання ПШС диспетчером $\{b_j\}$; $A \cap B$ – перетин множин дій з парировання ПШС пілотом A та диспетчером B

Кожна точка множин A, B відповідає одному з рішень пілота або диспетчера, яке характеризується значеннями критеріїв T, S . Множини A, B носять назву «множини ефективних планів» або «множини Парето» і характеризуються такою важливою властивістю, що на них жодне рішення не може бути покращено за одним з критеріїв без шкоди для іншого критерію [2].

Розглянемо перетин множин A і B – підмножину $A \cap B$, яка обмежена багатогранником $abcdefgh$ та складається з усіх елементів множини A , які водночас належать і множині B , та навпаки (всі елементи множини B , які належать A) і тільки них, тобто сумісних дій пілота та авіадиспетчера при виникненні ОВП (1):

$$A \cap B = \{t_{ij} \mid t_i \in A \wedge t_j \in B\}, \quad (1)$$

де t_i – час виконання дій з парирування ПШС пілотом; t_j – час виконання дій (технологічних процедур) з парирування ПШС диспетчером; t_{ij} – час сумісного виконання дій з парирування ПШС пілотом та авіадиспетчером.

Згідно рис. 1, прагнення мінімізувати рішення за критерієм T призводить до вибору точок a або h , а максимальне значення критерію S досягається при виборі рішення, якому на рис. 1 відповідають точки g або f . Підвищення безпеки польотів (критерій S) відбувається при зменшенні часу виконання дій з парирування ПШС операторами (критерій T) на відрізках ba, ef та при його збільшенні на відрізках cd, hg . Компромісне рішення з максимальною безпекою польотів S та часом сумісного виконання дій з парирування ПШС операторами T , який не перевищує критичного часу $T_{кр}$, знаходиться у точці g .

Розглянемо задачу оптимізації взаємодії пілота та авіадиспетчера в процесі CDM як задачу лінійного програмування з двома цільовими функціями та безліччю обмежень, обумовлених часом виконання дій з парирування ПШС операторами (2):

$$T_o = \sum_{i=1}^n \sum_{j=1}^m a_{Ti} b_{Tj} t_{ij} \rightarrow \min ; S_o = \sum_{i=1}^n \sum_{j=1}^m a_{Si} b_{Sj} t_{ij} \rightarrow \max ;$$

$$(2)$$

$$\sum_{i=1}^n \sum_{j=1}^m t_{ij} \leq t_{kp}, t_{ij} \geq 0, i = \overline{1, n}; j = \overline{1, m},$$

де t_{ij} – час сумісного виконання дій з парирування ПШС пілотом та авіадиспетчером; a_i, b_j – коефіцієнти, що характеризують умови, в яких здійснюється пошук спільних оптимальних рішень з парирування ПШС пілотом та авіадиспетчером відповідно; t_{kp} – критичний час виконання комплексу дій з парирування ПШС.

У загальному випадку визначення областей Парето в мультикритеріальних задачах пов'язане з подоланням значних труднощів [3]. Зокрема, задача оптимізації взаємодії пілота та авіадиспетчера в процесі CDM має велику розмірність, пов'язану з безліччю обмежень, обумовлених часом виконання дій (технологічних процедур) операторами, що затрудняє застосування класичних методів розв'язання мультикритеріальних задач, в т.ч. паретооптимального підходу. Наприклад, кількість можливих комбінацій сумісних дій пілота та авіадиспетчера $k=2$ із загальної кількості регламентованих технологічних процедур операторів у розробленій авторами тренувальній вправі «Взаємодія пілота та авіадиспетчера при відмові та пожежі двигуна на повітряному судні при наборі висоти після зльоту» [4] $l=22+22=44$ (3):

$$c_l^k = \frac{l!}{k! (l-k)!} . \quad (3)$$

$$c_{44}^2 = \frac{44!}{2! (44-2)!} = \frac{44!}{2! \cdot 42!} = 946.$$

Висновок. Двохвекторну задачу оптимізації взаємодії пілота та авіадиспетчера в процесі CDM при виникненні ПШС, яка розглядається за критеріями мінімізації часу виконання дій

з парирування ПШС операторами та максимізації безпеки польотів, доцільно розв'язувати методом Парето.

Для удосконалення CDM пілотом та авіадиспетчером в ПШС методом Парето можливе додавання до Парето-простору множин дій інших взаємодіючих суб'єктів аеронавігаційної системи (наприклад, льотного диспетчера) та застосування d -вимірної системи координат (наприклад, врахування показників регулярності та економічності польотів).

Література

1. Voronin A. Multi-Criteria Decision Making for the Management of Complex Systems. USA, Hershey: IGI Global, 2017. 201 p. DOI: 10.4018/978-1-5225-2509-7
2. Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач: монография. Москва: Физматлит, 2007. 256 с.
3. Ногин В.Д. Сужение множества Парето: аксиоматический подход. Москва: Физматлит, 2016. 272 с.
4. Касаткін М. В., Сікірда Ю. В., Шмельова Т. Ф., Гризодуб П. В. Моделювання консолідованого прийняття рішень екіпажем та диспетчером в особливих випадках в польоті. *Наука і техніка Збройних Сил Повітряних Сил України*. 2019. Вип. 1 (34). С. 30–38. DOI: 10.30748/nitps.2019.34.04

ПРО ОДНЕ УЗАГАЛЬНЕННЯ ЗАДАЧІ ПАРАЛЕЛЬНОГО УПОРЯДКУВАННЯ

Сіряк А.В., Турчина В.А.

*Дніпровський національний університет
імені Олеся Гончара*

Abstract. In this paper a problem of parallel ordering of graph vertices with non-constant number of vertices per place is considered. A known algorithm is shown to give suboptimal solutions with variable number of vertices per place. Required conditions are given for the algorithm to give an optimal solution.

Задачі паралельного упорядкування, як оптимізаційні задачі на графах, відносяться до класу задач теорії розкладів, зокрема, до тих випадків, в яких на порядок виконання завдань накладаються деякі технологічні обмеження. Ці обмеження зручно наглядно представляти орієнтованим графом [1].

Розглянемо граф $G = \{V, U\}$, в якому вершини відповідають роботам, які необхідно виконати, а дуги задають технологічні обмеження на порядок. Тобто якщо робота i безпосередньо передуює роботі j , то $(i, j) \in U$.

Паралельним упорядкуванням вершин орграфа $G = (V, U)$ називається таке лінійне упорядкування його вершин, при якому

$$\forall (i, j) \in U: i \in S[p], j \in S[q] \Rightarrow p < q.$$

Найбільш дослідженим випадком є задача побудови по заданому графу G та заданій ширині паралельного упорядкування мінімальної довжини. Виходячи з прикладних задач, що зводяться до задач паралельного упорядкування вершин орграфів, доцільно розглядати випадки, коли кількість вершин, які можна ставити на кожне місце в упорядкуванні, не обмежується однією константою.

Нехай задано послідовність натуральних чисел h_k . Необхідно побудувати таке паралельне упорядкування вершин графа G мінімальної довжини, щоб кількість вершин на місці k не перевищувала h_k .

Відомо, що для задачі побудови паралельного упорядкування розроблено алгоритм поліноміальної складності, що є точним для кореневих дерев при фіксованому h . Цей алгоритм базується на рівневному принципі.

Покажемо, що для узагальненої задачі цей алгоритм може давати субоптимальні результати.

Приклад 1.

Нехай є наступний граф.

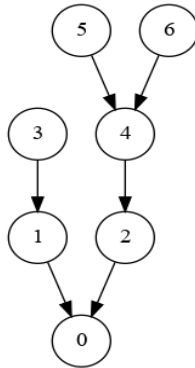


Рис.1. Граф з початковою різницею рівнів листів 1

І вектор обмежень.

$$h = (1, 3, 1, 1, \dots)$$

Тоді нескладно бачити, що алгоритм дає субоптимальний розв'язок.

$$optimal = \begin{pmatrix} 1 \\ 3, 5, 4, 2, 0 \\ 6 \end{pmatrix}; produced = \begin{pmatrix} 3 \\ 5, \frac{3}{6}, 4, 2, 1, 0 \end{pmatrix}.$$

Тому виникає питання виділення підкласу дерев та векторів обмежень h , для яких алгоритм, заснований на рівневому принципі, буде давати оптимальний розв'язок.

Зупинимося на деяких спеціальних деревах.

Лист - це вершина без вхідних дуг.

Розіб'ємо множину вершин на 2 підмножини V_1 та V_2 . В V_1 віднесемо вершини дерева, які не мають вхідних дуг, а в V_2 - вершини, що мають вхідні дуги.

Тоді під повним бінарним деревом будемо розуміти дерево, в якому в кожному вершину множини V_2 входять 2 дуги.

Рівень вершини - це відстань від вершини до кореня дерева (кількість вершин на шляху починаючи з даної до кореня).

Майже повне бінарне дерево - це дерево, в якому кожна вершина або не має вхідних дуг, або має 2 окрім, можливо, вершин передостаннього рівня, в які входять не більше ніж 2 дуги.

Сформулюємо умову, при якій алгоритм, заснований на рівневому принципі, дає оптимальний розв'язок.

Твердження 1.

Для майже повного бінарного дерева алгоритм, заснований на рівневому принципі, за умови, що $h_i \leq 2$, дає оптимальне паралельне упорядкування.

Доведення.

Наведемо конструктивне доведення цього твердження. Розглянемо крок алгоритму, заснованого на рівневому принципі.

Крок алгоритму. Нехай m - кількість листів на останньому рівні дерева.

Якщо $m > h_k$ то, згідно з алгоритмом, необхідно обрати будь-які h_k листів з останнього рівня. Після цього дерево залишиться майже повним бінарним деревом з такою самою кількістю рівнів (дійсно, кількість дуг, що входять до вершин передостаннього рівня, не могла збільшитися або зменшитися менше 0).

Якщо $m = h_k$ то, згідно з алгоритмом, необхідно обрати h_k вершин з останнього рівня. Після цього дерево стане повним бінарним деревом з кількістю рівнів на 1 меншим, ніж було (за визначенням повного бінарного дерева). Якщо рівень був останнім, то побудову завершено. Не складно бачити, що повне бінарне дерево є частинним випадком майже повного бінарного дерева, а отже дерево залишиться майже повним бінарним деревом.

Якщо $m < h_k$, то очевидно, що $m = 1$, $h_i = 2$. Позначимо лист на останньому рівні r , а вершину, в яку з r йде дуга, q (якщо така існує). Є кілька випадків:

1. Якщо рівень більше 2, то на передостанньому рівні є лист, який, згідно з алгоритмом, необхідно обрати. Наприклад, це може бути вершина q' з якої йде дуга у вершину, в яку йде дуга з q , така, що q' відмінна від q . Така обов'язково існує за визначенням майже повного бінарного дерева та є листом, оскільки єдиною вершиною на останньому рівні є вершина r , з якої йде дуга саме в вершину q .

2. Якщо рівень 2 або 1, то побудова упорядкування завершується єдиним можливим чином.

Скінченність алгоритму.

Як видно, крок алгоритму переводить майже повне бінарне дерево у майже повне бінарне дерево з меншою кількістю вершин або завершує побудову упорядкування. Оскільки кількість вершин у дереві є натуральним числом і зменшується з кожним кроком алгоритму, то побудова упорядкування завершується за скінчену кількість кроків.

Оптимальність розв'язку.

Наведемо деякі міркування щодо оптимальності побудованого упорядкування. Як видно з побудови, на всі місця, окрім, можливо, останніх 2, в паралельному упорядкуванні знайшлась максимально можлива кількість вершин. Останнє місце завжди містить 1 вершину (корінь дерева) незалежно від обмежень на кількість вершин на ньому. Це місце не може містити ніяких інших вершин оскільки з інших вершин є шлях у корінь, а тому вони мають бути раніше в упорядкуванні. Таким чином, вершина з передостаннього місця в упорядкуванні (якщо вона там одна, а ширина була 2) не може бути перенесена ні на яке інше місце, оскільки всі вони вже містять максимально можливу кількість вершин. Таким чином, побудоване упорядкування є найбільш коротким, а отже оптимальним.

Крім алгоритмів, що базуються на рівневному принципі вибору вершин, потребують дослідження і інші методи розподілу пріоритетів вершин [2].

