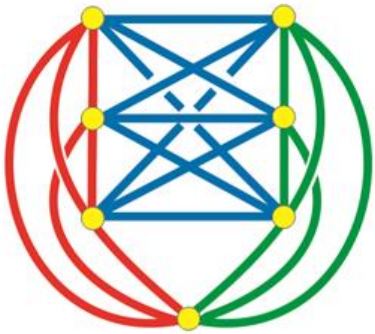
Міністерство освіти і науки України
Інститут кібернетики ім. В.М.Глушкова НАН України
Національний університет «Запорізька політехніка»
Льотна академія Національного авіаційного університету
Запорізький національний університет
Центральноукраїнський національний технічний університет
Центральноукраїнський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

Матеріали XXV Міжнародного
науково-практичного семінару імені А.Я. Петренюка

КОМБІНАТОРНІ КОНФІГУРАЦІЇ ТА ЇХНІ ЗАСТОСУВАННЯ

14-16 ***червня*** **2023** ***року***



Запоріжжя - Кропивницький
2023

УДК 519
К63

Рекомендовано до видання Вченою радою
Національного університету «Запорізька політехніка»
(Протокол № 11 від 26.06.2023 р.)

Редакційна колегія:
Георгій ДОНЕЦЬ, д. ф.-м. н., професор;
Леонід ГУЛЯНИЦЬКИЙ., член-кор. НАН України, д.т.н.;
Анна БАКУРОВА, д.е.н., професор;
Ігор КОЗІН, д.ф.-м.н., професор;
Марина СЕМЕНЮТА, к.ф.-м.н., доцент;
Василь ШЕНДЕРОВСЬКИЙ, д.ф.-м.н., професор.

К63 **Комбінаторні конфігурації та їхні застосування**: Матеріали XXV Міжнародного науково-практичного семінару імені А. Я. Петренюка, (Запоріжжя – Кропивницький, 14-16 червня 2023 року) [Електронний ресурс] / за ред. Г.П. Донця. Електрон. дані. – Запоріжжя: НУ «Запорізька політехніка», 2023. – 247 с. – 1 електрон. опт. диск (DVD-ROM); 12 см. – Назва з тит. екрана.

ISBN 978-617-529-420-8

У збірнику подано матеріали XXV Міжнародного науково-практичного семінару «Комбінаторні конфігурації та їхні застосування» імені А.Я. Петренюка, який відбувся 14-16 червня 2023 року. Коло обговорених проблем: комбінаторні розміщення, графи і задачі оптимізації та їхні застосування в різних галузях; програмне забезпечення синтезу та аналізу комбінаторних конфігурацій; методи машинного навчання; фрагментарні структури, топологія та історія розвитку української науки. Збірник складається з двох частин, в першу частину увійшли матеріали попереднього семінару. Для науковців, викладачів вищих навчальних закладів, а також аспірантів, студентів і фахівців із застосування комбінаторних конфігурацій.

УДК 519

ISBN 978-617-529-420-8

© НУ «Запорізька політехніка», 2023

Програмний комітет конференції

Георгій ДОНЕЦЬ,	д.ф.-м.н., професор, завідувач відділом економічної кібернетики Інституту кібернетики НАН України – голова програмного комітету,.
БАКУРОВА А.В.	д.е.н., професор, професор кафедри «Системного аналізу та обчислювальної математики» НУ «Запорізька політехніка»; співголова програмного комітету;
Марина СЕМЕНЮТА,	к.ф.-м.н., доцент, доцент кафедри фізико-математичних дисциплін та застосування інформаційних технологій в авіаційних системах Льотної академії НАУ;
Kamil AIDAZADE;	doctor of Phys. and Mathem. Sciences, Professor, Head Laboratory at Institute of Control Systems, Azerbaijan National Academy of Sciences, Corresponding member of the Azerbaijan National Academy of Sciences;
Vladimir DEINEKO;	PhD, DSc. Associate Professor (Reader), Warwick Business School, UK;
Olexandr KONOVALOV,	Dr., Lecturer, Software Sustainability Institute Fellow, School of Computer Science, University of St Andrews, UK;
Krassimir MARKOV,	University of Telecommunications and Posts, Sofia, Bulgaria;
Улугбек НАРЗУЛЛАЕВ,	к.ф.-м.н., доцент, декан факультету телекомунікаційних технологій і професійної освіти Самаркандського філіалу Ташкентського університету інформаційних технологій;
Леонід ГУЛЯНИЦЬКИЙ,	член-кор. НАН України, д.т.н., старший науковий співробітник, завідувач відділом методів комбінаторної оптимізації та інтелектуальних ІТ Інституту кібернетики ім. В.М. Глушкова НАН України;

Ігор КОЗІН,	д.ф.-м.н., професор, професор кафедри економічної кібернетики Запорізького національного університету;
Сергій КРИВИЙ,	д.ф.-м.н., професор, професор кафедри інтелектуальних програмних систем Київського національного університету ім. Т. Шевченка;
Микола МАЛЯР,	д.т.н., професор, професор кафедри кібернетики і прикладної математики Ужгородського національного університету;
Єлизавета МЕЛЕШКО,	д.т.н., професор, доцент кафедри кібербезпеки та програмного забезпечення Центральноукраїнського національного технічного університету;
Марина НОВОЖИЛОВА,	д.ф.-м.н, професор, завідувач кафедри комп'ютерних наук та інформаційних технологій Харківського національного університету міського господарства імені О.М. Бекетова,
Віталій ПЕРЕПЕЛИЦЯ,	д.ф.-м.н., професор, професор кафедри економічної кібернетики Запорізького національного університету;
Тетяна РОМАНОВА,	д.т.н., професор, провідний науковий співробітник відділу математичного моделювання й оптимального проектування Інституту проблем машинобудування ім. А.М. Підгорного НАН України;
Олексій СМІРНОВ,	д.т.н., професор, завідувач кафедри кібербезпеки та програмного забезпечення Центральноукраїнського національного технічного університету;
Петро СТЕЦЮК,	д.ф.-м.н., професор, завідувач відділом методів негладкої оптимізації Інституту кібернетики ім. В.М. Глушкова НАН України;
Надія ТИМОФІЄВА,	д.т.н., старший науковий співробітник відділу комплексних досліджень

	інформаційних технологій Міжнародного науково-навчального центру інформаційних технологій та систем НАН та МОН України;
Валентина ТУРЧИНА,	к.ф.-м.н., доцент, завідувач кафедри ислювальної математики та ематичної кібернетики Дніпровського іонального університету ім. О.Гончара;
Світлана УС,	к.ф.-м.н, доцент, професор кафедри «Системного аналізу і управління» Національного технічного університету «Дніпровська політехніка»;
Василь ШЕНДЕРОВСЬКИЙ,	д.ф.-м.н., професор, віце-президент Українського фізичного товариства;
Тетяна ШМЕЛЬОВА,	д.т.н., професор, професор кафедри аеронавігаційних систем Національного авіаційного університету;
Сергій ЯКОВЛЄВ,	заслужений діяч науки і техніки України, д.ф.-м.н., професор, професор Інституту інформаційних технологій Лодзьського політехнічного університету.

Організаційний комітет:

Микола КАСЬЯН,	к.т.н., доцент, декан факультету комп'ютерних наук і технологій НУ «Запорізька політехніка»; голова організаційного комітету;
Людмила ДЖУМА,	к.т.н., доцент, доцент кафедри фізико-математичних дисциплін та застосування інформаційних технологій в авіаційних системах Льотної академії НАУ;
Олег ДМІТРІЄВ,	д.т.н., професор, декан факультету льотної експлуатації Льотної академії НАУ;
Галина КОЗІНА,	к.ф.-м.н., доцент, доцент кафедри «Захист інформації» НУ «Запорізька політехніка»;

Григорій КОРНІЧ, д.ф.-м.н., професор, професор кафедри «Системний аналіз та обчислювальна математика НУ «Запорізька політехніка»;
Володимир ПЕТРЕНЮК, к.ф.-м.н., доцент, доцент кафедри кібербезпеки та програмного забезпечення Центральноукраїнського національного технічного університету;
Дмитро ПЕТРЕНЮК, к.ф.-м.н., молодший науковий співробітник відділу економічної кібернетики Інституту кібернетики НАН України;
Антон РЯБЕНКО, к.ф.-м.н., доцент, доцент кафедри «Системний аналіз та обчислювальна математика НУ «Запорізька політехніка»;
Катерина СУРКОВА, к.пед.н., доцент, завідувач кафедри фізико-математичних дисциплін та застосування інформаційних технологій в авіаційних системах Льотної академії НАУ;
Еліна ТЕРЕЩЕНКО, к.ф.-м.н., доцент, в.о. завідувача кафедри «Системний аналіз та обчислювальна математика НУ «Запорізька політехніка»;
Дмитро ШИРОКОРАД, к.ф.-м.н., доцент, ст викладач кафедри «Системний аналіз та обчислювальна математика НУ «Запорізька політехніка»;
Громадська організація «Системні дослідження

ВІТАЛЬНІ СЛОВА УЧАСНИКАМ СЕМІНАРУ

Привітання від НУ «Запорізька політехніка»

Шановні колеги!

Щиро вітаю вас на двадцять п'ятому міжнародному науково-практичному семінарі "Комбінаторні конфігурації та їхні застосування" імені А. Я. Петренюка. Отже, це вже двадцять п'ятий, ювілейний семінар, що сьогодні проводиться в таких неспокійних умовах.

Дозвольте привітати тих, хто з семінаром з самого його заснування і тих, хто приймає участь вперше. Сьогодні наш факультет КНТ, зокрема кафедра Системного аналізу та обчислювальної математики, разом з шановними колегами з Кропивницького, ГО «Системні дослідження» забезпечують платформу для співпраці між різними поколіннями науковців, незважаючи на пандемію і війну. Можливості сучасних інформаційних технологій дозволяють організовувати такі зустрічі незважаючи на географічні відстані між учасниками та конфліктне середовище.

Від свого імені, від імені колег, організаторів з Національного університету «Запорізька політехніка» хочу сказати, що ми щасливі, що маємо можливість підтримати у цей складний час проведення 25 семінару, що його тематика є цікавою для науковців і розвитку комп'ютерних наук і технологій. Це підтверджує географія і поважний склад учасників. Бажаю всім учасникам семінару плідної роботи, вдалих виступів, нових друзів.

І бажаю всім нам найшвидшої Перемоги!

Микола КАСЬЯН,
декан факультету комп'ютерних наук і технологій НУ «Запорізька політехніка», к.тех.н., доцент

Привітання від Льотної академії НАУ

Мені приємно вітати Вас з відкриттям семінару. Вітаю вас, шановні колеги: учасники конференції, редакційна колегія, оргкомітет і всі решта, хто сьогодні з нами від науково-педагогічного складу Льотної академії НАУ та її керівництва!