Література

1. Бурдюк В.Я. Алгоритмы параллельного упорядочения: [Текст] / В.Я. Бурдюк, В.А. Турчина– Днепропетровск: ДГУ, 1985. – 84 с.

2. Турчина В.А. Про методи розстановки пріоритетів вершин графа для задачі побудови паралельного упорядкування / В.А. Турчина, М.П. Пасинков // Питання прикладної математики і математичного моделювання: зб. наук. пр. - Д.: Вид-во "Ліра", 2014. - С. 189-199.

ПЕРЕРАХУВАННЯ УСІХ T_0 -ТОПОЛОГІЙ, ЗАДАНИХ ОДНИМ ВЕКТОРОМ, ТА ПІДРАХУНОК T_0 -ТОПОЛОГІЙ ПЕВНОЇ ВАГИ

Скрябіна А. В., Стеганцева П. Г.
Запорізький національний університет

ENUMERATION ALL OF THE T_0 -TOPOLOGIES THAT ARE SPECIFIED BY ONE VECTOR, AND COUNTING THE T_0 -TOPOLOGIES OF A CERTAIN WEIGHT

Abstracts. The topologies on a finite set have been described with the help of the topology vector. The properties of the dual and auto-dual topologies have been considered, and all the T_0 -topologies with weight $7 \cdot 2^{n-4}$ have been calculated.

У сучасних дослідженнях топологій на скінченній множині X застосовують методи теорії графів, частково впорядкованих множин, булевих функцій, гомотопічної топології та інші. В роботі [1] T_0 -топології досліджувались за допомогою вектора топології – впорядкованого набору цілих невід’ємних чисел, що визначають мінімальні околиці елементів множини X . Задання топології вектором не є однозначним – негомеоморфні топології можуть задаватись однаковими векторами. Тому цікавою є задача знаходження всіх топологій, які задаються одним вектором. Для дослідження топологій в певному класі важливо розпізнавати взаємно двоїсті та самодвоїсті топології.

Будемо позначати символами f_τ і $f_{C\tau}$ 2-КНФ топології τ та її доповнення $C\tau$ відповідно.

Топології τ і $C\tau$, а також відповідні їм 2-КНФ f_τ та $f_{C\tau}$ будемо називати *взаємно двоїстими*. Якщо після переходу від f_τ до $f_{C\tau}$ виявиться, що f_τ та $f_{C\tau}$ еквівалентні, то кожному з них будемо називати *самодвоїстою* 2-КНФ, а відповідні їм гомеоморфні топології – *самодвоїстими*.

Індексом елемента $a \in X$ в топології τ назвемо число $ind_\tau(a)$, яке дорівнює кількості відмінних від a елементів у її

мінімальному околі M_a [1]. Послідовність невід'ємних цілих чисел $(\alpha_1, \alpha_2, \dots, \alpha_n)$ є вектором деякої T_0 -топології на n -елементній множині тоді і тільки тоді, коли вона задовольняє наступним умовам: а) $\alpha_i \leq \alpha_{i+1}$, $i = \overline{1, n-1}$, б) $\alpha_i \leq i-1$, $i = \overline{1, n}$.

Нехай T_0 -топологія τ задана вектором $(\alpha_1, \alpha_2, \dots, \alpha_n)$. Тоді можна записати мінімальну базу деякої T_0 -топології у вигляді $B = \{\emptyset, M_1, \dots, M_n\}$, де M_i – мінімальний окіл точки x_i . Якщо врахувати всі можливі зв'язки між мінімальними околами M_i , то отримаємо всі T_0 -топології з заданим вектором.

Наприклад, якщо вектор T_0 -топології має вигляд $(0, \dots, 0, 1, 3)$, то можливі три випадки для мінімальних околів M_{n-1} та M_n : а) $M_{n-1} \subset M_n$; б) $M_{n-1} \cap M_n = \{x_i\}$; в) $M_{n-1} \cap M_n = \emptyset$. Будемо розглядати в кожному випадку лише одну з гомеоморфних топологій. Тоді відповідні мінімальні бази можна записати у вигляді:

- а) $\{\emptyset, \{x_1\}, \dots, \{x_{n-2}\}, \{x_1, x_{n-1}\}, \{x_1, x_2, x_{n-1}, x_n\}\}$;
- б) $\{\emptyset, \{x_1\}, \dots, \{x_{n-2}\}, \{x_1, x_{n-1}\}, \{x_1, x_2, x_3, x_n\}\}$;
- в) $\{\emptyset, \{x_1\}, \dots, \{x_{n-2}\}, \{x_1, x_{n-1}\}, \{x_2, x_3, x_4, x_n\}\}$.

T_0 -топології з одним вектором не обов'язково мають однакову кількість елементів (вагу T_0 -топології). В наведеному прикладі T_0 -топології з базами а) та б) мають вагу $7 \cdot 2^{n-4}$, а з базою в) – $27 \cdot 2^{n-6}$.

З іншого боку, T_0 -топології, задані різними векторами, можуть мати однакову вагу. Наприклад, T_0 -топологія з вектором $(0, \dots, 0, 2, 2)$ у випадку $M_{n-1} \cap M_n = \{x_i, x_j\}$ теж має вагу $7 \cdot 2^{n-4}$.

Отже, використання поняття вектору топології дозволяє розв'язувати задачу перерахування T_0 -топологій певної ваги.

Для задачі підрахунку всіх помічених T_0 -топологій певної ваги корисним інструментом є їх задання за допомогою 2-КНФ

булевих функцій [2]. Це пояснюється тим, що цей інструмент є зручним для розпізнавання взаємно двоїстих та самодвоїстих T_0 -топологій.

Розглянемо T_0 -топології з вагою $7 \cdot 2^{n-4}$. В роботі [3] показано, що такі T_0 -топології мають вектори: а) $(0, \dots, 0, 1, 3)$ у випадку $M_{n-1} \subset M_n$; б) $(0, \dots, 0, 1, 3)$ у випадку $M_{n-1} \cap M_n = \{x_i\}$; в) $(0, \dots, 0, 2, 2)$ у випадку $M_{n-1} \cap M_n = \{x_i, x_j\}$. У випадку а) мінімальна база має вигляд $B_1 = \{\emptyset, \{x_1\}, \dots, \{x_{n-2}\}, \{x_1, x_{n-1}\}, \{x_1, x_2, x_{n-1}, x_n\}\}$, а максимальна 2-КНФ $f_1 = (x_1 \vee \bar{x}_{n-1})(x_1 \vee \bar{x}_n)(x_2 \vee \bar{x}_n)(x_{n-1} \vee \bar{x}_n)$. Заміною $x_i \leftrightarrow \bar{x}_i$ отримаємо 2-КНФ двоїстої T_0 -топології і висновок про її несамодвоїстість. Аналогічний результат для б). У випадку в) мінімальна база і 2-КНФ мають відповідно вигляд $B_2 = \{\emptyset, \{x_1\}, \dots, \{x_{n-2}\}, \{x_1, x_2, x_{n-1}\}, \{x_1, x_2, x_n\}\}$ і $f_2 = (x_1 \vee \bar{x}_{n-1})(x_2 \vee \bar{x}_{n-1})(x_1 \vee \bar{x}_n)(x_2 \vee \bar{x}_n)$. Кратності $r_1 = 2$, $r_2 = 2$, $\bar{r}_{n-1} = 2$ і $\bar{r}_n = 2$ змінних допускають розбиття на пари, що визначають бієкцію, яка свідчить про еквівалентність 2-КНФ і самодвоїстість T_0 -топології.

Враховуючи цей результат, можемо порахувати число всіх помічених T_0 -топологій з вагою $7 \cdot 2^{n-4}$, воно дорівнює $\frac{9}{4} \cdot (n)_4 + (n)_5$ і співпадає з числом, наведеним в статті Kolli [4]. Отже, можна зробити висновок, що в даному класі інших топологій немає.

Література

1. Величко И. Г., Стеганцева П. Г., Башова Н. П. Перечисление топологий близких к дискретной на конечных множествах // *Известия вузов. Математика*. 2015. № 11. С. 23–31.

2. Башова Н. П. Стеганцев Е. В. 2-КНФ булевых функций, задающих топологии на конечном множестве. *Вестник ХНТУ*. 2015. № 3 (54). С. 16–20.

3. Стеганцева П.Г., Скрыбіна А.В. Топології на n -елементній множині, узгоджені з топологіями близькими до дискретних на $(n-1)$ -елементній множині. *Український математичний журнал*. Т. 73, №. 2. 2021, с. 238 -248.

4. Kolli M. Direct and elementary approach to enumerate topologies on a finite set. // *Journal of Integer Sequences*. 2007. Vol. 10. Article 07.3.1. P. 1–11.

МІНІМІЗАЦІЯ БУЛЕВИХ ФУНКЦІЙ У ПОЛІНОМНОМУ ТА ЗМІШАНОМУ БАЗИСАХ МЕТОДОМ ОБРАЗНИХ ПЕРЕТВОРЕНЬ

Соломко М. Т.

*Національний університет водного господарства та
природокористування*

MINIMIZATION OF BOOLEAN FUNCTIONS IN POLYNOMIAL AND MIXED BASES BY THE METHOD OF FIGURATIVE TRANSFORMATIONS

Abstract. The minimization of Boolean functions in polynomial and mixed bases by the method of figurative transformations is considered. The analysis of the result of simplification of the function in the Reed – Muller basis by the method of figurative transformations and the example of minimization of the function in the polynomial basis in order to compare the cost of realization of the minimum function and the number of procedural steps.

1. Вступ

Довільну булеву функцію $f(x_1, x_2, \dots, x_n)$ можна подати у поліномній нормальній формі (ПНФ) (Exclusive-OR Sum-Of-Product form – ESOP). При цьому, залежно від того, які змінні кон'юнктерів ПНФ f (усі чи деякі з них) мають або не мають знак інверсії, що визначає так звану полярність змінних, розрізняють класи AND/EXOR виразів ПНФ булевих функцій.

У загальному випадку їх називають поліномами Ріда–Маллера (Reed–Muller expressions – RM-поліноми) [1].

Розвиток мікроелектронної технології забезпечив створення елементів, які утворюють багатомісні диз'юнкції з виключенням (EXOR-елементи). Це, у свою чергу, забезпечило синтез аналогічних дворівневих AND/EXOR-схем, які містять такі ж елементи у другому каскаді. Структура цих схем описується формулами, подібними до диз'юнктивної нормально форми (ДНФ), в яких замість операторів диз'юнкції використовуються оператори диз'юнкції з виключенням.

Переваги цих формул виправдовується тим, що кількість логічних елементів у відповідних їм схемах зазвичай менша. Наприклад, після мінімізації ДНФ довільних булевих функцій чотирьох змінних міститься в середньому 4.13 кон'юнктерів, а у ESOP – тільки 3.66 [2,3]. При розгляді булевих функцій, типових для схем, що реалізують арифметичні операції, вираш ще більший. Крім того, AND/EXOR-схеми легше діагностуються [3,4].

Перспективним застосуванням при мінімізації булевих функцій є також і змішай базис.

2. Особливості застосування логічної операції супер-склеювання змінних для ПНФ булевих функцій

Для методу образних перетворень алгоритм спрощення функції з процедурою вставки двох однакових кон'юнктерів поліномної функції з наступною операцією супер-склеювання змінних може мати такий, наприклад, вигляд:

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & & \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{vmatrix} = \quad (1)$$

$$= \overline{x_1 x_3} \oplus \overline{x_1 x_2 x_3 x_4} \oplus x_1 x_2 x_3 x_4.$$

$$\begin{aligned}
& \overline{x_1 x_2 x_3 x_4} \oplus \overline{x_1 x_2 x_3 x_4} \oplus \overline{x_1 x_2 x_3 x_4} \oplus x_1 x_2 x_3 x_4 = \\
& = \overline{x_1 x_2 x_3 x_4} \oplus \overline{x_1 x_2 x_3 x_4} \oplus \overline{x_1 x_2 x_3 x_4} \oplus \\
& \oplus \overline{x_1 x_2 x_3 x_4} \oplus x_1 x_2 x_3 x_4 \oplus x_1 x_2 x_3 x_4 = \\
& = \overline{x_1 x_3} \oplus x_1 x_2 \overline{x_3 x_4} \oplus x_1 x_2 x_3 x_4.
\end{aligned} \tag{2}$$

Результати матричного (1) та алгебричного (2) способу мінімізації логічного виразу співпадають.

Порівняно складні алгоритми спрощення логічних виразів з процедурою вставки двох однакових кон'юнктернів функцій поліномного базису з наступною операцією супер-склеювання змінних (1) розширюють варіанти їхнього застосування. Це забезпечує збільшення ефективності процедури мінімізації булевих функцій у ПНФ методом образних перетворень.

3. Сингулярні функції

Для розв'язання оптимізаційних задач логічного синтезу потрібно мати ДНФ функції та RM -поліноми з мінімальною кількістю кон'юнктернів заданої функції $f(x_1, x_2, \dots, x_n)$. При цьому, якщо існує можливість вибору RM -полінома (за винятком $PPRM$ - і $NPRM$ -поліномів), то у випадку однакової кількості кон'юнктернів перевага надається RM -поліному з мінімальною сумарною кількістю літералів. А коли кількість останніх однакова мінімальним RM -поліномом вважається той, що має меншу кількість інвертованих літералів. У зв'язку з цим, кошт реалізації RM -полінома заданої функції $f(x_1, x_2, \dots, x_n)$ можна оцінювати числовим співвідношенням $k_0 / k_l / k_{in}$, де k_0 , k_l , k_{in} – кількість кон'юнктернів, літералів та інверторів відповідно [1]. Аналогічну оцінку кошту реалізації мінімальної функції можна застосувати і до ДНФ функції та, у певній мірі, до логічних функцій змішаного базису.

Переходи між базисами Буля і Ріда–Маллера здійснюються за допомогою сингулярної (особливої) функції, кон'юнктерми якої попарно ортогональні.

Для перетворення ДНФ функції у поліномну нормальну форму необхідно провести ортогоналізацію заданої функції. Наприклад, ортогоналізація функції $F(x_1, x_2, x_3, x_4) = \overline{x_1}x_3 + \overline{x_1}x_2x_4 + x_1\overline{x_2}x_3 + x_1\overline{x_3}x_4$, що задана у ДНФ методом образних перетворень має такий вигляд:

$$F(x_1, x_2, x_3, x_4) = \left| \begin{array}{ccc} 0 & 1 & \\ \textcolor{red}{0} & \textcolor{red}{1} & \textcolor{red}{1} \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{array} \right| = \left| \begin{array}{cccc} 0 & 1 & & \\ \textcolor{red}{0} & \textcolor{red}{1} & \textcolor{red}{0} & \textcolor{red}{1} \\ \textcolor{red}{0} & \textcolor{red}{1} & \textcolor{red}{1} & \textcolor{red}{1} \\ 1 & 0 & 0 & \\ 1 & 0 & 0 & \end{array} \right| = \left| \begin{array}{ccc} 0 & 1 & \\ \textcolor{red}{0} & \textcolor{red}{1} & \textcolor{red}{0} & \textcolor{red}{1} \\ 1 & 0 & 0 & \\ \textcolor{blue}{1} & & \textcolor{blue}{0} & \textcolor{blue}{0} \end{array} \right| = \left| \begin{array}{cccc} 0 & 1 & & \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & \\ \textcolor{blue}{1} & \textcolor{blue}{0} & \textcolor{blue}{0} & \textcolor{blue}{0} \\ \textcolor{blue}{1} & \textcolor{blue}{1} & \textcolor{blue}{0} & \textcolor{blue}{0} \end{array} \right| = \left| \begin{array}{ccc} 0 & 1 & \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & \\ \textcolor{blue}{1} & \textcolor{blue}{1} & \textcolor{blue}{0} & \textcolor{blue}{0} \end{array} \right|. \quad (3)$$

Кон'юнктерми функції (3) попарно ортогональні. В алгебричній формі функція (3) має вигляд (4):

$$F(x_1, x_2, x_3, x_4) = \overline{x_1}x_3 + \overline{x_1}x_2\overline{x_3}x_4 + x_1\overline{x_2}x_3 + x_1x_2\overline{x_3}x_4. \quad (4)$$

Функції (3), (4) є сингулярними, оскільки рівносильні перетворення для них можна здійснювати обравши алгебру одного з базисів – Буля $\{\wedge, \vee, \neg\}$, або Ріда–Маллера $\{\wedge, \oplus, 1\}$.

4. Результати мінімізації булевих функцій у базисі Ріда–Маллера методом образних перетворень

Рівносильні образні перетворення при мінімізації функцій у базисі Ріда–Маллера дають наступний результат:

- визначають герменевтику логічних операцій для класу еквівалентних бінарних матриць функцій базису Ріда–Маллера;

- протокол з відносно складними алгоритмами спрощення логічних виразів, який складається з процедури вставки двох однакових кон'юнктернів функцій поліномного базису з наступною операцією супер-склеювання змінних;

- забезпечують метод ортогоналізації заданих логічних функцій для встановлення сингулярних функцій;

- створюють алгебру у частині правил рівносильного перетворення булевих функцій у базисі Ріда–Маллера.

4. 1. Приклад мінімізації булевих функцій у базисі Ріда–Маллера методом образних перетворень

Спростити булеву функцію $f(x_1, x_2, x_3, x_4)$ у поліномному форматі (ПНФ), що задана Картою Карно (рис. 1) [5].

		ab			
		00	01	11	10
cd	00	1	1	1	1
	01	0	0	1	1
	11	1	1	1	1
	10	1	0	0	1

Рис. 1. Мінімізація булевої функції за допомогою Карти Карно

Рішення.

Оскільки кон'юнктерми вихідної функції попарно ортогональні (функція сингулярна), для спрощення заданої функції обираємо алгебру Ріда–Маллера. Мінімізація $f(x_1, x_2, x_3, x_4)$ (рис. 1) у поліномному форматі здійснюється наступними образними перетвореннями:

$$\begin{aligned}
 f = & \begin{vmatrix} 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 \\ 3 & 0 & 0 & 1 \\ 4 & 0 & 1 & 0 \\ 7 & 0 & 1 & 1 \\ 8 & 1 & 0 & 0 \\ 9 & 1 & 0 & 1 \\ 10 & 1 & 0 & 1 \\ 11 & 1 & 0 & 1 \\ 12 & 1 & 1 & 0 \\ 13 & 1 & 1 & 0 \\ 15 & 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{vmatrix} = \begin{vmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{vmatrix} = 1 \oplus \begin{vmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{vmatrix} = \\
= 1 \oplus & \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{vmatrix} = 1 \oplus \overline{x_1 x_3 x_4} \oplus \overline{x_2 x_3 x_4} = \overline{x_1 x_3 x_4} \oplus \overline{x_2 x_3 x_4} = (x_1 + x_3 + x_4) \oplus x_2 x_3 x_4.
 \end{aligned}$$

МПНФ функції $f(x_1, x_2, x_3, x_4)$ (рис. 1):

$$f_{\text{МПНФ}} = 1 \oplus \overline{x_1} \overline{x_3} x_4 \oplus x_2 x_3 \overline{x_4}. \quad (5)$$

Результатом спрощення функції $f(x_1, x_2, x_3, x_4)$ (рис. 1) методом образних перетворень є мінімальна функція (5), що містить шість літералів. Це на чотири літерали менше, порівняно з [5]. Ціна реалізації (5) $k_b / k_f / k_n = 3 / 6 / 3$.

Для мінімізації функції (рис. 1) застосовано алгоритм спрощення функції з процедурою вставки двох однакових кон'юнктерів ПНФ з наступною операцією супер-склеювання змінних (п. 2).

Спрощена функція $f(x_1, x_2, x_3, x_4)$ (рис. 1) у змішаному базисі:

$$f_{\text{МЗБ}} = (x_1 + x_3 + \overline{x_4}) \oplus x_2 x_3 \overline{x_4}. \quad (6)$$

Ціна реалізації (6) $k_b / k_f / k_n = 2 / 6 / 2$.

Обидві функції (5) і (6) представляють 3-рівневу логіку, однак у змішаному базисі мінімальна функція (6) має кращі показники реалізації.

Література

1. Рицар, Б.Є. Числова теоретико-множинна інтерпретація полінома Жегалкіна // Управління системами і машинами, № 1, 2013. – С. 11–19. URL:

<http://dspace.nbuv.gov.ua/handle/123456789/83125>

2. Sasao, T. Easily testable realizations for generalized Reed-Muller expressions // IEEE Trans. On Comp., June 1997. – 46, N 6. – P. 709–716. URL: <https://ieeexplore.ieee.org/document/600830>

3. Fujiwara H., Logic testing and design for testability // in Comp. Syst. Series. Cambridge, MA: Mass. Inst. Tech., 1986. DOI: 10.7551/mitpress/4317.001.0001.

URL: https://www.researchgate.net/publication/242519300_Logic_Testing_and_Design_Testability

4. Faraj Khalid. Design Error Detection and Correction System based on Reed-Muller Matrix for Memory Protection // Inter. J. of Computer Applications (0975-8887), vol. 34, no. 8, November 2011, pp. 42-55. DOI: 10.5120/4123-5929. URL:

<https://www.ijcaonline.org/archives/volume34/number8/4123-5929>

5. Акинина, Ю.С. Подвальный, С.Л. Тюрин С.В. Применение Карт Карно для полиномиального преобразования булевых функций // Информатика, вычислительная техника и управление. Весник Воронежского государственного технического университета, 2016. URL: <https://cyberleninka.ru/article/n/primenenie-kart-karno-dlya-polinomialnogo-preobrazovaniya-bulevyh-funktsiy>

ЗАСТОСУВАННЯ ПАКЕТУ GLPK ДЛЯ ЗНАХОДЖЕННЯ ВСІХ РОЗВ'ЯЗКІВ ЗАДАЧІ ПРО К-ПЛЕКС

Стецюк П.І., Хом'як О.М.

Інститут кібернетики імені В.М. Глушкова

APPLICATION OF THE GLPK PACKAGE TO FIND ALL SOLUTIONS OF THE K-PLEX PROBLEM

Abstract. An algorithm for finding all maximum k -plexes for an undirected graph is developed. The algorithm is based on the sequential addition of the linear Boolean programming problem with an additional constraint, which cuts off the already found maximum k -plexes. The algorithm is implemented in the Octave language using the GLPK (GNU Linear Programming Kit). The computational results for searching of all maximum k -plexes at $k = 1, 2, \dots, 8$ for the graph "lzc.128" and of all maximum cliques for graphs of the type "lzc" with 128, 256, 512, 1024 vertices are given.

Задачі пошуку підмножин із множини вершин графа з наперед заданими властивостями мають широке прикладне застосування. Однією з таких задач є задача знаходження максимального k -плекса в неорієнтованому графі [1], яка активно використовується при аналізі комунікацій в соціальних

мережах, аналізі комп'ютерних мереж, машинному розпізнаванні образів і т.ін. Навіть для графів не дуже великих розмірів ця задача виявляється однією з найбільш складних та цікавих в комп'ютерній науці. Нижче розглянемо алгоритм для знаходження усіх максимальних k -плексів у неорієнтованому графі.

Нехай $G=(V,E)$ – неорієнтований граф із множиною вершин $V=\{1,\dots,n\}$ та множиною ребер E , k – деяке натуральне число, $k=1,2,\dots$. Підмножина вершин S із V називається k -плексом графа G , якщо ступінь кожної вершини в індукованому підграфі $G[S]$ (підграфі, породженому підмножиною S) є не менше, ніж $|S|-k$. Якщо $k=1$, то k -плекс збігається з клікою (повним підграфом) графа – підмножиною вершин, кожні дві з яких з'єднані ребром. Якщо $k>1$, то k -плекс є ослабленням поняття кліки графа і відповідає слабкішим вимогам на включення вершини в k -плекс, ніж вимоги на включення вершини до кліки.

Нехай вершині $i \in V$ ($i=1,2,\dots$) відповідає $x_i \in \{0,1\}$ – булева змінна, що дорівнює одиниці, якщо $i \in S$, і нулю – якщо $i \in V \setminus S$. Граф G представимо у формі $G=(V,\Gamma)$, де $\Gamma = \{\Gamma(i), i=1,\dots,n\}$, а $\Gamma(i)$ – кінцеві вершини тих дуг, у яких початковою є вершина i , яка має ступінь $d_i = |\Gamma(i)|$. Комплементарний до G граф будемо позначати $\bar{G}=(V,\bar{\Gamma})$, де $\bar{\Gamma} = \{\bar{\Gamma}(i), i=1,\dots,n\}$, $\bar{d}_i = |\bar{\Gamma}(i)|$.