Ідея заснування семінару належала доктору фізико-математичних наук, професору, талановитому математику в області теорії скінчених інцидентних структур Анатолію Яковичу Петренюку, яку підтримали науковці Інституту кібернетики НАНУ (професор Г. П. Донець) і Кіровоградської льотної академії національного авіаційного університету (професор С. М. Неділько). Сподіваюся, що є ще багато учасників першого семінару, який відбувся у приміщенні Державної льотної академії України, за ініціативи науковців 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 висвітлити результати окремих вчених і наукових груп, які працюють в галузі дискретної математики та її застосувань.

Дякую Вам за підтримку у складний час для всіх нас, це також надихає нас на те, щоб організовувати семінар, щоб він продовжував своє життя. Бажаю приємного та результативного спілкування, натхнення творчих успіхів, здоров'я, миру.

Марина СЕМЕНЮТА,
завідувач кафедри Льотної академії НАУ, к.ф.-м.н., доц.

Привітання від Інституту кібернетики імені В.М. Глушкова НАН України

Шановні друзі!

Маю честь і радість вітати всіх учасників ювілейного XXV Міжнародного науково-практичного семінару імені А.Я. Петренюка «Комбінаторні конфігурації та їхні застосування», який зробив вагомий внесок в розвиток математики, а саме напряму комбінаторні конфігурації. Також просили передати вітання учасникам семінару академік НАН України, член-кореспондент НАН України, професор, доктор ф.-м.н., директор інституту кібернетики їм. В,М, Глушкова НАН України Сергієнко І.В. та академік НАН України, член-кореспондент НАН України, професор, доктор ф.-м.н., заступник директора інституту кібернетики їм. В.М. Глушкова НАН України Хіміч О.М.

Тематика семінару дуже важлива, тому що застосування комбінаторних конфігурацій розширюються з кожним роком, а для цього потрібні глибокі теоретичні засади. К.Берж майже 60 років тому розробив методологію, формалізацію комбінаторних конфігурацій. І ця методологія дозволила розширити класи задач, в яких фігурують комбінаторні конфігурації до задач оптимізації на нескінчених просторах, на розмитих просторах. Деякі задачі з цієї тематики розглядалися на XX семінарі. Але тема досліджень не вичерпана, наприклад задачі маршрутизації транспортних засобів зараз трансформувалися в задачі планування місій за участю БПЛА і це в науковому плані ставить задачі розвитку і дослідження комбінаторних конфігурацій вищих порядків. Наш семінар – це один із способів підняти та розвивати наукові дослідження в складних умовах для України. Сподіваюся, що участь в цьому заході буде не тільки корисною, але і приємною. 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.

Бережіть себе і своїх близьких!

Леонід ГУЛЯНИЦЬКИЙ.,
член-кор. НАН України, д.т.н., старший науковий співробітник, завідувач відділом методів комбінаторної оптимізації та інтелектуальних ІТ Інституту кібернетики ім. В.М. Глушкова НАН України

ДОПОВІДІ

ЧАСТИНА І

BOUNDARIES OF THE ASSORTATIVITY INDEX FOR SCALE-FREE NETWORKS

Shergin V.L.[1], Chala L.E.[1], Udovenko S.G.[2]
[1]Artificial Intelligence dept. NURE, Kharkov, Ukraine,
[2]Informatics and Computer Technics dept. KhNUE, Kharkov, Ukraine

Abstract — Assortativity coefficient is a measure of the tendency of nodes in networks to connect with similar in some way, or dissimilar ones. The assortativity coefficients for most of the real-world networks are close to zero and far from the theoretical bounds ±1. Thus, one might think that this property is not very characteristic of complex networks. But the point is, that assortativity properties should not be studied apart of the distribution of nodes by degrees. Most complex networks are scale-free, so the nodes follow an asymptotically power-law distribution. Fixing the distribution of the degree of nodes forms an essential constraint both on the network structure and on the boundaries of the assortativity coefficient. The problem of estimating the bounds of assortativity coefficients for scale-free networks were solved. It was found, that these boundaries are narrowing with increasing the network size.

Keywords— Assortativity Boundaries, Scale-Free Networks, Power Distribution, Barabasi-Albert Model.

It is widely known, that most of the complex real-world networks are scale-free (SF), i.e. the degree of their nodes is distributed by some power law family of distribution (such as Yule-Simon, Benford, zeta etc.), at least asymptotically [1]-[2].

Despite the importance of the nodes degree distribution, it is not an exhaustive characteristic of networks. While degree distribution is a first order graph metric, the *assortativity* [3], that is, the tendency of nodes to link with similar or dissimilar ones, is a second order one. Assortativity is also an important property of networks. It is known [1], [3]-[4], that social networks are positively assortative, while biological, technical and cognitive ones are disassortative.

Despite the large number of studies of scaling and assortativity, these characteristics are most often studied separately that is considered as an essential disadvantage of the current state of the complex networks theory. While the structure of SF-networks with extreme assortativity is widely known [6]-[8], the bounds of assortativity coefficient for the SF-networks with known power exponent are unknown. Estimating this bounds forms the goal of current research.

An *assortativity coefficient* is defined as correlation coefficient of the nodes by their degrees [1] and in general case is in the range $-1 \le r \le 1$. More convenient form of this coefficient is given in [4]:

$$r = \frac{N_3 - S_2^2 / S_1}{S_3 - S_2^2 / S_1}, \tag{1}$$

where

$$S_k = \sum_{i=1}^{n} d_i^k, \quad N_3 = d^T A d = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} d_i d_j, \tag{2}$$

A is adjacency matrix of the network, d_i is node degrees.

Without loss of generality, we can assume that the nodes are ordered by a decreasing degree, so index number i of the node d_i is its range.

As it follows from (1)-(2), network assortativity coefficient (r) depends on degree distribution d_i, so networks mixing cannot be studied apart from scaling. If the distribution of nodes by degree follows to a power law, then the range distribution of nodes asymptotically also follows to a power-family distribution:

$$d_i \approx c \cdot i^{-\beta}, \quad \frac{1}{2} \le \beta < 1. \tag{3}$$

Let's fix the network size and the values of nodes degree. Then, the structure of SF-network with extreme assortativity can be found by greedy algorithms of simple structure [6]-[8]. Links are distributed starting from the richest node. Network is extremely disassortative if on each step current node links with as poorest one as possible. Likewise, while creating an extremely assortative network, current node links with as richest one as possible.

The structure of SF-network with extreme assortativity is shown on Fig.1. As one can see, an extremely disassortative network is tends to be *bipartile* [8],

while in opposite case (extreme assortative mixing) network is close to be a set of almost isolated clusters, all but the largest of which are almost regular.

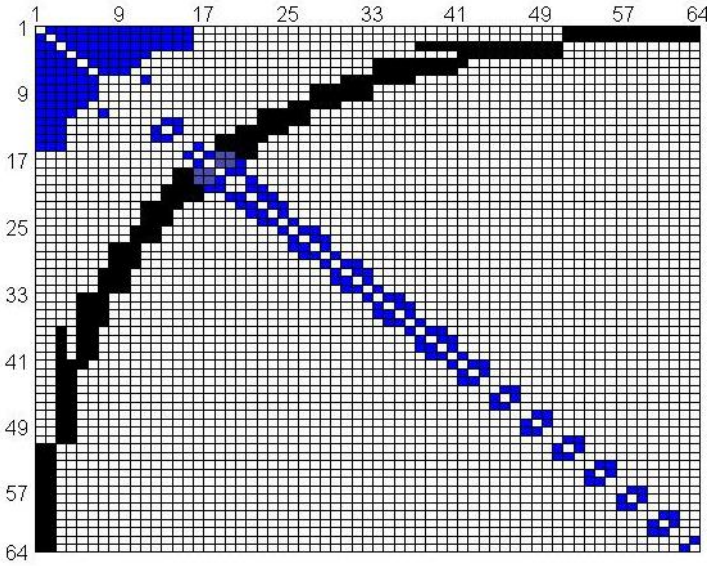


Fig.1 *An adjacency matrix of extremely assortative (blue) and disassortative (black) BA-network of size n=64*

The terms S_1, S_2, S_3 in (1) does not depends on mixing properties (i.e. on adjacency matrix A), but only on network size n and scaling factor β. For SF-model having degree distribution (3) this terms are:

$$S_1 = \frac{c \cdot n}{1-\beta} = 2L, \quad S_2 = O(L^2 \cdot n^{2\beta-2}), \quad S_3 = O(L^3 \cdot n^{3\beta-3}) \tag{4}$$

where *L* is total number of links.

In disassortative case according to generative algorithm the minimum value of N_3 (2) is

$$N_3^{\min} \approx \frac{2c^3}{2\beta-1} n^{-\beta}. \tag{5}$$

By substituting (4) and (5) into (1), we get an asymptotic estimate of the lower bound of assortativity coefficient:

$$r_{\min} = -O(n^{-(1-\beta)}). \tag{6}$$

In the extremely assortative case the contribution to N_3 of clusters of "arrow body" is much less than that of "arrow-head" which is equal to

$$N_{31} = \sum_{i=1}^{c+1} d_i \sum_{j=1}^{d_i+1} d_j \approx \frac{c}{1-\beta} \sum_{i=1}^{c+1} (d_i)^{2-\beta} \approx \frac{c^{4-3\beta+\beta^2}}{(1-\beta)^3}. \tag{7}$$

In accordance with (4) and (7), the upper bound of assortavitity coefficient (1) can be estimated as

$$r_{\max} = O((cn^{\beta})^{1-3\beta+\beta^2}). \tag{8}$$

It can be noted that the obtained estimates of the boundaries of the assortativity index can also be used to normalize the "self-similarity index" [9], which is closely related to the assortativity one.

Obtaining more accurate estimates for the model of elastic SF-networks is treated as a direction for further research.

References

1. Newman, M.(2003) "Mixing patterns in networks", *Phys.Rev.E*,vol.67
2. Dorogovtsev, N., Mendes, J. (2002) "Evolution of networks", *Adv. in Physics*, Vol. 51(4), pp. 1079-1187.
3. Barabási, A.-L., Pósfai, M. (2016). *Network science*. Cambridge: Cambridge University Press. ISBN: 9781107076266 1107076269
4. Noldus, R., Van Mieghem, P. (2015) "Assortativity in complex networks", *J. Complex Networks*, vol. 3, pp. 507-542, 2015.
5. Sendiña-Nadal, I. (2016) "Assortativity and leadership emerge from anti-preferential attachment in heterogeneous networks", *Scientific reports,* vol. 6 21297, doi:10.1038/srep21297
6. Shergin, V., Chala, L., Udovenko, S. (2019). "Assortativity Properties of Scale-Free Networks." 2019 IEEE International Scientific-Practical

Conference Problems of Infocommunications, Science and Technology (PIC S&T), 2019, pp. 723-726. doi: 10.1109/PICST47496.2019.9061369

7. Shergin, V., Udovenko, S., Chala, L. (2021). “Assortativity Properties of Barabási-Albert Networks.” 10.1007/978-3-030-43070-2_4.