Для пошуку максимального k -плекса у графі $G=(V,\Gamma)$ розглянемо таку задачу лінійного булевого програмування [1]:

знайти

$$\rho_k(G) = \max \sum_{i \in V} x_i \quad (1)$$

при обмеженнях:

$$\sum_{j \in \Gamma(i)} x_j \leq (k-1)x_i + \bar{d}_i(1-x_i), \quad \forall i \in V, \quad (2)$$

$$x_i \in \{0,1\}, \quad \forall i \in V. \quad (3)$$

Лінійні обмеження (2) побудовані таким чином, щоб підмножина $S \subset V$ була k -плексом. Ці обмеження задають вимоги на те, щоб у підграфі $G[S]$ кількість дуг, що виходять із кожної вершини $i \in S$, була не меншою за $|S| - k$. Враховуючи, що змінна x_i дорівнює одиниці для всіх $i \in S$ та дорівнює нулю для всіх $i \in V \setminus S$, цільова функція (1) означає максимізацію кількості вершин у підмножині $S \subset V$ та дозволяє знайти k -плекс з максимальною кількістю вершин – максимальний k -плекс у графі $G = (V, \Gamma)$.

Задача (1)–(3) може мати як один, так і багато розв’язків. Для того, щоб знайти усі її розв’язки, можна використати алгоритм [2], який полягає у послідовному доповненні задачі додатковим лінійним обмеженням, яке відсікає уже знайдені розв’язки. Нехай уже знайдено m максимальних k -плексів S_j , $j = 1, \dots, m$. Додамо до задачі (1)–(3) таке сімейство лінійних обмежень:

$$\sum_{i \in V(S_j)} x_i \leq \rho_k(G) - 1/2, \quad j = 1, \dots, m. \quad (4)$$

Якщо в задачі (1)–(4) значення цільової функції дорівнює значенню цільової функції у задачі (1)–(3), то це означає, що ми знайшли новий $(m+1)$ максимальний k -плекс, який не співпадає ні з одним із m існуючих. Якщо в задачі (1)–(4) значення цільової функції є меншим за значення цільової функції у задачі (1)–(3), то це означає, що ми отримали достатню

умову того, що інших максимальних k -плексів не існує, окрім тих m , які були знайдені раніше.

Описаний алгоритм для пошуку $n_{sol} < \mathbf{ncalls}$ розв'язків у задачі (1)–(3) реалізований мовою Octave. Задачі лінійного булевого програмування (1)–(3) та (1)–(4) розв'язуються за допомогою octave-функції **glpk** [3], яка використовує відомий пакет GLPK (GNU Linear Programming Kit) для розв'язання задач лінійного програмування та змішаного цілочисельного програмування. Зауважимо, що за кожний послідовний запуск функції **glpk** алгоритм або знаходить ще один максимальний k -плекс, або закінчує свою роботу, якщо всі n_{sol} k -плексів уже знайдено. У випадку, якщо кількість розв'язків у задачі (1)–(3) є більшою за **ncalls**, то в результаті роботи алгоритму отримаємо **ncalls** максимальних k -плексів. Для того, щоб продовжити пошук нових k -плексів, потрібно збільшити величину **ncalls**.

Алгоритм перевірено на ряді тестових прикладів. Дані для графа читаються з текстового файлу у DIMACS-форматі. Нижче наведемо результати обчислювальних експериментів для пошуку всіх максимальних k -плексів для графа “lzc.128” (128 вершин, 2240 ребер) при різних значеннях $k = 1, 2, \dots, 8$ (табл. 1) та для пошуку всіх максимальних клік у чотирьох графах типу “lzc” з 128, 256, 512, 1024 вершинами (табл. 2). Кількості вершин та ребер для графів “lzc.256”, “lzc.512” та “lzc.1024” наведені в першій колонці табл. 2. Максимальний за розмірами граф “lzc.1024” містить 1024 вершин та 33280 ребер. Обчислювальні експерименти здійснювались на комп'ютері з процесором Intel Core i5-9400f: частота 2.9 ГГц та оперативна пам'ять 16Гб.

В табл. 1 наведено кількості знайдених розв'язків задачі (1)–(3) (колонка n_{sol}); найменший (колонка t_{\min}) та найбільший (колонка t_{\max}) часи, які затрачені на розв'язок однієї із задач (1)–(3), (1)–(4); середній час (колонка t_{avr}) визначався як сумарний час, затрачений **glpk** на розв'язання задач (1)–(3), (1)–(4), розділений на $n_{sol} + 1$.

Таблиця 1 – Затрати часу (в секундах) на пошук n_{sol} максимальних k -плексів у графі “lzc.128”: $k = 1,2,...,8$,
 $ncalls = 1001$

k	n_{sol}	t_{min}	t_{max}	t_{avr}		k	n_{sol}	t_{min}	t_{max}	t_{avr}
1	2	1.09	1.11	1.10		5	630	5.93	49.04	14.25
2	2	2.61	2.71	2.66		6	56	4.68	17.80	7.85
3	2	10.32	11.39	10.84		7	196	2.74	9.73	4.66
4	728	2.92	16.08	7.14		8	>1000	19.99	218.16	58.61

Таблиця 2 – Затрати часу (в секундах) на пошук максимальних клік у графах типу “lzc” з 128, 256, 512, 1024 вершинами:
 $ncalls = 3$

n , $ E $	Знайдені максимальні кліки	t_{avr}
128, 2240	{64, 96, 112, 120, 124, 126, 127, 128} {1, 2, 3, 5, 9, 17, 33, 65}	1.10
256, 5632	{1, 2, 3, 5, 9, 17, 33, 65, 129} {128, 192, 224, 240, 248, 252, 254, 255, 256}	13.43
512, 13824	{1, 2, 3, 5, 9, 17, 33, 65, 129, 257} {256, 384, 448, 480, 496, 504, 508, 510, 511, 512}	234.92
1024, 33280	{1, 2, 3, 5, 9, 17, 33, 65, 129, 257, 513} {512, 768, 896, 960, 992, 1008, 1016, 1020, 1022, 1023, 1024}	3808.30

В табл. 2 наведено по дві знайдені максимальні кліки для чотирьох графів типу “lzc” та доведено, що інших максимальних клік немає, так як кількість знайдених клік менша за $ncalls = 3$.

Література

1. Balansundaram B., Butenko S., Hicks I.V. Clique Relaxations in Social Network Analysis: The Maximum k-plex Problem // Operations Research. – 2011. – Vol. 59, No. 1. – P. 133–142.
2. Стецюк П.І., Слабоспицька О.О., Ушакова О.О. Максимальні незалежні множини вершин графа та їх застосування в керуванні проектами // Питання прикладної математики і математичного моделювання. – Д.: РВВ ДНУ, 2016. – Вип. 16. – С. 151–162.
3. Eaton J.W., Bateman D., Hauberg S. GNU Octave Manual Version 3. – Network Theory Ltd, 2008. – 568 p.

ФОРМАЛІЗАЦІЯ ЗАДАЧІ ПОШУКУ СПОРІДНЕНOSTІ ЗА МОРФОЛОГІЧНИМИ ОЗНАКАМИ

Терещенко Е.В.

Національний університет «Запорізька політехніка»

FORMALIZATION OF SEARCHING RELATIONSHIP'S PROBLEM BY MORPHOLOGICAL CHARACTERISTICS

Abstract. The paper proposes a graph-theoretic formalization of searching relationship's problem by morphological characteristics. Two statements of searching relationship's problem by morphological characteristics are formulated in the form of optimization problems on an n-partite graph. An algorithm for solving the problem of identifying significant characteristics from a set for training is proposed. The complexity of this algorithm is estimated.

1. Вступ

Комплекс морфологічних ознак визначається генотипом живого організму. Це є підставою для пошуку генетичної спорідненості рослинного матеріалу за морфологічними ознаками якісного та кількісного характеру [1]. Метою досліджень Інституту олійних культур НААН України є встановлення можливостей використання морфологічних ознак якісного та кількісного характеру для з'ясування генетичної

спорідненості та ідентифікації ліній соняшнику. Дослідження проводили у науковій сівозміні Інституту олійних культур НААН. Технологія вирощування класична, посів ручними саджалками, спостереження всіх фаз розвитку рослин, встановлення олійності з застосуванням ЯМР[2]. Матеріалом для вивчення слугувала колекція з 39 ліній соняшнику. До дослідів 2016- 2018 рр. включено лінії соняшнику, з яких частина була створена як лінії аналогії шляхом бекросування з метою введення в геном гену морфологічної маркерної ознаки, частина ліній схожих за морфологією та з проведеною шляхом схрещування ідентифікацією, а частина відомих ліній з їх предками створеними шляхом селекційного добору. Лінії були оцінені за 34 морфологічними ознаками, масою 1000 насінин, олійністю, розмірами насіння. Математична формалізація дозволить застосовувати специфічні алгоритми вирішення задачі [3].

2. Теоретико-графова модель для розв'язання задачі пошуку спорідненості ліній соняшнику за морфологічними ознаками

Побудуємо теоретико-графову модель для розв'язання задачі пошуку спорідненості за морфологічними ознаками [4].

Введемо $(n+1)$ -дольний граф $G=(V_0, V_1, V_2, \dots, V_n, E)$, де $V_0 = \{v_k^0\}, k = \overline{1, m}$ – множина вершин, які взаємно однозначно відповідають лініям соняшника, $V_i, i = \overline{1, n}$ – вершини, що взаємно однозначно відповідають варіантам реалізації i -тої морфологічної ознаки, загальна кількість морфологічних ознак n , E – множина ребер $e_{k,l}^i = (v_k^0, v_l^i)$, де $v_k^0 \in V_0, v_l^i \in V_i, i = \overline{1, n}$, що відображає наявність l -ого варіанта реалізації i -ої морфологічної ознаки для k -ої лінії.

Розглянемо розбиття графа G на підграфи S_n , які є $(n+1)$ -вершинними зірками з центральними вершинами $v_k^0 \in V_0$ та висячими вершинами $v_l^i \in V_i, i = \overline{1, n}$. Введемо верхній індекс в ідентифікатор зірки, який відповідатиме номеру вершини $v_k^0 \in V_0$, тобто $S_n^k = (\{v_k^0, v_l^1, \dots, v_l^n\}, \{e_{k,l}^1, \dots, e_{k,l}^n\})$ – це $(n+1)$ -вершинна зірка з центральною вершиною v_k^0 .

Граф G є об'єднанням $\bigcup_{k=\overline{1,m}} S_n^k$, причому $S_n^k \cap S_n^p = \emptyset$, $k \neq p$, $k, p = \overline{1, m}$.

Застосуємо операцію перетину графів [4] на помічених зірках з додатковою умовою: будемо вважати $v^0 \equiv v_k^0 \equiv v_p^0$, для всіх $v^0 \in V_0$ та $k, p = \overline{1, m}$, $v_k^0 \in V_0$, $v_p^0 \in V_0$. Тоді перетин двох помічених зірок S_n^k та S_n^p є поміченою зіркою $S_N^{k,p}$

$$S_n^k \cap S_n^p = S_N^{k,p} = (SV, SE), \\ SV = \{v^0, v_l^1, \dots, v_z^N\}, SE = \{e_l^1, \dots, e_z^N\}, |SE| = N, N \leq n. \quad (1)$$

Узагальнення операції перетину з урахуванням додаткової умови для довільних помічених зірок

$$S_{N'}^{I'} \cap S_{N''}^{I''} = S_N^I = (SV, SE), |SE| = N, N \leq \min(N', N''). \quad (2)$$

Потужність множини ребер $|SE| = N$ може слугувати мірою спорідненості. Задачу пошуку спорідненості сформулюємо як задачу виокремлення деякої підмножини на множині зірок $\{S_n^k\}$, яка має максимальну потужність множини ребер перетину $\bigcap_{k=\overline{1,m}} S_n^k$, тобто $|SE| \rightarrow \max$.

3. Виокремлення множини значущих морфологічних ознак

Паралельно з задачею спорідненості ставиться задача про мінімальну кількість морфологічних ознак, що дозволяє визначити спорідненість ліній. Розв'язання задачі розділяється на два етапи.