8. Xulvi-Brunet, R., Sokolov, I.M. (2005) “Changing correlations in networks: assortativity and dissortativity” Acta Phys.Pol.B,36: 1431.

9. Li, L., Alderson, D., Tanaka, R., Doyle, J.C., Willinger, W. (2005) "Towards a Theory of Scale-Free Graphs: Definition, Properties, and Implications (Extended Version)". arXiv:cond-mat/0501169

EXPLICIT ESTIMATES OF THE PROBABILITIES OF NASH-EQUILIBRIUM STRATEGIES IN LUPI GAME

Shergin V.L.[1], Chala L.E.[1], Udovenko S.G.[2]
[1]Artificial Intelligence dept. NURE, Kharkov, Ukraine,
[2]Informatics and Computer Engineering dept. KNUE, Kharkov, Ukraine

Abstract — LUPI is the game in which players choose positive integers. Winner is the player who picks the lowest unique number. LUPI is the simplest extreme congestion game, in which even one other player’s choice of the same number reduces one's payoff to zero. This primitive lottery is used as a model of some auctions, patent races etc. Poisson-Nash equilibrium strategies for this game are known, but the desired exact probabilities are an implicit function of the game parameters. Finding explicit estimates of these probabilities, as well as an estimate of the threshold value, forms the goal of the current research.

Keywords — LUPI game, Poisson-Nash equilibrium, logint function

In the Least Unique Positive Integer (LUPI) game players, choose integers from 1 to *K* simultaneously, and the lowest unique number wins a payoff of 1, while all others earn 0 [1]. In the simplest form of this lottery, the number of players is considered to be a random variable having a Poisson distribution with mean n. The Poisson assumption proves to be easier to work with than a fixed n because of the environmental equivalence (EE) property [2]. LUPI game used as the model of asymptotic case for Lowest Unique Bid Auctions (LUBA) [3].

Due to the symmetry of the game, all players use the same mixed strategy $\mathbf{p} = (p_1, p_2, ..., p_K)$, where p_k is the desired probability of choosing k. Let $\Pr(X(k) = i)$ is the probability that the number of players who pick k in equilibrium is exactly i. According to the above assumption, this

probability follows a Poisson distribution. Hence, the the expected payoffs v_k when choosing the number k are equal to probabilities $\pi(k)$ of k being the winning number:

$$v_k = \pi(k) = \Pr(X(k) = 0) \cdot \prod_{i=1}^{k-1} \Pr(X(i) \neq 1) = e^{-np_k} \prod_{i=1}^{k-1} \left(1 - np_i e^{-np_i}\right)$$

.

(1)

It was shown [1], that the Nash equilibrium $\mathbf{p}$ has full support, i.e. $p_k > 0$ over all $1 \le k \le K$. According to the basic condition of the Nash equilibrium, the expected payoffs must be equal to each other: $v_1 = v_2 = ... = v_K = v$.

A detailed study of the properties of the Nash equilibrium for the LUPI game was provided by Östling et.al [1]. In particular, they shown that the sequence p_k is strictly decreasing. Moreover, this sequence may have super-exponential cutoff k^* [3], such that $p_k = O(\exp(-2^k))$ for $k > k^*$. If so, then probabilities for $k > k^*$ being the winning number are negligible and such numbers can be discarded despite of full support.

Estimation this cutoff value $k^*(n, K)$ as function on number of players (n) and the maximum number allowed (K) is the first interest of current paper.

According to (1), the desired values p_k can be found sequentially and are fully determined by the initial value p_1. Therefore, these probabilities can be easily found by any simple numerical method as a solution to the normalization condition ($\sum p_k = 1$) w.r.t p_1. However, for large values of K, this approach is associated with both a large amount of calculations and the accumulation of a computational error. This fact forms the main interest of current research: estimating the desired probabilities $p_k(n, K)$ as explicit functions, i.e. without loops and without numerical solution of nonlinear equation.

If we denote

$$z_k = \exp(np_k) \quad \Leftrightarrow \quad p_k = \frac{1}{n}\ln(z_k), \tag{2}$$

then the sequence (1) can be rewritten as recurrence relation

$$z_{k+1} = z_k - \ln(z_k). \tag{3}$$

It should be noted that the exact solution of the recurrence (3) cannot be written in an explicit form. In [3], the original difference equation is approximated by a differential one ($\frac{dz}{dk} = -\ln(z)$), the solution of which has the form

$$k = c - \mathrm{li}(z), \tag{4}$$

where $\mathrm{li}(z)$ is a logarithmic integral function (also called as "logint"), c is an integration constant. As a result, the estimate of cutoff value has the form $k^* = c + \mathrm{li}(n+1)$.

Despite the rather high accuracy of such approximation, this approach has significant limitations: firstly, the uncertainty of the constant c, and secondly, the impossibility of obtaining an inverse relationship (i.e., $z(k)$) in an explicit form.

We propose to estimate the cutoff as the point at which the curvature of the curve $z(k)$ is maximum. Hence, the second (central) difference of this recurrent sequence should be maximum:

$$\Delta^{(2)} z_k = (z_{k+1} - z_k) - (z_k - z_{k-1}) = \ln z_{k-1} - \ln(z_{k-1} - \ln z_{k-1}) \tag{5}$$

The maximum of function (5) is reached at $z_{k-1} = e$, so $z_k = e - 1 \approx 1.718$. Therefore, cutoff is the solution of $z_{k^*} = e - 1$.

Then we found, that sequence (3) can be approximated much more simply than (4), namely as

$$k^* - k = a \cdot z_k + 2\sqrt{z_k} - 3, \quad a \approx 0.116465577755347. \tag{6}$$

In fact, (6) is an approximation of the logint function $\mathrm{li}(y)$ based on the interval $k^* - k \in [e-1; 10000]$.

Using (6), it is easy both to find the cutoff k^* and to approximate explicitly all values of the array z.

According to (3), the normalization condition can be transformed into a telescopic series:

$$n = n\sum_{k=1}^{K} p_k = \sum_{k=1}^{K} \ln(z_k) = \sum_{k=1}^{K} (z_k - z_{k+1}) = z_1 - z_{K+1}. \tag{7}$$

Suppose that $k^* < K$. Then $z_{K+1} = 1$, so, $z_1 = n+1$, and as per (6):

$$k^* = a \cdot z_1 + 2\sqrt{z_1} - 2. \tag{8}$$

Since $z_{\min} = 1$, and given that k is an integer, the maximum allowable value of k, for which estimate (6) is applicable, is

$$k_{\max} = \left\lceil k^* + 1 - a \right\rceil. \tag{9}$$

Thus, if the value of $k_{\max}$ (9) does not exceed K, then all elements of z_k for $k > k_{\max}$ can be set to unity. Moreover, the remaining elements can be expressed as an explicit function of k:

$$z_k = \left(\frac{-1 + \sqrt{1 + a(3 + k^* - k)}}{a} \right)^2, \quad k = 1, \ldots, k_{\max}. \tag{10}$$

In the opposite case ($k_{\max} > K$), value of z_1 can be evaluated as:

$$z_1 = \left(\frac{n + b^2}{2b} \right)^2, \quad \text{where} \quad b = (K - a \cdot n) / 2. \qquad (11)$$

The value of k^* is still evaluated by (8), $k_{\max}$ is set to K, hence all elements of the array *z* are estimated by (10).

In both cases, after calculating the auxiliary array *z*, the probabilities of Poisson-Nash equilibrium strategies can be obtained from (2):

$$p_k = \frac{\ln z_k}{n}. \qquad (12)$$

According to numeric simulations, the proposed algorithm for estimating the probabilities of equilibrium provides high accuracy.

References

1. Östling, Robert & Wang, Joseph & Chou, Eileen & Camerer, Colin. (2007). "Testing Game Theory in the Field: Swedish LUPI Lottery Games." Stockholm School of Economics, Working Paper Series in Economics and Finance. 3. doi:10.1257/mic.3.3.1.
2. Myerson, Roger B. (1998). "Population Uncertainty and Poisson Games." *International Journal of Game Theory*, 27 (3): 375-92.
3. Pigolotti, Simone & Bernhardsson, Sebastian & Juul, Jeppe & Galster, Gorm & Vivo, Pierpaolo. (2012). "Equilibrium Strategy and Population-Size Effects in Lowest Unique Bid Auctions." *Physical review letters*. 108. doi:10.1103/PhysRevLett.108.088701

FEATURES OF PROVISION OF FUNCTIONAL STABILITY OF AIRCRAFT ONBOARD EQUIPMENT COMPLEX

Kalashnyk G.A., Kalashnyk-Rybalko M.A.

Flight Academy of the National Aviation University

Abstract. The authors have presented the features of the formation and maintenance of the functional stability properties of aircraft onboard equipment complex under the influence of destabilizing factors.

Let's consider the complex of onboard equipment of the aircraft as a distributed intelligent system of the form:

$$G(V,L), V = \{ v_i \}, L = \{ l_{ij} \}, \; i, j = 1,2, \ldots, n, \qquad (1)$$

where the set of vertices V of the graph $G(V,L)$ corresponds to the set of information (intelligent) modules of the complex of on-board equipment with dimension n, the set of edges L corresponds to the set of connections l_{ij} between information modules (i, j).

We present a graph of the model of the onboard equipment complex in the form of an adjacency matrix [1]:

$$A = \{ a_{ij} \}, \quad i, j = 1,2, \ldots, N, \qquad (2)$$

where a_{ij} – elements of the adjacency matrix of the diagnostic graph, which have a value of 1, if there is an edge:

$$l_{ij} = \{ v_i, v_j \}, \in L, \qquad (3)$$

and 0 – if the edge does not exist; v_i, v_j – modules of the distributed system of the onboard equipment complex, connected by information communication lines l_{ij}.

The space of parameters P of the onboard equipment complex is represented as $\langle V, L, F \rangle$, where the set of vertices V of the graph $G(V, L)$ corresponds to the set of information (intelligent) modules of the on-board equipment complex, the set of edges L corresponds to the set of connections l_{ij} between information modules (i, j), F – the set of functions of information (intelligent) modules of the onboard equipment complex and connections between them.

We will assume that the functioning of the on-board equipment complex is determined by a set of parameters $p\left\langle v, l, f\right\rangle$, where the subset $v \subseteq V$, $\left|v\right| \le \left|V\right|$, the subset $l \subseteq L$, $\left|l\right| \le \left|L\right|$, and the subset $f \subseteq F$, $\left|f\right| \le \left|F\right|$. Accordingly, the parameter $p\left\langle v, l, f\right\rangle$ – is an element of the set of parameter spaces $P\left\langle V, L, F\right\rangle$. The change in the internal state of the on-board equipment complex over time $w\left(t, p\right)$ depends on p. Thus, $t \in T$, where T – a time interval of functioning of aircraft onboard equipment.