На першому для кожної з q груп родичів виконується операція (1)-(2) перетину зірок, центральні вершини яких відповідають спорідненим лініям колекції. Для цього виконується операція (1) перетину графів з додатковою умовою для першого та другого представника групи. Далі виконується операція (2) для отриманого результату та наступного представника групи. Для групи родичів перетином є зірка $S_{N_x}^x$, $x = \overline{1, q}$. В результаті проведеного першого етапу отримано множину $\{S_{N_x}^x\}$. На другому етапі на множині $\{S_{N_x}^x\}$ виконується

операція складання за модулем два з тією ж додатковою умовою ототожнення вершин множини $V_0 : v^0 \equiv v_k^0 \equiv v_p^0$, для всіх для всіх $v^0 \in V_0$ та $k, p = \overline{1, m}$, $v_k^0 \in V_0, v_p^0 \in V_0$. Результатом операції є зірка $S_N^I = (SV, SE)$, множина висячих вершин $SV \setminus v^0$ якої перелічує ознаки, що відрізняють лінії між собою. Потужність множини $SV \setminus v^0$ та SE може слугувати нижньою межею міри спорідненості $N = \inf_{0 \leq x \leq q} (N_x)$. Складність алгоритму $O((m/q)!n)$. При наявності множини значущих ознак $SV \setminus v^0$ для розв'язання задачі пошуку спорідненості ліній перевіряється відповідність на множині значущих ознак потужністю N_x для кожної з q груп родичів. Якщо відповідність не знайдено, то цей екземпляр утворює нову $q+1$ групу родичів.

4. Пошук спорідненості як задача пошуку ланцюга максимальної довжини на $(n+1)$ -дольному графі

Сформулюємо задачу пошуку спорідненості двох ліній k та p як задачу на $(n+1)$ -дольному графі G : знайти ланцюг максимальної довжини L , який починається v_k^0 долі V_0 та має кінцеву вершину теж у долі V_0 : або вершину v_p^0 , або вершину v_k^0 , і в цьому випадку ланцюг є циклом. На кожному непарному місці в ланцюзі стоїть вершина долі V_0 , на парному - долі $V_i, i \in \{1, 2, 3, \dots, n\}$. При співпадині у двох лініях k та p кількості ознак N виконується співвідношення $L = 2N$. Кількість простих циклів в ланцюзі $[L/4]$.

5. Висновки та обговорення

Запропонована теоретико-графова модель дозволяє застосовувати стратегії пошуку в ширину та глибину на графах при відповідній організації структури даних.

Робота виконана в рамках договору про співробітництво між Інститутом олійних культур НААНУ та Національним університетом «Запорізька політехніка», ДБ05028 «Аналіз біометричної інформації» (2018-2021pp).

Література

6. Махова Т.В., Ведмедева К.В. Використання ідентифікаційних морфологічних ознак у колекціях соняшнику

// Генетика та селекція сільськогосподарських культур – від молекули до сорту: матеріали II інтернет конференції молодих учених (м.Київ, 30 серпня 2018р. // НААН, СГІ-ННЦ, М-во аграр. Політики та прод. України, Укр. Ін.-т експертизи сортів рослин – 2018. С.5-6. <http://sgi.in.ua/images/Price/Tezisi.pdf>

7. Методика проведення експертизи сортів рослин групи олійних на відмінність, однорідність і стабільність / Наказ Міністерства аграрної політики та продовольства України 16 грудня 2016 року No 547 <https://sops.gov.ua/uploads/page/5a5f1c971fe13.pdf>

8. Ведмедева К.В. Випробування методів встановлення спорідненості ліній сояшнику / К.В. Ведмедева, Е.В. Терещенко, А.В. Литвиненко //Олійні культури: сьогодення та перспективи. Збірник тез Міжнародної наукової інтернет-конференції (14 травня 2020 р.). Запоріжжя. ІОК НААН, 2020. – С.10-11.

9. Rausch A. (2019) Harary (1974): Graph Theory. In: Holzer B., Stegbauer C. (eds) Schlüsselwerke der Netzwerkforschung. Netzwerkforschung. Springer VS, Wiesbaden. https://doi.org/10.1007/978-3-658-21742-6_57

СИТУАЦІЯ НЕВИЗНАЧЕНОСТІ В ЗАДАЧАХ ШТУЧНОГО ІНТЕЛЕКТУ, ЩО ПОВ'ЯЗАНА З КОМБІНАТОРНИМИ КОНФІГУРАЦІЯМИ

Тимофієва Н. К.

*Міжнародний науково-навчальний центр інформаційних
технологій та систем НАН та МОН України*

SITUATION OF UNCERTAINTY IN ARTIFICIAL INTELLIGENCE PROBLEMS ASSOCIATED WITH COMBINATORY CONFIGURATIONS

Abstract. The situation of uncertainty in the problems of artificial intelligence arises not only due to fuzzy or incomplete input and current information, but also due to the special structure of combinatorial configurations, which are the argument of the objective function, and the methods of its modeling. The vagueness

of the input information in the problems of this class is due to the special structure of such a combinatorial configuration as placement with repetitions, which are modeled the input data.

Вступ. Прикладні задачі комбінаторної оптимізації, як правило, розв'язуються з урахуванням різного виду невизначеності. Вона виникає внаслідок неповної вхідної та поточної інформації, а також пов'язана з нечіткими вхідними даними. Але невизначеність в задачах цього класу проявляється завдяки особливій структури множини комбінаторних конфігурацій, які є аргументом цільової функції, та способу її моделювання.

Постановка задачі. Задача полягає у виявленні невизначеності в задачах штучного інтелекту, яка пов'язана із структурою аргументу цільової функції (комбінаторною конфігурацією).

Підхід, що пропонується. Розв'язання поставленої задачі проводиться шляхом аналізу задач штучного інтелекту, виявлення аргументу цільової функції, яким є комбінаторні конфігурації різних типів. Якщо їхня множина складається з підмножин ізоморфних комбінаторних конфігурацій, то в цій задачі має місце невизначеність, яка пов'язана з особливою структурою аргументу цільової функції. Для її вирішення використовуються різні підходи, зокрема і самоналагоджувальні алгоритми.

Невизначеність в задачах штучного інтелекту. Як показують дослідження прикладних задач із штучного інтелекту переважна їхня частина при знаходженні оптимального розв'язку потребує перебору варіантів. Перебірним же задачам властива комбінаторна природа. Тому задачі цього класу зводяться до задач комбінаторної оптимізації. Аргументом цільової функції в них є комбінаторні конфігурації.

Як в комбінаторній оптимізації, так і в задачах штучного інтелекту невизначеність пов'язана [1]: I) з неоднозначністю результату, одержаного за змодельованою цільовою функцією або вибраною мірою подібності у разі нечіткої вхідної інформації, який не задовольняє меті дослідження; II) з вибором способу оцінки точності роботи певного алгоритму; III) з

особливою структурою множини комбінаторних конфігурацій, що є аргументом цільової функції; IV) з неповною вхідною та поточною інформацією; V) з нечітко розробленими правилами обробки та оцінки інформації; VI) з неоднозначністю при виборі оптимального розв'язку за кількома критеріями в багатокритеріальній оптимізації.

Під *невизначеністю* в комбінаторній оптимізації розуміємо ситуацію, при якій внаслідок нечіткої чи неповної вхідної та поточної інформації неможливо одержати однозначний результат, або за вибраною мірою подібності, за змодельованими цільовими функціями, в яких не враховано структуру комбінаторних конфігурацій, за розробленими правилами обробки та оцінки інформації одержаний оптимальний результат не збігається з метою дослідження.

Розв'язання задач штучного інтелекту також проводиться в умовах невизначеності різних видів. При розв'язанні значної частини задач цього класу основна увага приділяється невизначеності, що пов'язана з неповною вхідною та поточною інформацією, а також з нечіткими вхідними даними. В цьому разі вирішення цієї проблеми проводять шляхом аналізу поведінки системи за певний проміжок часу. На основі отриманого аналізу встановлюється закономірність, яка враховується при прогнозуванні майбутніх результатів на поточному відліку часу.

Ситуація невизначеності, яка пов'язана із структурою комбінаторних конфігурацій.

Цей вид невизначеності виникає внаслідок того, що множина комбінаторних конфігурацій, які є аргументом цільової функції, складається з підмножин і на певному їхньому впорядкуванні закономірність зміни значень змодельованої цільової функції однакова незалежно від вхідних даних. Результат розв'язку задачі в цьому випадку – неоднозначний, завдяки чому виникає ситуація невизначеності.

Закономірність зміни значень цільової функції в задачах комбінаторної оптимізації залежить від упорядкування комбінаторних конфігурацій (аргументу) $w \in W$ [2]. Розглянемо

структуру їхньої множини W . Підмножину $W_\eta \subset W$ назвемо підмножиною ізоморфних комбінаторних конфігурацій, якщо її елементи – ізоморфні комбінаторні конфігурації (комбінаторні конфігурації, кількість елементів у яких однакова, називаються ізоморфними). Множина W складається з підмножин ізоморфних комбінаторних конфігурацій W_η . На підмножині W_η цільова функція змінюється так, як і на множині перестановок. Множини A і B , якими задається певна задача, назвемо базовими. Числове значення зв'язків між елементами цих множин назвемо вагами, які є вхідними даними. Упорядкуємо підмножини $W_\eta \subset W$ (крім перестановок) починаючи з $\eta=1$ і закінчуючи $\eta=n$ (η – кількість елементів у w , n – кількість елементів у множинах A , B).

Наведемо таку теорему.

Теорема. Якщо в задачах комбінаторної оптимізації множина W складається з підмножин W_η , а оптимізація проводиться за сумарним або середнім значенням ваг між елементами базової множини, то цільова функція на заданому вище впорядкуванні W_η в W – дискретна кусково-монотонна функція (відповідно неспадна або незростаюча) незалежно від структури вхідних даних.

В задачі кластеризації, яка має місце в задачах штучного інтелекту, оговорена в теоремі ситуація невизначеності вирішується шляхом розроблення самоналагоджувальних алгоритмів або введенням в процесі розв'язання задачі змінних критеріїв [3]. В задачі клінічної діагностики моделюються кілька цільових функцій. Якщо за цими функціями одержуємо оптимальний розв'язок для однієї і тієї ж комбінаторної конфігурації (аргументу), то одержаний результат збігається з метою дослідження. В іншому разі виникає ситуація невизначеності. В розпізнаванні мовлення невизначеність також пов'язана із структурою комбінаторних конфігурацій, які можуть бути як аргументом цільової функції, так і вхідними даними. Сигнали, які відповідають одному і тому ж слову,

промовленому багато разів одним і тим же диктором або різними дикторами, відрізняються між собою. Таких різних варіантів може бути нескінченна кількість тому, що складові слова – фонемі утворюються комбінацією елементів мовленнєвого тракту і утворюють комбінаторну конфігурацію – розміщення з повтореннями. Звідси має місце нечіткість вхідних даних.

Висновок. Отже, в задачах штучного інтелекту комбінаторні конфігурації можуть бути як аргументом цільової функції так і вхідними даними. В цих задачах ситуація невизначеності виникає внаслідок особливої структури множини комбінаторних конфігурацій, які є аргументом цільової функції. Для виходу з цієї ситуації оцінка результату проводиться як за однією так і за кількома цільовими функціями, а також вирішується уведенням в процесі розв'язання задачі змінних критеріїв.

Література

1. Тимофієва Н.К. Про розв'язання задач комбінаторної оптимізації в умовах невизначеності. *Вісник Вінницького політехнічного інституту*. 2012. № 6. С. 157–162.
2. Тимофієва Н.К. Теоретико-числові методи розв'язання задач комбінаторної оптимізації: автореф. дис... докт. техн. наук: 01.05.02. Київ, 2007. 32 с.
3. Тимофієва Н.К. Самоналагоджувальні алгоритми знаходження невизначених параметрів у задачах комбінаторної оптимізації. *УСiМ*. 2009. № 4 С.43 – 50.

УЗАГАЛЬНЕННЯ АНОМАЛІЙ В ЗАДАЧАХ УПОРЯДКУВАННЯ

Турчина В. А., Челпанова О. О.

Дніпровський національний університет імені Олеся
Гончара

GENERALIZATION OF ANOMALIES IN ORDERING PROBLEMS

Abstract. One of the promising areas of research of discrete optimization problems is the direction associated with the detection of anomalies in these problems.

Based on Graham's work on the study of anomalies that arise in scheduling problems, namely the cases where intuitive conclusions are not supported, the problem of parallel ordering of the vertices of a graph is considered in this paper. The main question that appears is whether it is possible for anomalies to appear when several conditions are combined, which separately provide for the unambiguous existence of anomalies.

As a result, it becomes necessary to select subclasses of tasks for which the indicated anomalies will not exist and subclasses of tasks for which they will definitely appear.

З появою цікавих результатів Р. Л. Грехема [1], одним із напрямів дослідження задач дискретної оптимізації став напрям, пов'язаний з виявленням у таких задачах так званих аномалій. Під аномаліями розуміються випадки, коли здається інтуїтивно зрозумілі висновки не підтверджуються. Грехем розглянув аномалії, що виникають у задачах теорії розкладів. Він розглянув задачу, коли задана скінчена множина робіт та скінчена множина виконавців. На порядок виконання робіт задані технологічні обмеження та відомий час виконання кожної роботи. Крім того, він розглядав випадки, коли задана бажана послідовність виконання робіт. Необхідно розподілити роботи по виконавцях так, щоб вони були завершені за мінімальний час. При аналізі цієї задачі ним були виявлені наступні, так звані аномалії:

1. Зменшення часу робіт.
2. Послаблення обмежень на порядок робіт.
3. Збільшення кількості виконавців.

4. Зміна порядку виконання робіт.

У даній роботі розглядається відома задача паралельного упорядкування вершин орграфів, яка формулюється наступним чином. Задано зважений орієнтований ациклічний граф $G=\{V,U\}$, ($|V|=n$), що задає технологічні обмеження на порядок виконання робіт. V – множина робіт, U – множина технологічних обмежень, а $(i,j) \in U$ означає, що робота i повинна бути виконана до початку виконання роботи j . Вага вершини i – це час τ_i виконання відповідної роботи з номером i . Необхідно так розмістити вершини орграфа на мінімальну кількість місць в упорядкуванні S , щоб на кожному місці стояло не більше m вершин (m – задана кількість виконавців) і не порушувались технологічні обмеження на порядок виконання робіт.

Для цієї задачі вказані вище аномалії, як показано в [2], мають місце. Виникає питання, чи будуть мати місце аномалії, якщо розглядати випадки:

- одночасного зменшення часу виконання завдань та збільшення кількості ресурсів;
- послаблення технологічних обмежень та збільшення кількості ресурсів;
- одночасного зменшення часу виконання завдань та послаблення технологічних обмежень.

Розглянемо приклад, що відповідає першому випадку.

Приклад 1. Нехай задано зважений орієнтований ациклічний граф $G=\{V,U\}$ (рис. 1), $V=\{1,2,...,9\}$, $m=3$, вага кожної вершини дорівнює τ_i . Кожна вершина на рисунку позначається як « i/τ_i » («номер вершини/вага вершини»).

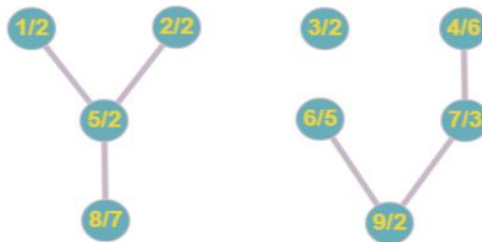


Рис.1. Граф G

По побудованому оптимальному упорядкуванню S^* (табл.1) знаходимо його довжину l , що дорівнює значенню часу виконання всіх робіт виконавцем, який витратив на всі роботи максимальний час.

Таблиця 1.

Упорядкування S^*			
2/2	3/2	6/5	
1/2	5/2	8/7	
4/6		7/3	9/2

$$l(S^*)=11.$$

Тепер одночасно зменшимо час виконання завдань та збільшимо кількість ресурсів:

$$\tau_i' = \tau_i - 1, m' = m + 1.$$

Отримаємо граф G' (рис.2).

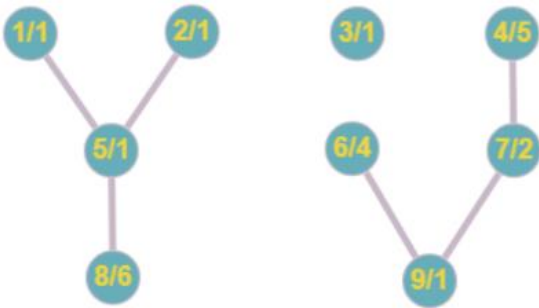


Рис.2. Граф G'

Побудуємо для графа G' оптимальне упорядкування S'^* (табл.2) та знайдемо довжину l' .

Упорядкування S^*

1/1	5/1	8/6		
2/1	6/4			
4/5			7/2	9/1
3/1				

$$l(S^*)=8.$$

Як видно із прикладу, послаблення обмежень дозволяє зменшити загальний час виконання. Зрозуміло, що такі випадки не завжди будуть мати місце. Тому, необхідно виділяти підкласи задач, для яких вказані аномалії не вплинуть на оптимальність розв'язку та підкласи задач, коли вони точно будуть впливати. Крім того, цікавим є питання про можливі аномалії в інших задачах дискретної оптимізації.

Література

1. Graham R. L. Bounds on multiprocessing timing anomalies // SIAM J. Appl. Math., 1969. V. 17. P. 416–429.
2. Турчина В. А. Временные аномалии в задачах составления расписаний / В. А. Турчина, Н. К. Федоренко // Динамические системы. – 2011. – Т. 1 (29), №1. – С. 89-102.

СТАТИСТИЧНІ МЕТОДИ ПРИЙНЯТТЯ РІШЕНЬ І СЕРЕДНІ ВТРАТИ

Черноусова Ж. Т.

*Національний технічний університет України «Київський
політехнічний інститут імені Ігоря Сікорського»*

STATISTICAL DECISION MAKING METHODS AND RISK

Abstract. Strategic action planning depending on the values of a random variable of the experiment result is considered. The problem of the relationship of Bayesian risk and total average losses of mixed actions is investigated. The

conditions under which the total average losses of mixed actions are not less than the Bayesian risk are found.

Розглянемо стратегічне планування дій (правило для прийняття рішення) залежно від випадкового значення спостереження (для кожного результату експерименту). Нехай задані R_{ij} – невід’ємні втрати для кожної комбінації j -ої дії (рішення) та i -ого стану параметра, та відомі ймовірнісні розподіли результатів спостереження в залежності від станів параметра. Тоді для деякої стратегії можна визначити розподіли ймовірностей P_{ij} , $\sum_j P_{ij}=1$, дій для різних станів параметра. Потім розрахувати значення середніх втрат (математичного сподівання) мішаної дії для кожного стану параметра.

Виникає питання, як співвідносяться байесівський ризик (точна нижня грань середніх втрат (ризиків) по всім діям) та загальні середні втрати мішаних дій, якщо відомі ймовірності станів параметру Q_i .

Зауважимо, що в загальному випадку, умовні (залежні від стану параметра) ймовірності вибору відповідних дій не обов’язково однакові.

Дослідимо для наочності випадок трьох дій і трьох станів параметра. Доведення легко розповсюджується на ситуації більшої розмірності.

Загальні середні втрати мішаних дій $Z=Q_1 \cdot (P_{11} \cdot R_{11} + P_{12} \cdot R_{12} + P_{13} \cdot R_{13}) + Q_2 \cdot (P_{21} \cdot R_{21} + P_{22} \cdot R_{22} + P_{23} \cdot R_{23}) + Q_3 \cdot (P_{31} \cdot R_{31} + P_{32} \cdot R_{32} + P_{33} \cdot R_{33}) \geq P_{k1} \cdot (Q_1 \cdot R_{11} + Q_2 \cdot R_{21} + Q_3 \cdot R_{31}) + P_{l2} \cdot (Q_1 \cdot R_{12} + Q_2 \cdot R_{22} + Q_3 \cdot R_{32}) + P_{m3} \cdot (Q_1 \cdot R_{13} + Q_2 \cdot R_{23} + Q_3 \cdot R_{33}) = P_{k1} \cdot \rho_1 + P_{l2} \cdot \rho_2 + P_{m3} \cdot \rho_3$, де P_{k1} – мінімальне значення з ймовірностей 1-ої дії по всім станам параметра, P_{l2} та P_{m3} мають аналогічний зміст для 2-ої та 3-ої дій (k, l, m можуть бути між собою як різними, так і однаковими), ρ_1, ρ_2, ρ_3 – позначення відповідних виразів в останніх дужках – середні ризики 1-ої, 2-ої та 3-ої дій.

Без втрати загальності, нехай номери дій вибрані так, що $\rho_1 \leq \rho_2 \leq \rho_3$.

По-перше, припустимо, що $P_{k1} + P_{l2} + P_{m3} = 1$.

Тоді $P_{k1} \cdot \rho_1 + P_{l2} \cdot \rho_2 + P_{m3} \cdot \rho_3 = P_{k1} \cdot \rho_1 + (1 - P_{k1}) \cdot \rho_2 +$

$$\begin{aligned}
 &+ (Pk1-1) \cdot p2 + Pl2 \cdot p2 + Pm3 \cdot p3 = Pk1 \cdot p1 + (1-Pk1) \cdot p2 + \\
 &+ (Pk1 + Pl2-1) \cdot p2 + Pm3 \cdot p3 = Pk1 \cdot p1 + (1-Pk1) \cdot p2 + (-Pm3) \cdot p2 + \\
 &+ Pm3 \cdot p3 = Pk1 \cdot p1 + (1-Pk1) \cdot p2 + Pm3 \cdot (p3-p2) \geq p1.
 \end{aligned}$$

По-друге, нехай $Pk1+Pl2 \geq 1$.

$$\begin{aligned}
 &\text{Тоді } Pk1 \cdot p1 + Pl2 \cdot p2 + Pm3 \cdot p3 = Pk1 \cdot p1 + (1-Pk1) \cdot p2 + \\
 &+ (Pk1-1) \cdot p2 + Pl2 \cdot p2 + Pm3 \cdot p3 = Pk1 \cdot p1 + (1-Pk1) \cdot p2 + \\
 &+ (Pk1+Pl2-1) \cdot p2 + Pm3 \cdot p3 \geq p1.
 \end{aligned}$$

По-третє, припустимо, що $-Pm3 < Pk1+Pl2-1 < 0$.

Звідси $0 < 1-Pk1-Pl2 < Pm3$.

$$\begin{aligned}
 &\text{Тоді, аналогічно, } Pk1 \cdot p1 + Pl2 \cdot p2 + Pm3 \cdot p3 = Pk1 \cdot p1 + (1- \\
 &Pk1) \cdot p2 + (Pk1-1) \cdot p2 + Pl2 \cdot p2 + Pm3 \cdot p3 = Pk1 \cdot p1 + (1-Pk1) \cdot p2 - \\
 &- (1-Pk1-Pl2) \cdot p2 + Pm3 \cdot p3 \geq p1.
 \end{aligned}$$

Отже, для розглянутих випадків загальні середні втрати Z мішаних дій *не менше* байесівського ризику, за “довільних” ймовірностей станів параметра. Тобто за наведених вище умов, мішані дії не зменшують байесівський ризик, отриманий при чистих діях.

Тепер розглянемо випадок, наведений в [1, стор.16].

$$R11=1, R12=2, R13=3, \quad P11=1/2, P12=1/2, \mathbf{P13=0},$$

середні втрати = 1,5;

$$R21=5, R22=2, R23=3, \quad \mathbf{P21=0}, \quad P22=1/2, P23=1/2,$$

середні втрати = 2,5;

$$R31=7, R32=6, R33=3, \quad \mathbf{P31=0}, \quad \mathbf{P32=0}, \quad P33=1,$$

середні втрати = 3.

Візьмемо $Q1=Q2=Q3=1/3$, тоді

$$\min (p1=13/3; p2=10/3; p3=3) = 3,$$

загальні середні втрати мішаних дій $Z=1/3 \cdot 1,5 + 1/3 \cdot 2,5 + 1/3 \cdot 3 = 7/3 < 3$.

Для наведеного прикладу $Pk1=0, Pl2=0, Pm3=0$ і $\mathbf{Pk1+Pl2+Pm3=0 < 1}$. В цьому випадку загальні середні втрати Z мішаних дій *менше* байесівського ризику.