The strategy for implementing the principles of functional stability of the onboard equipment complex is as follows. At the construction stage, the provision of functional stability is carried out by introducing structural, hardware, software and time redundancy. For this purpose, the optimal (according to the criterion of maximum of functional stability) structure of the aircraft on-board equipment complex, characterized by structural and hardware redundancy, is determined. The main failures and errors in the operation of the onboard equipment complex are parried by software. To do this, software and temporal redundancy are introduced into the software to control, diagnose and decrypt diagnostic information. Then for the functional stability of the on-board equipment complex, as the property of a complex system will be fair:

$$\forall\, v \subseteq V\ \exists\, v_{\max,}\ v_{\min,}\ \Delta v \neq \varnothing,$$

$$\forall\, l \subseteq L\ \exists\, l_{\max,}\ l_{\min,}\ \Delta l \neq \varnothing,$$

$$\forall\, f \subseteq F\ \exists\, f_{\max,}\ f_{\min,}\ \Delta f \neq \varnothing,$$

where

$$v = v_{\max} \cup v_{\min,}\ \Delta v = v_{\max} \cap v_{\min}\ ,$$

$$l = l_{\max} \cup l_{\min,}\ \Delta l = l_{\max} \cap l_{\min}\ ,$$

$$f = f_{\max} \cup f_{\min,}\ \Delta f = f_{\max} \cap f_{\min}\ ,$$

where $v_{\max,}$, $l_{\max}$, $f_{\max}$ – maximum subsets in terms of capacities, respectively, of elements, lines of communication, functions of elements of the

aircraft on-board equipment complex, for which the conditions are fair: $|v_{max}| = |V|$, $|l_{max}| = |L|$, $|f_{max}| = |F|$,

v_{min}, l_{min}, f_{min} – minimum subsets in terms of capacities, respectively, of elements, lines of communication, functions of elements of the aircraft on-board equipment complex, for which the conditions are fair: $|v_{min}| < |V|$, $|l_{min}| < |L|$, $|f_{min}| < |F|$,

$\Delta v, \Delta l, \Delta f$ – formal description of redundancies, respectively, of elements, lines of communication, functions of elements of the aircraft on-board equipment complex.

Provision of the functional stability of the on-board equipment complex includes stages of self-diagnosis, failure prediction by verification of the distributed knowledge base, failure parrying, restoration of the structure of the on-board equipment complex without significant restructuring of verification means [2, 3, 4].

But to provide the functional stability of the on-board equipment complex, it is necessary to take into account the rapid parrying of failures, or the resumption of work in a minimum of time. Therefore, the functional stability of the on-board equipment complex describes a family of probabilities of many probable indicators that ensure operation under the influence of disturbance, perhaps with some estimates of characteristics, but the ability to resume work in a given minimum time:

$$P(F_{avionics\tau}) = P\left\{ F_{avionics\tau}\left[w(t, \langle v, l, f \rangle), t \le \tau \right] \in G^{\tau}_{A_1} \right\}, \quad (4)$$

where $P(F_{avionics\ \tau})$ – the set of probabilistic indicators of functional stability;

$w(t, \langle v, l, f \rangle)$ – the internal state of the system, which is an element of the phase space P;

p – a parameter that determines the duration of the effects of destabilizing factors and the moments of their effects;

v – the set of elements of the system model;

l – the set of model connections;

f – the set of functions of elements and connections of the system;

A – the set of system model resources $a = \langle v, l, f \rangle$;

$G_{A_1}^{\tau}$ – the set of values of all functions of the complex of onboard equipment for the model of its operation under the influence of destabilizing factors over time τ.

In this case, the functional stability of the aircraft on-board equipment complex is provided using the principle of synergistic regenerative control in some minimum of time.

References

1. Mashkov O., Chumakevych V., Sokulsky O., & Chyrun L. Features of determining controlling effects in functionally-stable systems with the recovery of a control. *Mathematical Modeling and Computing*. 2019. 6, 80–86.

2. Калашник Г.А., Калашник-Рибалко М.А**.** Ознаки та критерії функціональної стійкості інтегрованого комплексу бортового обладнання сучасного повітряного судна та перспективні напрямки його розвитку. *Збірник наукових праць Харківського національного університету Повітряних Сил*. 2021. №2(42). С.7-15.

3. Калашник-Рибалко М.А., Калашник Г.А. Удосконалення методу синтезу оптимально розподіленої структури інтегрованого комплексу бортового обладнання літального апарата за показниками функціональної стійкості. *Наука і техніка Повітряних Сил Збройних сил України*. 2021. №1(42). С.41-49.

4. Калашник Г.А., Калашник-Рибалко М.А. Проблеми забезпечення функціональної стійкості комплексу бортового обладнання сучасного повітряного судна. *Наука і техніка Повітряних Сил Збройних сил України*. 2021. №3(44). С.59-65.

DECISION-MAKING MODELING IN CASE OFEMERGENCY “CAPTAIN INCAPACITATION AT TAKE OFF”

Shmelova T., Yatsko M., Sierostanov I.

National Aviation UniversityNational Aviation University, aircompany ”Bees Airlines”Aircompany ”Bees Airlines”

Abstract. Optimization deterministic model of decision making in an emergency "Pilot incapacitation" have presented. Behavioral models are used

for synchronization actions of all operators, to support and timely predicting of human-operator actions in the emergence. Models are used in the Decision Support Systems of aviation personnel in emergencies.

Introduction. Analysis of aviation events shows that the cause of more than 80% of them is the human factor. Moreover, most aviation specialists are sufficiently trained as professionals. The occurrence of an emergency situation in flight is characterized by incompleteness and uncertainty of information, decision-making (DM) takes place in conditions of acute shortage of time and significant psycho-physiological load, which is one of the reasons for violations [1]. Finding the best solution in such conditions requires processing much more information than under normal flight conditions. The development of emergency decision-making models is relevant to improving existing Decision Support Systems (DSS).

The work is dedicated to the problem of developing emergency situation (ES), called "Pilot incapacitation". The data shows that the cause of approximately three accidents per thousand (15 per 1,000 deaths) is the result of pilot incapacitation. DM modeling by aviation specialists in conditions of certainty, risk, and uncertainty, the main causes of the accident are considered, a systematic analysis of the situation (decomposition of operational procedures) using the method of expert assessments determined the average time of procedures and consistency of experts) [1]. The deterministic DM model of an aircraft pilot in case of an emergency situation such as "Pilot incapacitation" was created.

Main part. Pilot incapacitation is defined as any condition which affects the health of a crew member during the performance of duties - associated with the duty/position assigned to him which renders him incapable of performing the assigned duties. The definition includes either total or partial incapacitation which does not allow the fulfilment of duties in the "normal" way.

Iincapacitation occurs frequently compared with other routinely trained non-normal conditions. It has occurred in all age groups and during all phases of flight. Incapacitation occurs in many forms ranging from sudden death to subtle, partial loss of mental or physical performance. Subtle incapacitations are the most dangerous and they occur the most frequently. Incapacitation effects can range from loss of function to unconsciousness or death. The key to early recognition of pilot incapacitation is the regular use of crew resource management concepts during flight deck operation.

Proper crew coordination involves checks and crosschecks using verbal communications. Routine adherence to standard operating procedures and standard profiles can aid in detecting a problem. Suspicion of some degree of gross or subtle incapacitation should also be considered when a crewmember

Tab. 1. Structural and temporal table “Captain incapacitation at V1”

№	Actions in case of «pilot incapacitation»	Action	Order	t, s.
1	2	3	4	5
1	Call for intention of take control and continuation takeoff: «Continue, My controls»	a_1	-	2
2	Rotate aircraft at rotation speed V_R	a_2	a_1	2
3	Retract landing gear when positive rate of climb	a_3	a_2	2
4	Engage AP at 400’ feet AGL	a_4	a_3	2
5	Check the position of essential controls and switches	a_5	a_4	10
6	Accelerate aircraft to flaps retraction speed	a_6	a_5	12
7	Retract flaps	a_7	a_6	8
8	Inform ATC	a_8	a_7	6
9	Call cabin crew to help incapacitated pilot.	a_9	a_8	6
10	Follow the SID	a_{10}	a_8	....
11	Brief cabin crew to restrain the incapacitated pilot and provide other assistance	a_{11}	a_9	60
12	*Pilot:* Check weather, type of approach and airport facilities at departure airdrome to evaluate ability to come back or choose appropriate alternate. *Cabin crew:* Restrain the incapacitated pilot and slide the seat back, assess him medical condition, provide with necessary care, call other cabin crew to find doctor on board.	a_{12}	a_{11} a_{12}	300
13	Make a decision to proceed suitable airdrome according actual situation	a_{13}	a_{12}	60
14	Prepare cabin for approach	a_{14}	a_{13}	120
15	Brief cabin crew for approach	a_{15}	a_{14}	60
16	Declare “MAY DAY” and request vectoring for approach and medical assistance	a_{16}	a_{13}	12
17	Proceed as instructed by ATC	a_{17}	a_{16}	.....
18	Slow the aircraft to extend flaps and gear	a_{18}	a_{17}	20
19	Extend flaps and gear for landing	a_{19}	a_{18}	60
20	Read with a help of cabin crew landing check list	a_{20}	a_{19}	20
21	Perform landing	a_{21}	a_{20}	60
1	2	3	3	5

22	Check activation of autobrakes and speedbrakes systems or apply manually brakes and rise speedbrakes lever	a_{22}	a_{21}	2
23	Apply reverse thrust	a_{23}	a_{21}	2
24	Keep directional control with pedals and differential braking, when required	a_{24}	a_{22}	12
25	Stop the aircraft, set parking brakes, and announce "Attention Cabin Crew! At stations!", and	a_{25}	a_{24}	4
26	Inform ATC and request assistance (stairs, etc).	a_{26}	a_{25}	60
27	Take the left seat for further taxi, when requested by ATC to vacate the active RW.	a_{27}	a_{26}	60

does not respond to any verbal communication associated with a significant deviation from a standard procedure or standard flight profile. Failure of any crewmember to respond to a second request or a checklist response is cause for investigation.

In process of DM modeling in case of emergency "Pilot incapacitation" was used the recommendations of the Flight Crew Training Manual "Two communication rule" [2 - 4]. Take-off is the most critical case when pilot incapacitation requires immediate decision and prompt actions.

Proper crew coordination and use of standard call-outs are vital. Failure of crew members to timely respond and/or act on "80 Knots" and "V1" call-outs will require the following immediate actions from any pilot. DM Modeling in the event of an emergency is considered at the second stage of takeoff at a decision speed V1. Decomposition of the emergency "Captain incapacitation", the list and the average time of the pilot's actions are given in tab. 1.

According to the structural and temporal table, a DM deterministic model in the emergency situation is constructed in the form of a network graph, the critical time of execution of all actions by the pilot (Tcr = 988 s), critical path is determined (Fig. 1).

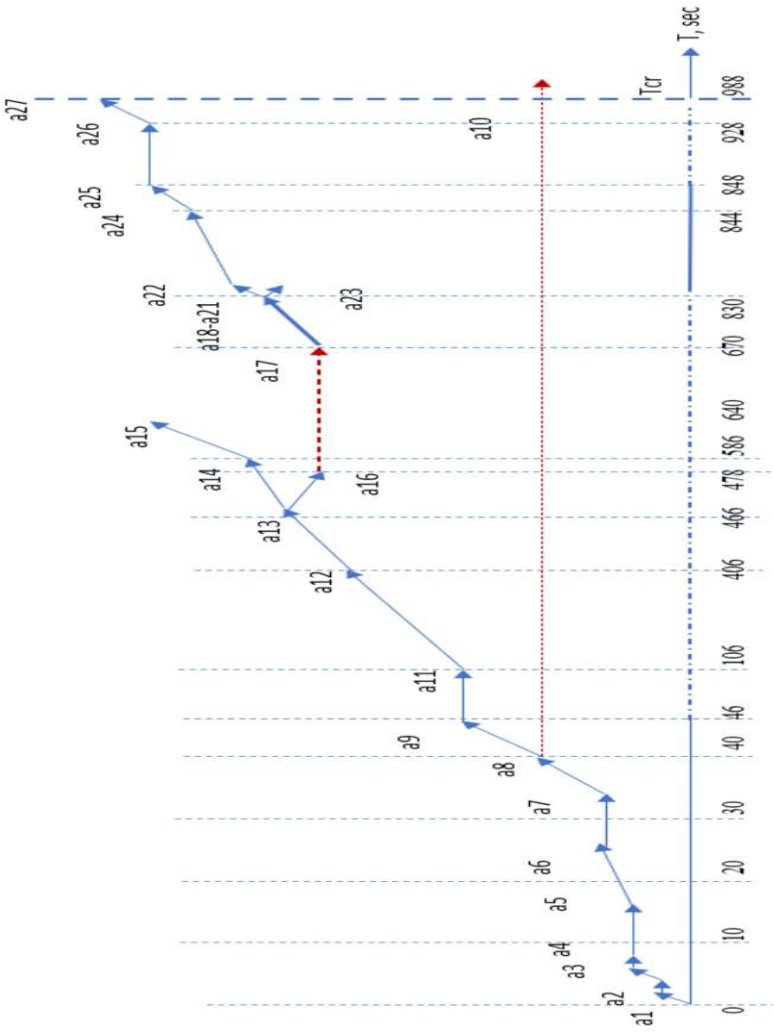
Conclusions. The paper considers the emergency situation "Pilot incapacitation", conducted a systematic analysis of the pilot's actions, created a DM deterministic model, determined the critical time for parrying the existing emergency situation. In the future it is planned to analyze the DM of specialists involved in the process of solving the emergency situation (Air traffic controller, medical services, emergency services, etc.), construction of deterministic and stochastic joint DM models for integrated DMSS [5; 6].

References

1. Intelligent Expert Decision Support Systems: Methodologies, Applications and Challenges /Abdel-Badeeh M. Salem, Tetiana Shmelova -

International Publisher of Progressive Information Science and Technology Research, USA, Pennsylvania, 2018. – P. P.215-242.

Fig. 1. DM deterministic model in the event of "Pilot incapacitation"



2. SkyBrary “Pilot Incapacitation”. URL: https://skybrary.aero/articles/pilot-incapacitation (Last accessed: 19.04.2022).

3. Flight Crew Training Manual Boeing 737 NG. URL: https://aviation-is.better-than.tv/B737NG_FCTM_(31-10-05).pdf (Last accessed: 19.04.2022).

4. Quick Reference Handbook Boeing 737 NG. URL: https://aviaforum.ams3.cdn.digitaloceanspaces.com/data/attachment-files/2013/11/616397_c032fd6493cfbdc1527928147668092a.pdf

5. Research Anthology on Decision Support Systems and Decision Management in Healthcare, Business, and Engineering Chapter 56 Applications of Decision Support Systems in Aviation / Tetiana Shmelova, Yuliya Sikirda – USA: IGI-Global Publ, 2021. – P. 1177-1195.

6. Shmelova, T., Chialastri, A., Sikirda, Y., Yatsko, M. Models of Decision-Making by the Pilot in Emergency “Engine Failure During Take-Off” (2021) In CEUR Vol 3101 Workshop Proceedings, CEURWS http://ceur-ws.org/Vol-3101/Paper26.pdf

ПРО ЧИСЛО БОНДІВ 3-РЕБЕРНО ЗВ’ЯЗНОГО ГРАФА

Глухов О.Д., Репета В.К.
Національний авіаційний університет

ON THE NUMBER OF BONDS OF A 3-EDGE CONNECTED GRAPH

Abstract. In this article considers some estimates for the number of *k*-bonds of 3-connected graphs.

В даній статті розглядаються оцінки числа *k*-бондів 3-реберно зв’язного графа.

Нехай $G = G_n$ - 3-реберно зв’язний граф на n вершинах з множиною вершин G^0 і множиною ребер G_n^1 , $|G^0| = n$, $|G^1| = m$, в якому допускаються кратні ребра і петлі; для кожного ребра u позначимо через ∂u множину його кінців (одну або дві вершини).

Для будь-якої множини $A \subset G^0$ графа G також позначимо:

$J(A,G) = \{u \in G^1 : \partial u = \{a,b\}, a \in A, b \notin A\}$.

Якщо при цьому породжені підграфи $G[A]$ і $G[G^0 \setminus A]$ зв’язні, то $J(A,G)$ - називається бондом (мінімальним по включенню розрізом)

графа G , а підграф $G[A]$ -називається торцевим підграфом цього графа [1].

Бонд $J(A,G)$ називається *k*-бондом, якщо $|J(A,G)|=k$; граф $G[A]$ тоді називається торцевим *k*-підграфом графа G , а множина A вершин графа G відповідно називається *k* -торцевою підмножиною. Мінімальну по включенню *k* -торцеву підмножину назвем примітивною. Ребро u назвемо сингулярним, якщо воно належить деякому примітивному *k*-бонду, але не належить жодному не примітивному бонду.

Зауважимо, що для числа $\sigma_k(G)$ *k*-бондів *k*-реберно зв'язного графа відомі наступні оцінки [2]:

$$\sigma_k(G)=O(n^2),\ k=2j\ ;$$
$$\sigma_k(G)=O(n),\ k=2j-1.$$

Наступні твердження стосуються 3-реберно зв'язних графів.

Лема 1. Якщо A, B 3-торцеві підмножини множини вершин 3-реберно зв'язного графа G і $|A|\le|B|\le n/2$, то $A\subseteq B$ або $A\cap B=\varnothing$.

Лема 2. Якщо A примітивна 3-торцева підмножина множини вершин 3-реберно зв'язного графа G і $|A|>1$, то $G^1[A]\neq\varnothing$ і $\forall u\in G^1[A]$ $G\setminus u$ - також 3-реберно зв'язний граф.

Позначимо через G^* граф, який отриманий з даного графа G заміною кожної пари ребер u_1, u_2, таких , що $\partial u_1=\{a,c\}, \partial u_2=\{c,b\}, \rho(c)=2, \rho(a)>2, \rho(b)>2$, на одне ребро u, $\partial u=\{a,b\}$ (ланцюг довжини 2 з вершиною степеня 2 заміняється одним ребром).

Лема 3. Якщо A примітивна 3-торцева підмножина множини вершин 3-реберно зв'язного графа G і $|A|=1$. $u\in J(A,G)$ і u сингулярне ребро, то граф $(G\setminus u)^*$ - також 3-реберно зв'язний.

За допомогою лем 1-3 можна довести наступне твердження.

Теорема 1. Якщо G_n - 3-реберно зв'язний граф на n вершинах, $n \geq 3, k \geq 3$ то

$$\sigma_k(G_n) \leq 2^{k-2} \begin{pmatrix} n \\ k-2 \end{pmatrix}.$$

Доведення. Зауважимо спочатку, що достатньо розглянути тільки критичні 3-реберно зв'язні графи, тобто такі графи, у яких видалення будь-якого ребра призводить до зменшення реберної зв'язності. Згідно з лемою 3 в критичних графах існує таке ребро *u*, що граф $(G \setminus u)^*$ буде 3-реберно зв'язним. Звідки випливає нерівність:

$$\sigma_k(G) \leq 2\sigma_{k-1}((G \setminus u)^*) + \sigma_k(G_u),$$

де G_u - граф, що отриманий з графа G стягуванням ребра *u* .

Далі індукція завершує доведення теореми.

Зауважимо, що для планарних 3-зв'язних графів мають місце такі оцінки [3 , 4] :

Теорема 2. Якщо G - планарний 3-зв'язний граф, то

$$\sigma_3(G) \leq 4n/3.$$

Теорема 3. Якщо G - простий (без петель і кратних ребер) планарний 3-зв'язний граф, то при $k \geq 4$ має місце наступна нерівність:

$$\sigma_k(G) \leq \frac{1}{2k}(2m)^{k/2} .$$

Отримані результати дозволяють висунути наступні припущення щодо оцінки числа $\sigma_k(G)$ *k*-бондів довільного *3*-реберно зв'язного графа.

Припущення 1. Існує таке $\alpha < 1$ і така стала С що $\sigma_k(G_n) = Cn^{\alpha k}$.

Припущення 2. Існує таке така стала С що $\sigma_k(G_n) = Cn^{k/2}$.

Література

1. Татт У. Т. Теория графов. – М.: Мир, 1968. -484с.

2. Диниц Е.А., Карзанов А.В., Ломоносов М.В. О структуре системы минимальных реберных разрезов графа. –Исследования по дискретной оптимизации, Москва, Наука, 1976, с. 290-306.

3. Глухов О.Д. Про зв'язність планарних pr- графів пуассонівського типу// II Український математичний конгрес, 27-29 серпня 2009 р.: тези доп. – К., 2009. – режим доступу: http://www.imath. kiev.ua/~congress2009.

4. Глухов О.Д Про планарні квазівипадкові графи пуассонівського типу. Моделювання та інформаційні технології. Збірник наукових праць ІПМЕ НАНУ, вип.57, Київ, 2010, с. 10-12.

ПРО ЗВ'ЯЗОК ЗАДАЧ ІЗОМОРФІЗМУ ГРАФІВ ТА УПОРЯДКУВАННЯ ЇХ ВЕРШИН

Караваєв К. Д., Турчина В.А.
Дніпровський національний університет імені Олеся Гончара

ON THE RELATION BETWEEN GRAPH ISOMORPHISM AND PARALLEL SEQUENCING PROBLEMS

Abstract. The paper investigates the possibility of constructing a complete graph invariant based on lengths of optimal sequencings obtained for different values of the sequencing width. It is shown that such an invariant will have a number of impractical limitations. Nonetheless, they may give rise to a new direction of research considering finding the length of an optimal sequencing.