Якщо

$$R11=1, R12=2, R13=3, \quad P11=1/2, \mathbf{P12=1/3}, \mathbf{P13=1/6},$$

середні втрати = 5/3;

$$R21=5, R22=2, R23=3, \quad P21=1/2, \mathbf{P22=1/3}, \mathbf{P23=1/6},$$

середні втрати = 11/3;

$R_{31}=7, R_{32}=6, R_{33}=3, P_{31}=0, P_{32}=1/3, P_{33}=2/3,$
середні втрати = 4.

У разі $Q_1=Q_2=Q_3=1/3$ загальні середні втрати мішаних дій
 $Z = 1/3 \cdot 5/3 + 1/3 \cdot 11/3 + 1/3 \cdot 4 = 28/9 > 3$.

Для наведеного прикладу $P_{k1}=0, P_{l2}=1/3, P_{m3}=1/6$ і
 $P_{k1}+P_{l2}+P_{m3}=1/2 < 1$. В цьому випадку загальні середні втрати
 Z мішаних дій *більше* байесівського ризику.

Таким чином, при $P_{k1}+P_{l2}+P_{m3} < 1$ загальні середні втрати
 Z мішаних дій можуть бути як *меншими*, так і *не меншими*, ніж
байесівський ризик, отриманий при чистих діях.

За наявності тільки двох дій, міркуючи аналогічно, за умов
 $P_{k1}+P_{l2} \geq 1$ мішані дії не зменшують байесівський ризик,
отриманий при чистих діях.

Якщо $P_{k1}+P_{l2} < 1$, то для наступного прикладу
($Q_1=Q_2=Q_3=1/3$):

$R_{11}=1, R_{12}=2, P_{11}=1/2, P_{12}=1/2,$ середні втрати = $3/2$;

$R_{21}=5, R_{22}=2, P_{21}=1/3, P_{22}=2/3,$ середні втрати = 3;

$R_{31}=7, R_{32}=6, P_{31}=1/4, P_{32}=3/4,$ середні втрати = $25/4$;

$\min (p_1=13/3; p_2=10/3) = 10/3$, загальні середні втрати мішаних
дій $Z=1/3 \cdot 3/2 + 1/3 \cdot 3 + 1/3 \cdot 25/4 = 43/12 > 10/3$,

$P_{k1}=1/4, P_{l2}=1/2$ і **$P_{k1}+P_{l2}=3/4 < 1$** . В цьому випадку загальні
середні втрати мішаних дій Z *більше* байесівського ризику.

При

$R_{11}=1, R_{12}=2, P_{11}=1/2, P_{12}=1/2,$ середні втрати = 1,5;

$R_{21}=5, R_{22}=2, P_{21}=0, P_{22}=1,$ середні втрати = 2;

$R_{31}=7, R_{32}=6, P_{31}=0, P_{32}=1,$ середні втрати = 6;

$Q_1=Q_2=Q_3=1/3,$

$\min (p_1=13/3; p_2=10/3) = 10/3$, загальні середні втрати мішаних
дій

$Z = 1/3 \cdot 1,5 + 1/3 \cdot 2 + 1/3 \cdot 6 = 19/6 < 10/3$,

$P_{k1}=0, P_{l2}=1/2$ і **$P_{k1}+P_{l2}=1/2 < 1$** . В цьому випадку загальні
середні втрати Z мішаних дій *менше* байесівського ризику.

Отже, при $P_{k1}+P_{l2} < 1$, за наявності тільки двох дій, загальні
середні втрати Z мішаних дій можуть бути як *меншими*, так і *не меншими*, ніж байесівський ризик, отриманий при чистих діях.

Література

1. Чернов Г., Мозес Л. Элементарная теория статистических решений. М.: «Советское радио», 1962. – 406 с.
2. Де Гроот М. Оптимальные статистические решения. – М.: «Мир», 1974. – 496с.

ДОСЛІДЖЕННЯ МЕТОДІВ ПОБУДОВИ СУЧАСНИХ НЕЙРОННИХ МЕРЕЖ ДЛЯ ЗАДАЧ АВТОМАТИЗАЦІЇ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ

Шимко С.В., Бащенко Д.В., Мелешко Є.В., Маключенко С.І.,
*Центральноукраїнський національний технічний університет,
м. Кіровоградський*

RESEARCH OF METHODS OF CONSTRUCTION OF MODERN NEURAL NETWORKS FOR THE APPLICATIONS OF IMAGES CLASSIFICATION AUTOMATION

Abstract. In this work, modern methods of building neural networks for image classification, which allow automating such routine operations as image search, analysis of medical images, text recognition, detection of threats on video surveillance cameras, etc were researched.

В наш час класифікація зображень – одна з основних і широко використовуваних задач комп'ютерного зору.

Класифікація зображень дозволяє віднести досліджуване зображення до певної категорії об'єктів, що може бути застосоване у наступних застосуваннях:

- Автоматичний контроль, наприклад, у виробничих цілях;
- Надання допомоги людям в задачах ідентифікації, наприклад, система ідентифікації об'єктів;
- Процеси управління, наприклад, промисловими роботами;
- Виявлення подій, наприклад, для візуального спостереження або підрахунку людей;

- Задачі взаємодії, наприклад, для пристроїв взаємодії людини з комп'ютером;
- Моделювання об'єктів або середовищ, наприклад, аналіз медичних зображень або топографічне моделювання;
- Навігація, наприклад, автономним транспортним засобом або мобільним роботом;
- Організація інформації, наприклад, для індексування баз даних зображень і послідовностей зображень.

Для реалізації нейронних мереж уже існує багато бібліотек програмування. Розглянемо найвідоміші з них.

TensorFlow – відкрита програмна бібліотека для машинного навчання, розроблена компанією Google [1]. TensorFlow було початково розроблено командою Google Brain для внутрішнього використання в Google, поки її не було випущено під відкритою ліцензією Apache 2.0 9.11.2015 року. TensorFlow може працювати на декількох центральних процесорах та графічних процесорах (включно з додатковими розширеннями CUDA).

TensorFlow може застосовуватися для розробки програмного забезпечення автоматизованого опису зображень, такого як DeepDream.

Переваги TensorFlow: простий вбудований високорівневий API, багатофункціональність, гарна підтримка під мобільні пристрої, детальна документація.

Недоліки TensorFlow: складне налагодження та труднощі з внесенням змін (пов'язані з використанням статичних обчислювальних графів). Також до недоліків слід віднести відсутність підтримки заздалегідь навчених моделей.

FANN (Fast Artificial Neural Network) – безкоштовна бібліотека швидкої штучної нейронної мережі з відкритим вихідним кодом, яка реалізує багатошарові штучні нейронні мережі на мові C [2]. Включає в себе структуру для простої обробки наборів даних для навчання.

Переваги бібліотеки для роботи з нейронними мережами FANN: проста у використанні, універсальна, хороша документація з прикладами, швидка, доступні прив'язки до більш ніж 20 мов програмування, для бібліотеки також доступні декілька графічних користувацьких інтерфейсів.

До недоліків слід віднести невелику кількість різних можливих архітектур нейронних мереж, доступні тільки 1) одношарові мережі прямого поширення; 2) багатошарові мережі прямого поширення; 3) мережі з радіальної базисної функцією. Інші ж архітектури недоступні.

Одною з найважливіших задач при побудові нейронних мереж є підбір архітектури мережі. Від правильного підбору архітектури залежить спроможність нейронної мережі виконувати поставлені до неї завдання, а також точність її роботи.

Найпершою архітектурою нейронної мережі для класифікації зображень був перцептрон Розенблата [2]. Така нейронна мережа мала лише один прихований прошарок нейронів. Сучасні нейронні мережі для розпізнавання зображень містять багато прихованих прошарків нейронів, системи побудовані за допомогою них називаються системами з глибоким навчанням [4]. Такі багатошарові мережі найчастіше використовують алгоритм навчання BackPropagation – зворотного поширення похибки.

В наш час архітектура багатошарової нейронної мережі зі зворотним поширенням похибки найбільш популярна для розпізнавання зображень, вона ефективна та легка у навчанні.

Типова нейронна мережа зворотного поширення похибки має вхідний прошарок, вихідний прошарок та декілька прихованих прошарків між ними. Теоретично обмежень відносно числа прихованих прошарків не існує, їх кількість залежить від складності задачі, що треба вирішити, та кількості вхідних даних. Визначення потрібного числа прихованих прошарків і числа нейронів у них є важливою задачею при моделюванні мережі. Навчання такої нейронної мережі відбувається з використанням алгоритму градієнтного спуску, який використовується з метою мінімізації помилки роботи багатошарової мережі та отримання бажаного виходу. Метод навчання полягає в поширенні сигналів помилки від виходів мережі до її входів, в напрямку, зворотному прямому поширенню сигналів у звичайному режимі роботи.

Література

1. TensorFlow tutorials – 2021. – [Electronic resource] – Access mode: <https://www.tensorflow.org/tutorials/>
2. Fast Artificial Neural Network Library – 2021. – [Electronic resource] – Access mode: <http://leenissen.dk/fann/wp/>
3. Глибовець М.М., Олецкий О.В. Штучний інтелект: Підручник. – К.: Вид. дім "КМ Академія". – 2002. – 366 с.
4. Николенко С., Кадури́н А., Архангельская Е. Глубокое обучение. – СПб.: Питер. – 2018. – 480 с.

АНАЛІЗ МЕРЕЖЕВОГО ГРАФІКУ ЗА ДОПОМОГОЮ ЕНТРОПІЇ

Яценко А.К., Дубровін В.І, Твердохліб Ю.В.

Національний університет «Запорізька політехніка»

Freshcode

Abstract: Entropy-based anomaly detection is being studied to improve the traditional approaches to network flow analysis based on volume and rules. This work discusses methods using the Shannon, Renyi and Tsallis entropies.

Keywords: anomaly detection; entropy; netflow; network traffic measurement

1 Вступ

Одним із найбільш ефективних індикаторів аномального поводження мережі є аналіз трафіку в реальному масштабі часу. Виявлення аномалії – це ідентифікація спостережень, які не відповідають очікуваній поведінці. З цією метою використовуються різноманітні параметри мережевого трафіку, такі, як адреси та порти джерела та призначення, кількість з'єднань та інше [1].

Основна мета цієї роботи – довести, що підхід, заснований на ентропії, підходить для виявлення сучасних шкідливих програм, схожих на ботнет, на основі аномальних мережевих шаблонів.

2 Інформаційна ентропія

Інформаційна ентропія – це міра невизначеності, яка пов'язана з випадковою величиною. Ентропія характеризує

ймовірність P , з якою встановлюється той чи інший стан, є мірою хаотичності або незворотності.

Для розподілу ймовірностей $p(X = x_i)$ дискретної випадкової величини X ентропія Шеннона [2] визначається:

$$H_s(X) = \sum_{i=1}^n p(x_i) \log_a \frac{1}{p(x_i)} \quad (1)$$

де X – ознака, яка може приймати значення $\{x_1, \dots, x_n\}$, $p(x_i)$ – функція ймовірності результату x_i , n – кількість можливих станів. З метою виявлення аномалії зазвичай використовуються вибіркові ймовірності, оцінені за кількістю випадків x_i у часовому вікні t . Значення ентропії залежить від випадковості (вона досягає максимуму, коли ймовірність $p(x_i)$ однакова для всіх x_i), але також і від значення n . Для того, щоб виміряти лише випадковість, слід застосовувати нормалізовані форми.

Ентропія Шеннона передбачає компроміс між внеском від основної маси розподілу та хвоста. Для контролю над цим компромісом були запропоновані два параметризовані узагальнення ентропії Шеннона, відповідно Реньї [3] і Цалліса [4]. Якщо параметр, позначений як α – позитивний, він демонструє основну масу (концентрацію подій, що часто трапляються), якщо значення від'ємне – він відноситься до хвоста (дисперсія, спричинена рідкісними подіями).

Ентропія Реньї:
$$H_{R\alpha}(X) = \frac{1}{1-\alpha} \log_a \left(\sum_{i=1}^n p(x_i)^\alpha \right)$$

(2)

Ентропія Цалліса:
$$H_{T\alpha}(X) = \frac{1}{1-\alpha} \left(\sum_{i=1}^n p(x_i)^\alpha - 1 \right) \quad (3)$$

Обидві параметризовані (Реньї та Цалліса) ентропії:

- стосуються концентрації для $\alpha > 1$ та дисперсії для $\alpha < 1$,
- сходяться до ентропії Шеннона для $\alpha \rightarrow 1$,
- відповідають потужності X при $\alpha = 0$.