Одним з відомих прикладів задач теорії розкладів є задача оптимального упорядкування вершин орієнтовних графів. Широке дослідження цієї задачі почалося у середині минулого століття і було пов'язане з удосконаленням конвеєрів автомобільного виробництва шляхом оптимізації порядку виконання робіт на них. Інтуїтивний поліноміальний розв'язок формалізованої задачі виявився оптимальним, що дало підстави сподіватися, що й у загальному випадку задача є розв'язною за поліноміальний час.

Подальші зусилля дослідників були спрямовані на розробку точних методів, заснованих на схемах направленого перебору, та пошук

підкласів задач, для яких існують поліноміальні алгоритми знаходження точних розв'язків. Такі алгоритми знаходять своє застосування в різноманітних сферах: у процесорах та програмному забезпеченні обчислювальних систем, виробничій та споживчій сферах, приладобудуванні, тощо.

Розглянемо класичну постановку задачі оптимального упорядкування. Нехай маємо n завдань, порядок виконання яких має підпорядковуватись деяким наявним виробничим залежностям. Математичну модель такої задачі природно подавати у вигляді орієнтовного графу $G(V, U)$, у якому вершини відповідають завданням, а дуги – виробничим залежностям. В класичній постановці вважається, що всі виконавці універсальні та 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 $(i, j) \in U$ випливає, що вершина i розташовується в упорядкуванні S лівіше вершини j, тобто з того, що $(i, j) \in U \wedge (i \in S[p], j \in S[q])$ випливає, 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 $i = 1, \dots, n$.

Задача 1 (класична). По заданим графу G і значенню ширини h побудувати паралельне упорядкування мінімальної довжини.

Задача 2 (узагальнена). По заданим графу G і вектору значенньширини $h_i, i = 1, \dots, n$ побудувати паралельне упорядкування мінімальної довжини, в якому на i-ому місці буде стояти не більше h_i вершин.

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

Однією з найважливіших задач теорії графів є перевірка пари графів на ізоморфність.

Означення 4. Два орієнтовні графи G та H є ізоморфними, якщо між множинами їх вершин існує взаємо однозначна відповідність, що зберігає суміжність вершин та орієнтацію дуг.

Означення 5. Інваріантом графу G називається деяка характеристика графу, найчастіше числова, пов'язана із графом G, яка

зберігається для будь-якого графу, що є ізоморфним G. Прикладами інваріантів є кількість вершин чи дуг у графі.

На разі невідомо чи належить ця задача до класу P або NP-повних, тому вона віднесена до окремого класу складності GI.

Розглянемо питання взаємозв'язку цих задач. Так у роботі [1] було досліджено вплив перевірки ізоморфізму графів на швидкість збіжності алгоритму методу гілок та меж для таких задач. Метою подальших досліджень став зворотній зв'язок.

Так, оскільки складність задачі перевірки графів на ізоморфність, взагалі кажучи не перевищує складності задачі оптимального упорядкування, то цікавою є принаймні теоретична можливість побудови повного інваріанту на основі розв'язання аналогічних задач упорядкування вершин для графів.

Можна одразу зробити ряд спостережень. По-перше, зрозуміло, що довжини упорядкувань для ізоморфних графів при розв'язанні як класичних, так і узагальнених задач, є інваріантами за будь-яких значень h та h_i. По-друге, якщо розглянемо тривіальну задачу з $h = 1$, то отримані довжини будуть дорівнювати кількості вершин у графах, тобто цей інваріант додатково перевіряти не потрібно. По-третє, відомо, що транзитивні дуги у графі не впливають на довжини отриманих упорядкувань, а отже не зможемо відрізнити графи, що відрізняються лише наявністю таких дуг. По-четверте, для класичної задачі зміна орієнтації всіх дуг на протилежні не призводить до зміни оптимальної довжини, тому для побудови інваріанту необхідно розглядати також узагальнені задачі.

Враховуючи попередні спостереження, дослідимо тепер можливість перевірки співпадіння кількості дуг у графах, якщо серед них немає транзитивних.

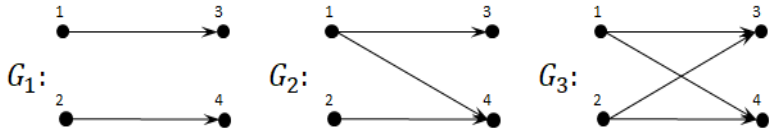


Рис.1. *Співпадіння графів з різною кількістю дуг*

Розглянемо графи з рис. 1. Графи G_1 та G_3 можна відрізнити, якщо побудуємо їх розв'язки для задачі з $h_i = \{1,2,1,1\}$. Легко побачити, що для графу G_1 довжина оптимального упорядкування складає 3, а для графу G_3 – 4. Якщо розглянемо тепер пару графів G_1 та G_2, то побачимо, що довжини їх оптимальних упорядкувань для такої задачі співпадають.

Більш того, вони співпадуть для будь-яких значень h_i. В цьому можна переконатися застосувавши наступні міркування. Нехай $h_1 = 2$, тоді можемо розмістити на перше місце лише вершини 1 та 2, після чого в обох випадках отримаємо ізоморфні графи, а отже вони будуть мати однакову довжину упорядкування згідно першого спостереження. Аналогічно, якщо $h_1 = 1$, тоді, розмістивши на цьому місці вершину 1, також отримаємо ізоморфні графи, а отже й однакові довжини оптимальних упорядкувань. Аналогічно можна переконатися, що не зможемо відрізнити ці графи при застосуванні будь-яких перетворень, що не впливають на ізоморфізм.

З попереднього прикладу випливає необхідність перевірки кількості дуг у графі окремо. Залишається перевірити можливість відрізнити графи, якщо вони мають однакову кількість дуг та не мають транзитивних дуг.

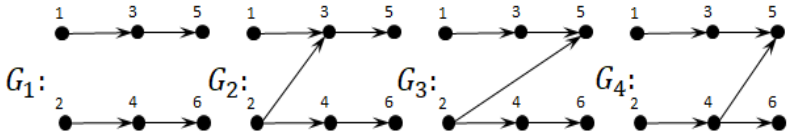


Рис.2. *Співпадіння графів з однаковою кількістю дуг*

Розглянувши графи G_2, G_3 та G_4 з рис. 2 та застосувавши до них попередні міркування, переконаємося, що вони також будуть всі мати однакові довжини оптимальних упорядкувань для довільних h_i. Більш цього, ці довжини співпадатимуть з оптимальними довжинами для графу G_1 для тих самих значень h_i.

З усього зазначеного можна зробити висновок, що побудувати повний інваріант на основі розв'язання задач оптимального упорядкування, якщо і можливо, то за великої кількості непрактичних обмежень на досліджувані графи.

З іншого боку, цікавим є подальше дослідження можливості виділення дуг, які не впливають на довжину упорядкування, для спрощення графів, що є суттєвим для задач великої розмірності.

Література

1. Караваєв К.Д., Турчина В.А. Аналіз впливу автоморфізму графу на схеми направленого перебору. – Питання прикладної математики і математичного моделювання. Дніпро, 2021 р. – С. 94-104.

МЕТОД ОТЖИГА НА ФРАГМЕНТАРНОЙ СТРУКТУРЕ ДЛЯ ЗАДАЧИ РАЗМЕЩЕНИЯ ПРОИЗВОДСТВА

Козин И.В., Алломов З.К.
Запорожский национальный университет, Украина
Ургенчский государственный университет, Узбекистан

ANNEALING METHOD ON A FRAGMENTAL STRUCTURE FOR THE PROBLEM OF PRODUCTION LOCATION

Abstract. The problem of placing production with restrictions on production volumes is considered. To search for suboptimal solutions to the problem, an algorithm is proposed based on the annealing method on a fragmental structure.

Рассматривается классическая задача размещения производства [1] в следующей постановке: задано множество $U = \{u_1, u_2, ..., u_n\}$ пунктов возможного размещения производства и множество $V = \{v_1, v_2, ..., v_m\}$ клиентов, которые могут обслуживаться в этих пунктах. Для каждой пары *i*-й пункт производства, *j*-й клиент определены затраты α_{ij} на обслуживание клиента в соответствующем пункте производства. Каждый клиент должен обслуживаться в одном и только в одном пункте производства. Для каждого пункта производства i определены затраты b_i на содержание этого пункта. Задача состоит в том, чтобы определить, какие пункты производства должны быть открыты (по крайней мере один) и каково распределение клиентов между этими пунктами.

$$F(x, y) = \sum_{i=1}^{n} y_i b_i + \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_{ij} x_{ij} \to \min$$

$$\sum_{i=1}^{n} x_{ij} = 1; \quad \sum_{i=1}^{n} x_{ij} y_i = 1, \; j = 1, 2, ..., m$$

$$x_{ij} \in \{0,1\}, \; y_i \in \{0,1\}, \quad i = 1, 2, ..., n; \; j = 1, 2, ..., m$$

Для решения поставленной задачи предлагается использовать фрагментарную модель [2]. Множеством элементарных фрагментов является множество связей пункт обслуживания – клиент $U \times V$,

элементы которого нумеруются числами $1,2,...,nm$. Фрагментарный алгоритм размещения заключается в следующем:

а) составляется список элементарных фрагментов(связей) упорядоченный в соответствии с некоторой перестановкой чисел $1,2,...,nm$;

б) выбирается очередной по порядку фрагмент (ij) из списка. Клиент с номером *j* закрепляется за пунктом обслуживания с номером *i*;

в) все связи клиента *j* с другими пунктами обслуживания удаляются из списка;

г) алгоритм заканчивает работу, когда список станет пустым.

Результатом работы фрагментарного алгоритма будет распределение клиентов между пунктами обслуживания, которое является допустимым решением рассматриваемой задачи.

Показано, что любое допустимое решение задачи размещения может быть получено путем надлежащего выбора перестановки элементарных фрагментов.

Для поиска оптимального решения задачи предлагается использовать метод имитации отжига [3] на фрагментарной структуре. Алгоритм метода имитации отжига для этой задачи будет состоять из следующих шагов:

Шаг 0. На этом шаге задается коэффициент охлаждения $\beta \in (0,1)$, величина максимальной температуры $T_0 > 0$ і максимальное число шагов *K* алгоритма. Выбирается случайным образом начальная перестановка, которая задает соответствующее решение задачи $x^* = (i_1, i_2, ..., i_n)$ и вычисляется значение целевой функции $\varphi(x^*)$.

Шаг *k*($1 \le k \le K$). Случайным образом выбираются две позиции *s* и *t* в текущей перестановке $x^* = (i_1, i_2, ..., i_n)$ такие, что $1 \le s < t \le n$. Формируется новая перестановка $x^k = (i_1, i_2, ..i_{s-1}, i_t, i_{t-1}...i_s, i_{t+1}, ..., i_n')$, в которой элементы $i_s, i_{s+1}, ..., i_{t-1}, i_t$ переставляются в обратном порядке. Вычисляется значение целевой функции $\varphi(x^k)$ на перестановке x^k. Вычисляется новое значение температуры $T_k = \beta T_{k-1}$. Если $\varphi(x^k) \le \varphi(x^*)$, то полагаем $x^* = x^k$ и переходим к очередному шагу алгоритма. Если

$\varphi(x^k) > \varphi(x^*)$, то вычисляем вероятность $p_k = \exp\left(\frac{\rho(x^*) - \rho(x)}{T_{k-1}}\right)$.