Параметр α – сприйнятливність функціоналу до відхилення стану системи від рівноважного. Чим більше α , тим швидше зменшується ентропія при відхиленні системи від стану рівноваги.

3 Аналіз трафіку за допомогою ентропії

Для того, щоб перевірити, чи підхід, заснований на ентропії, підходить для виявлення сучасного шкідливого програмного забезпечення, подібного до ботнету, на основі аномальних шаблонів мережі, був запропонований детектор аномалії мережі на основі ентропії [1]. Принцип роботи програмного комплексу представлений на рис 1.

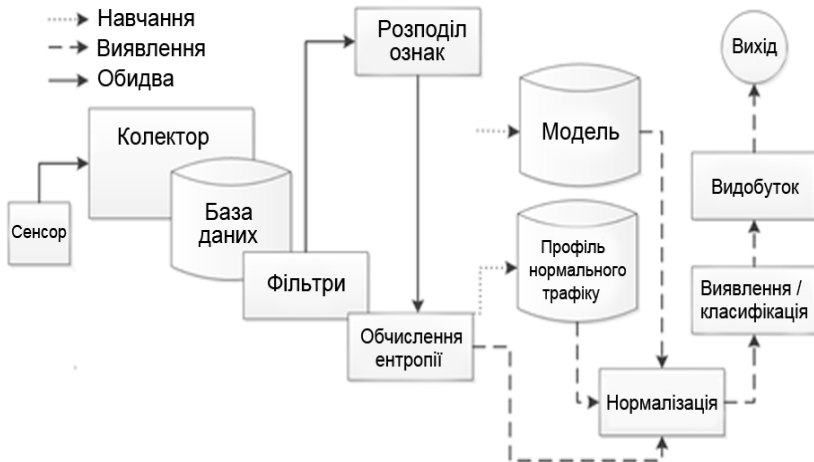


Рис 1. Структурна схема програмного комплексу аналізу трафіку за допомогою ентропії [1].

Програмний комплекс аналізує мережеві дані, зафіксовані сенсорами NetFlow. Припускаються типові сенсори, такі як маршрутизатори або спеціальні сенсори, наприклад, Softflowd [5], підключений до TAP або порти SPAN на комутаторах. Потoki аналізуються протягом фіксованих інтервалів часу (кожні 5 хв). Зібрані потоки реєструються в БД, а потім аналізуються. Для обмеження області пошуку аномалій передбачені фільтри за напрямком, протоколом та підмережею. Далі, у залежності від режиму, розраховуються ентропії Цалліса чи Реньї позитивних та негативних значень α для розподілу характеристик трафіку. Для методу Шеннона використовується внутрішня ентропія Реньї з значенням $\alpha = 1$.

На фазі виявлення спостережувана ентропія порівнюється зі значеннями \min і \max , що зберігаються у профілі, згідно з наступним правилом визначення порогу аномалії:

$$r_{\alpha}(x_i) = \frac{H_{\alpha}(x_i) - k \cdot \min_{\alpha}}{k \cdot (\max_{\alpha} - \min_{\alpha})}, k \in \langle 1..2 \rangle \quad (4)$$

Значення $r(x_i) < 0$ або $r(x_i) > 1$ вказують на аномальну концентрацію або дисперсію відповідно. Ця аномальна дисперсія або концентрація для різних розподілів ознак характерна для аномалій.

Виявлення базується на відносному значенні ентропії щодо відстані між \min та \max . Коефіцієнт k у формулі визначає запас для мінімальної та максимальної меж і може бути використаний для цілей налаштування. Високе значення k , наприклад $k = 2$, обмежує кількість помилкових спрацьовувань, тоді як низьке значення ($k = 1$) збільшує частоту виявлення.

4 Висновки

Метод ентропії є одним з найбільш перспективних методів раннього виявлення мережових атак, що дозволить технічному персоналу можливість вжити заходів щодо мінімізації збитку від шкідливого трафіку.

У даній роботі не розглядалися змішані види атак, це один з перспективних напрямків досліджень.

Література

1. P. Bereziński, B. Jasiul and M. Szpyrka, "An entropy-based network anomaly detection method", *Entropy*, 2015, 17(4), p.p.2367-2408.
2. Shannon, C. A Mathematical Theory of Communication. *Bell Syst. Tech. J.* 1948, 27, 379–423.
3. Renyi, A. *Probability Theory; Enlarged version of Wahrscheinlichkeitsrechnung, Valoszinusegyszamitas and Calcul des probabilites*. English translation by Laszlo Vekerdi; North-Holland: Amsterdam, The Netherlands, 1970, p. 573.
4. Tsallis, C. Possible generalization of Boltzmann-Gibbs statistics. *J. Stat. Phys.* 1988, 52, p.p. 479–487.

5. Damien Miller, Hitoshi Irino — Softflowd — Traffic flow monitoring. Available online:
<https://www.freebsd.org/cgi/man.cgi?query=softflowd&sektion=8&manpath=freebsd-release-ports>

РЕЗОЛЮЦІЯ
XXIII Міжнародного науково-практичного семінару
«Комбінаторні конфігурації та їхні застосування»
13 травня 2021 р.

Учасники XXIII Міжнародного науково-практичного семінару «Комбінаторні конфігурації та їхні застосування», організованого за підтримки Міністерства освіти і науки України, Національним університетом «Запорізька політехніка» (кафедрою системного аналізу та обчислювальної математики) та Льотною академією Національного авіаційного університету (м.Кропивницький), представили 40 навчальних закладів та організацій України та іноземні. В семінарі брали участь представники вищих навчальних закладів та організацій:

- Національний університет «Запорізька політехніка»
- Льотна академія Національного авіаційного університету
- Інститут кібернетики імені В. М. Глушкова НАН України
- Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»
- Київський національний університет імені Тараса Шевченка
- Харківський національний університет радіоелектроніки
- Національний авіаційний університет
- Запорізький національний університет
- Центральноукраїнський національний технічний університет
- Харківський національний університет Повітряних Сил ім. І. Кожедуба
- Харківський національний університет міського господарства імені О.Бекетова
- Національний технічний університет «Дніпровська політехніка»
- Військовий інститут телекомунікацій та інформатизації імені Героїв Крут

- Міжнародний науково-навчальний центр інформаційних технологій та систем НАН та МОН України
- Національний університет «Львівська політехніка»
- ДВНЗ «Донбаський державний педагогічний університет»
- Запорізький обласний інститут післядипломної педагогічної освіти
- ДВНЗ «Ужгородський національний університет»
- Донецький національний медичний університет
- Інститут фізики НАН України
- Луцький національний технічний університет
- Дніпровський національний університет імені Олеся Гончара
- Харківський національний університет імені В. Н. Каразіна
- Національний аерокосмічний університет імені М. Є. Жуковського "ХАІ"
- Національний університет водного господарства та природокористування
- Центральноукраїнський державний педагогічний університет імені В. Винниченка
- Центральноукраїнський національний технічний університет
- Відкритий міжнародний університет розвитку людини «Україна»
- Громадська організація «Системні дослідження»
- Українське фізичне товариство
- Інститут проблем машинобудування ім. А.М. Підгорного НАН України
- Інститут олійних культур Національної академії аграрних наук України
- Logic7, м.Запоріжжя
- Freshcode, м.Запоріжжя
- Warwick Business School, Great Britain

- Інститут інформаційних теорій і застосувань, Софія, Болгарія
- Самаркандський філіал Ташкентського університету інформаційних технологій
- Білоруський державний університет
- Вірменський державний педагогічний університет імені Х. Абовяна
- Среванський державний університет

Учасники семінару розглянули найбільш актуальні питання, пов'язані із комбінаторними конфігураціями та їхніми застосуваннями до вирішення сучасних проблем моделювання та прийняття рішень в управлінні складними системами та процесами за такими напрямками:

1. Комбінаторні розміщення, графи і задачі оптимізації на них, автомати, скінченні алгебри, математична логіка і теорія множин, теорія ігор і теорія прийняття рішень та їхні застосування в різних галузях;

2. Реалізація математичних моделей на дискретних та неперервних множинах (автоматизовані системи керування, системи автоматизованого проектування, системи підтримки прийняття рішень);

3. Програмне забезпечення синтезу та аналізу комбінаторних конфігурацій;

4. Розпізнавання образів, нейромережі та методи машинного навчання;

5. Фрагментарні структури, алгоритми та метаевристики;

6. Топологія та історія розвитку української науки;

7. Кодування, інформаційна безпека та захист інформації.

Проблематика конференції привернула до себе увагу фахівців із різних галузей знань. У науковому заході взяли участь 109 учасників, серед яких:

30 докторів наук (фізико-математичних – 14, технічних – 12, педагогічних – 3, економічних – 1);

47 кандидатів наук (фізико-математичних – 26, технічних – 17, педагогічних – 1; економічних – 1; історичних – 1; медичних – 1);

8 викладачів та науковців без наукового ступеня;

15 аспірантів; 9 студентів та магістрантів.

Серед учасників конференції були присутні висококваліфіковані науковці, молоді вчені, аспіранти та магістранти з вищих навчальних закладів міст України: Запоріжжя, Кропивницький, Дніпро, Київ, Львів, Харків, Ужгород, Рівне, Слов'янськ. Також приймали участь іноземні науковці з країн: Болгарія, Білорусь, Вірменія, Великобританія, Узбекистан.

Учасники конференції обговорили такі проблемні питання:

- моделювання задач вибору на основі нечітких відношень;
- фрагментарний підхід до вирішення багатокритеріальних задач;
- аналіз методів розмітки графів;
- методи імітаційного моделювання та проблеми розробки рекомендаційних систем;
- вирішення задач різних класів обчислювального інтелекту;
- вирішення задач пошуку ідентичних структур;
- задачі перерахування графів та топологій певного виду;
- застосування нейромереж до обробки сигналів різного роду;
- формалізація процесів прийняття рішень в умовах невизначеності;
- методика викладання теорії інформації та комбінаторного аналізу у вищих навчальних закладах;
- історія розвитку української науки та української наукової мови.

Усвідомлюючи роль і значення розвитку фундаментальних і прикладних досліджень в галузях науки, де

широко використовуються комбінаторні конфігурації, учасники конференції вважають за необхідне:

1. Продовжити наукові дослідження в галузі комбінаторних конфігурацій та їхніх застосувань. Організовувати майбутні семінари, поєднуючи онлайн форму роботи разом з традиційною офлайн (за можливістю).

2. Поширювати наукові зв'язки з вітчизняними та закордонними провідними закладами для організації і реалізації спільних наукових та освітніх проєктів. Рекомендувати для проведення наступних семінарів підготовку матеріалів на англійській мові.

3. Розширити географію проведення семінару в наступні роки, змінюючи місце його проведення (Кропивницький, Запоріжжя, інші пропозиції за рішенням оргкомітету).

4. Розмістити інформацію про проведений семінар на сайті Міністерства освіти і науки України та опублікувати резолюцію конференції і збірник матеріалів на сайтах Національного університету «Запорізька політехніка», Льотної академії Національного авіаційного університету (м.Кропивницький), Центральноукраїнського національного технічного університету.

Співголова оргкомітету
зав. каф. СА та ОМ

Г.В.Корніч

КОМБІНАТОРНІ КОНФІГУРАЦІЇ ТА ЇХНІ ЗАСТОСУВАННЯ

Матеріали XXIII Міжнародного науково-практичного семінару імені А.Я. Петренюка

13-15 травня 2021 року

Редакційна колегія не несе відповідальності за матеріали,
опубліковані в збірнику. Усі вони подані в авторській редакції та
виражають персональну позицію учасників семінару

Підписано до друку 07.06.2021 р.
Формат 60х84/16. Папір офсетний.
Гарнітура Times New Roman. Друк ризограф.
Ум. друк. арк. 13,0
Наклад 300 прим. Зам. №0501

Приватне підприємство «Ексклюзив-Систем»
Свідоцтво держ. реєстру ДК№4470 від 17.01.2013р.
25006, м. Кропивницький, вул. Шевченка, 25
тел. 050-92-20913