Если $p_k > rand()$, где $rand()$ - функция, которая возвращает равномерно распределенное случайное число в диапазоне [0,1], то полагаем $x^* = x^k$ и переходим ко очередному шагу. В противном случае просто переходим к шагу с номером $k + 1$. Алгоритм заканчивает работу, когда проведено заданное число шагов K. Текущая перестановка x^*, которая определена на последнем шаге берется в качестве субоптимального решения задачи.

Литература

1. Береснев В. Л. Экстремальные задачи стандартизации / В. Л. Береснев, Э. Х. Гимади, В. Т. Дементьев. – Новосибирск: Наука, 1978. – 333 с.

2. 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

3. Лопатин А.С., Метод отжига,/А.С.Лопатин // Стохастическая оптимизация в информатике. СПб. : Изд-во СПбГУ, 2005. Вып. 1. С. 133–149.

ПРОТОКОЛ ОБМІНУ ПОВІДОМЛЕННЯМИ НА ОСНОВІ ЛІНІЙНИХ ФУНКЦІЙ І ОПЕРАТОРІВ

Кривий С.Л., Нортман Ю.О.

Київський національний університет імені Тараса Шевченка
Київ, Україна

Анотація. Пропонується протокол обміну інформацією га основі властивостей лінійних функцій і операторів з реалізацією на мові програмування Java.

PROTOCOL FOR EXCHANGE INFORMATION ON THE BASE OF LINEAR FUNCTIONS AND OPERATORS

Abstract. A protocol for exchange information on the base of linear function and non degenerate linear operators is proposed. This protocol is realized in programming language Java.

1. Протокол обміну повідомленнями

Пропонований протокол обміну повідомленнями ґрунтується на властивостях операторів у лінійному просторі і складності алгоритмів розв'язання систем лінійних неоднорідних діофантових рівнянь в множині натуральних чисел. Використовуючи деяку послідовність таких відображень зі зсувами на деякі задані вектори, дістаємо можливість за результуючим значенням оператора знайти його початкове значення. В протоколі всього чотири кроки, які описують взаємодію між Алісою і Бобом.

Крок 0. Аліса і Боб просять Джона згенерувати для них спільний ключ c – ціле додатне число або вектор таких чисел заданої розмірності. Після отримання такого ключа, виконується такий протокол.

Крок 1. 1) Аліса будує систему лінійних виразів Ax, де A матриця розмірності $p\times q$ $(p < q)$, не вироджені матриці розмірності $p\times p$ $B_0, B_1,\ldots, B_s$ і вектори $b_0, b_1,\ldots, b_s, b$, де b, b_i – приватні вектори зсуву з цілими координатами, $i=1,2,\ldots,s$;

2) Будує систему виразів

$$D(x)= B_s(B_{s-1}(\ldots B_1(B_0(A(x))+b_0)+b_1)\ldots) +b_{s-1})+b_s)+b \text{ або}$$

$$D(x)= B_s(B_{s-1}(\ldots B_1(B_0(A(x)+b_0)+b_1)\ldots)+b_{s-1})+ b_s)+b;$$

3) Висилає Бобу вирази Ax, і $D(x)$ відкритим каналом.

Крок 2. 1) Боб обирає довільний вектор x розмірності $1\times q$ і обчислює значення $Ax=v$, яке є приватним;

2) Обчислює значення $D(x)=d$ і висилає Алісі значення $d'= d +c$ відкритим каналом.

Крок 3. 1) Аліса за значенням d' обчислює значення

$$B_0^{-1}(B_1^{-1}(\ldots(B_{s-1}(B_s^{-1}(d'-c)-b)-b_s)-b_{s-1})\ldots-b_1) -b_0=A(x)$$

за допомогою обернених матриць до матриць B_i.

Надійність наведеного протоколу ґрунтується на секретному ключі c та складності розв'язання системи лінійних діофантових рівнянь в множині натуральних чисел, тобто на криптографічній функції f_c: $Dx+c = d'$. Оскільки саме повідомлення в процесі обміну не фігурує, то очевидними діями для його знаходження є розв'язання системи рівнянь вигляду $Dx+c = d'$. Але в результаті такого розв'язання зловмисник отримує вектор-розв'язок y, який не має ніякого відношення до реального

повідомлення.. Оскільки значення вектора x і c зловмиснику невідоме, то знайти значення Ax він не може.

Складність розв'язання системи лінійних діофантових рівнянь (а точніше, побудови гільбертового базису множини всіх розв'язків системи) в множині натуральних чисел в загальному випадку належить класу складності ***#NP***[1]. Зловмисник може розв'язати відповідну систему, але потім необхідно підібрати значення ключа c, за яким знаходиться явний текст, що виглядає досить проблематичним на підставі нескінченності способів вибору вектора c. Таким чином, в даному протоколі використовується криптографічна функція з секретом $f_c=D(x)+c$, де секретом виступає вектор c.

Складність обчислень в наведеному протоколі не висока і єдине, що потребує затрат часу – це обчислення послідовності добутків матриць і обернених до них матриць, на підставі яких будується вираз D. Але із наведеного вище опису протоколу і обчислень випливає, що можна обійтися двома-трьома матрицями, оскільки послідовність добутків не вироджених матриць буде матрицею і її можна знайти за матрицями A і D. Тому досить мати дві-три матриці і для них знаходити обернені матриці. Проблему обчислення оберненої матриці легко розв'язати, якщо вибирати матриці спеціальним чином. Добре відомо [2], що для матриці вигляду

$$M = \begin{pmatrix} I_n & A & 0 \\ 0 & I_n & B \\ 0 & 0 & I_n \end{pmatrix}$$

обернена матриця має вигляд

$$M^{-1} = \begin{pmatrix} I_n & -A & AB \\ 0 & I_n & -B \\ 0 & 0 & I_n \end{pmatrix}.$$

Це дає можливість кожного разу обирати різні матриці в процесі побудови D.

2. Обмін наперед заданими повідомленнями

Вияснимо, яким чином можна обмінятися заданим повідомленням v, оскільки в реальній ситуації v не завжди може бути розв'язком системи $Ax=v$. Для того щоб обмінятися повідомленням v описаним протоколом, потрібна сумісність системи $Ax = v$. Це потребує додаткових двох

пересилань між абонентами з метою вибору матриці A для забезпечення сумісності системи $Ax = v$. Якщо таке узгодження відбулося, то далі все йде за описаним протоколом.

Можливий і інший спосіб такого узгодження без додаткових пересилань. Таку можливість дає вибір області, над якою розв'язується система $Ax = v$. Якщо Аліса і Боб вибрали скінченне поле G_n, яке ізоморфне полю лишків Z_n за модулем великого простого числа n і систему $Ax = v$, серед рівнянь якої немає лінійно залежних, то така система буде завжди сумісною над цим полем. Крім того, в цьому випадку *TSS*-алгоритм [3] будує базис множини всіх розв'язків системи. Отже, таким способом можна передавати повідомлення довільної довжини.

Зауважимо, що складність розв'язання систем лінійних рівнянь в полі лишків за модулем простого числа належить поліноміальному класу складності [2].

3. Практична частина

Для реалізації протоколу була обрана мова програмування Java 11, оскільки вона є платформо-незалежною, а також зручною для використання. Вихідний код програми компілюється і конвертується у байткод, який може бути запущений на будь-якому пристрої, де встановлена віртуальна Java машина (JVM) без необхідності перекомпіляції та внесення змін у початковий код. Оскільки в проекті використовуються зовнішні бібліотеки, такі як фреймворк логування Log4J та клієнт для брокера повідомлень RabbitMQ, то було вирішено використовувати систему збірки Maven версії 3.8.1. Інформація для збірки міститься у файлі pom.xml, який знаходиться у корені проекту.

Важливу роль у надійності протоколу відіграє вибір числа p, яке має бути великим простим числом. Тому у реалізації використовується тип BigInteger з пакету java.math, який підтримує цілочисельні значення в діапазоні від $-2^{\text{Integer.MAX_VALUE}}$ до $2^{\text{Integer.MAX_VALUE}}$ [4]. Значення Integer.MAX_VALUE є константою і дорівнює 2147483647. Крім того, даний клас надає готовий зручний функціонал операцій для роботи над числами в полі лишків F_p, такий як знаходження мультиплікативного оберненого до числа a, що буде використовуватися при реалізації алгоритму [5].

Література

1. Hermann M., Juban L., Kolaitis P. G. On the Complexity of Counting the Hilbert Basis of a Linear Diophantine System. Springer Verlag. - LNCS. - 1999. - № 1705. - P. 13-32.

2. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы построения и анализа (2-е изд.). Издательский дом «Вильямс». – 2005.–1290 с.

3. Кривий С.Л. Лінійні діофантові обмеження та їх застосування. К.: Інтерсервіс. – 2021.– 260 с

4. AMQP Advanced Message Queuing Protocol Specification [Електронний ресурс] // 0-9-1. – 2008. – Режим доступу до ресурса: https://www.rabbitmq.com/resources/specs/amqp0-9-1.pdf.

5. Videla A. RabbitMQ in Action / A. Videla, J. Williams. – Shelter Island: Manning Publications Co., 2012. – 288 с.

МЕТОД ПРОГРАМНОГО ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ З ВИКОРИСТАННЯМ ГРАФОВОЇ БАЗИ ДАНИХ NEO4J

Мелешко Є.В.,
Центральноукраїнський національний технічний університет, м. Кропивницький

A METHOD OF COMPUTER SIMULATION MODELING OF A RECOMMENDATION SYSTEM USING THE GRAPH DATABASE NEO4J

Abstract. A method of computer simulation modeling of user and bot behavior in a recommendation system, which allows generating datasets for testing the algorithms for generating recommendations, was proposed. The developed method makes it possible to simulate the behavior of both ordinary users and bots, which makes it possible to create datasets for testing the robustness of recommender systems to information attacks, as well as for testing the effectiveness of methods for detecting and neutralizing botnets. The structure of relations between users and objects of the recommender system was modeled using the theory of complex networks. Information attacks of bots were modeled on the basis of known models of profile-injection attacks on recommender systems.

У цій роботі було створено програмну імітаційну модель поведінки користувачів рекомендаційної системи для тестування методів генерації списків рекомендацій, напр., з метою визначення та порівняння показників точності та стійкості різних методів.

Для моделювання структури зв’язків у соціальному графі

рекомендаційної системи було вирішено взяти за основу, але модифікувати з врахуванням специфіки задачі, принципи, на яких базується модель Барабаши-Альберт [1] (а саме, «ріст» та «бажане приєднання»), так як вона проста в реалізації та дозволяє створити стохастичний граф з властивостями соціальних мереж. Етапи запропонованого методу генерації структури соціального графу рекомендаційної системи розглянуті у роботі [2]. Наступним етапом було моделювання поведінки користувачів та ботів рекомендаційної системи. Модель вирішено розробити таким чином, щоб одержувати набір даних схожий на MovieLens datasets [3] за структурою та статистичними властивостями.

Етапи методу програмного імітаційного моделювання звичайних користувачів та ботів у рекомендаційній системі:

1 етап. Ініціалізація параметрів системи, зокрема, вибір кількості користувачів та об'єктів, кількості можливих властивостей у них, проценту активних користувачів, проценту популярних об'єктів, кількості часових ітерацій, після яких модель завершить свою роботу тощо. При необхідності моделювати інформаційну атаку обирається тип атаки, кількість ботів та кількість цілей атаки. Відбувається створення набору можливих властивостей для елементів системи, реалізованих у моделі прихованими факторами, та генерація шаблонів кластерів елементів на основі цих властивостей.

2 етап. Створення «Зерна» соціального графу рекомендаційної системи – до графу додається початкова кількість користувачів та об'єктів, деякій кількості об'єктів виставляються оцінки (кількість оцінок залежить від вибраної щільності графу). Початкова кількість користувачів, початкова кількість об'єктів та щільність графу – налаштовувані параметри. Кожному новому користувачу та об'єкту системи привласнюється значення зміщення, певний кластер та відповідний йому вектор значень прихованих факторів, що визначає ступінь його приналежності до кожної з можливих властивостей для елементів системи. У «Зерні» профілі ботів відсутні, вони починають приєднуватися до мережі на 3 етапі.

3 етап. На кожній ітерації часу моделі до графу приєднується певна кількість користувачів та об'єктів. Ця кількість визначається випадковим чином та лежить у межах від 0 до N, де N менше загальної кількості елементів системи відповідного типу. Також на кожній ітерації часу моделі відбувається вибір деякої кількості пар користувачів та об'єктів для виставлення оцінок. Ймовірність перегляду об'єкту користувачем визначається за формулою (1):

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

де q_{1i} – кількість переглядів у i-го об'єкту, q_{2i} – кількість оцінок у i-го об'єкту, q_{3i} – кількість коментарів у i-го об'єкту.

4 етап. Зупинка роботи імітаційної моделі, збереження згенерованого набору даних у файл для подальшого використання у методах формування та тестування списків рекомендацій.

Генерація оцінки для звичайного користувача рекомендаційної системи здійснюється на основі «прихованих факторів» відповідного об'єкта та користувача. Оцінка для пари користувач-об'єкт визначається за наступними формулами:

$$d_{u,m} = \frac{\sum_{i=0}^{n} \left| f_{u,i} - f_{m,i} \right|}{n}, \quad (2)$$

$$r_{u,m} = \Psi\left(5 d_{u,m} + b_u + b_m\right), \quad (3)$$

де $d_{u,m}$ – дистанція між користувачем u та об'єктом m у багатомірному просторі прихованих факторів, може приймати значення від 0 до 1; n – кількість прихованих факторів у системі; $f_{u,i}$ – 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.

Для генерації оцінок ботів було використано 3 відомі моделі атак на рекомендаційну систему ін'єкцією профілів [4-8]: випадкова, середня та популярна атаки. Генерація оцінки для пар бот-об'єкт здійснюється за формулою (4):

$$r_{u,m} = \begin{cases} attack\,\text{Pattern}(), \text{ якщо об'єкт} - \text{"звичайний"} \\ 5.0, \text{ якщо об'єкт} - \text{"цільовий"} \end{cases}, \quad (4)$$

де $attack\,\text{Pattern}()$ – функція, що генерує оцінки для нецільових об'єктів відповідно до типу обраної моделі атаки.

Програмна імітаційна модель рекомендаційної системи була

розроблена на мові програмування Python та з використанням графової СУБД Neo4j [9]. СУБД Neo4j має наступні структурні елементи, що дають змогу побудувати соціальний граф: node – вузол графа, relation – ребро графа, node label – мітка вузла, relation identirfier – можуть містити різні дані, node ID – унікальний ідентифікатор вузла. Перевагами СУБД Neo4j є гнучка модель даних, аналіз в реальному часі, можливості простого пошуку та фільтрації даних, масштабованість та надійність, наявність візуального інтерфейсу, а також існування додатку Neo4j Aura для застосування у вигляді хмарного сервісу. Отже, СУБД Neo4j надає всі необхідні інструменти для створення програмної імітаційної моделі соціального графу, що можна в подальшому застосовувати для моделювання та дослідження поведінки користувачів та ботів у рекомендаційних системах.

Було проведено серію експериментів для перевірки працездатності розробленої програмної імітаційної моделі. У ході експериментів було згенеровано множину наборів даних. Кожен набір даних було розділено на робочу та тестову вибірку. На основі тестової вибірки перевірялася точність прогнозування вподобань. Результати проведених експериментів показали, що джиттер досліджуваних значень точності, повноти та RMSE прогнозування вподобань у більшості практичних випадків впевнено вкладається в допустимі межі коливань, а отже поведінка користувачів у програмній моделі не була випадковою, а імітувала поведінку реальних користувачів з певними вподобаннями. Це підтверджує достовірність розробленої програмної імітаційної моделі рекомендаційної системи.

Література

1. Barabási A.-L. (2018), *Network science*. Cambridge University Press, 475 p. – URL: http://networksciencebook.com/

2. Meleshko Ye. (2019), *Computer model of virtual social network with recommendation system.* Innovative technologies and scientific solutions for industries, №2(8), P. 80-85. DOI: https://doi.org/10.30837/2522-9818.2019.8.080

3. Harper F.M., Konstan J.A. (2015), *The MovieLens Datasets: History and Context.* ACM Transactions on Interactive Intelligent Systems (TiiS), 19 p. URL: https://doi.org/10.1145/2827872

4. Chirita P.A., Nejdl W., Zamfir C. (2005), *Preventing shilling attacks in online recommender systems.* In Proceedings of the ACM Workshop on Web Information and Data Management, P. 67-74.

5. Ricci F., Rokach L., Shapira B., Kantor P.B. (Editors) (2011), *Recommender Systems Handbook.* Boston: Springer, 842 p. DOI: https://doi.org/10.1007/978-0-387-85820-3
6. Neo4j Documentation (2021), Official website of the graph database Neo4j. URL: https://neo4j.com/docs/

ДОСЛІДЖЕННЯ МЕТОДІВ ЗБЕРІГАННЯ ДАНИХ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ НА ОСНОВІ ЗВ'ЯЗНИХ СПИСКІВ

Міхав В.В., Мелешко Є.В., Якименко М.С.,
Бащенко Д.В.
Центральноукраїнський національний технічний університет, м. Кропивницький

RESEARCH METHODS OF DATA STORING OF A RECOMMENDATION SYSTEM BASED ON LINKED LISTS

Abstract. In the work, research was carried out of various data structures that can be used to store information of the recommender system. In particular, such as linked list, unrolled linked list, hash table, B-tree, B+-tree and binary decision diagram. To carry out experiments comparing the effectiveness of using various data structures in terms of time and memory costs, a computer model of a simplified recommendation system was developed, in which three main entities were distinguished - an agent, a session, and an object. The best results were obtained with data storing methods using unrolled and inverted unrolled linked lists. Therefore, it was decided to also conduct an additional series of experiments with these data structures for different sizes of the list block. The unrolled list showed the best results in terms of memory used and session generation time. The inverted unrolled list showed an advantage in the generation of recommendations.

Було проведено дослідження різних структур даних, які можна використати для створення бази даних рекомендаційної системи [1, 2], а саме нижченаведених.

Зв'язний список (linked list) – структура даних, у якій кожен елемент містить поля даних та вказівник на наступний елемент. Основна перевага цієї структури полягає у сталому часі додавання нового елементу. Проте для кожного елементу потрібно виділяти новий блок

пам’яті, тому менеджер пам’яті спричиняє значні затримки та накладні витрати пам’яті в процесі роботи.

Розгорнутий зв’язний список (**unrolled list**) – зв’язний список, кожен елемент якого містить масив логічних елементів. Це дозволяє об’єднати переваги масивів та зв’язних списків. Об’єднання блоків логічних елементів у список дозволяє додавати нові елементи без зміни розміру блоку пам’яті, економити пам’ять на вказівниках та ефективніше використовувати кеш процесора завдяки послідовному розташуванню елементів. При послідовному заповненні списку гарантується, що незаповненим лишиться не більше одного блоку елементів.

Хеш-таблиця (hash map) – структура даних, у якій пошук елементу здійснюється на основі його ключа. На розташування елемента у хеш-таблиці вказує хеш-значення його ключа. Якщо декілька елементів мають однаковий хеш, то виникає колізія. Існує два методи розв’язання колізій – закрита та відкрита адресації. При закритій адресації кожен елемент таблиці – це зв’язний список, і усі елементи з однаковим хешем додаються до одного списку. Це є найпростішим способом розв’язання колізій, але вимагає використання додаткової пам’яті для вказівників і не дозволяє використовувати переваги кешування при обході елементів хеш-таблиці. При відкритій адресації у випадку колізії обирається нова позиція елементу. Нова позиція може обиратися як за допомогою додаткової хеш-функції, так і шляхом зміщення позиції на декілька елементів. Пошук елементу повторюється, доки не буде досягнуто порожнього запису у таблиці. Відкрита адресація використовує фіксований об’єм пам’яті і не потребує додаткових вказівників, але для ефективності операцій вставки і пошуку таблиця має бути заповнена не більш, ніж на 50%, тож це спричиняє додаткові витрати пам’яті.

B-дерево (b-tree) – структура даних представлена збалансованим та сильно розгалудженим деревом пошуку. Кожен вузол B-дерева, крім листків, є упорядкованим списком, у якому чергуються ключі і вказівники на потомків. Ключі вузла вказують інтервал, у якому знаходяться ключі потомку. **B+-дерево (B+-tree)** відрізняється тим, що воно зберігає усі значення у листкових вузлах, а листкові вузли мають посилання на сусіда, завдяки чому можна обійти усі значення без обходу всього дерева. Завдяки великій розгалуженості дерева підтримується його мала висота, що дозволяє переглядати невеликий об’єм даних за один прохід, а завдяки правилам побудови значення зберігаються у порядку зростання ключа.

Бінарні діаграми рішень (BDD) – економна форма представлення булевих функцій у вигляді орієнтованого ациклічного графу. Вершини графу представляють аргументи функції, листки – її двійкові значення.

Для додавання і вилучення ребер та зміни ваги ребер необхідно мати можливість редагувати дані графу. БДР дають можливість зберігати дані у стисненому вигляді та швидко отримувати значення функції за її параметрами, але редагування БДР вимагає складних обчислень. При представленні булевих функцій у формі БДР стало можливим розв'язувати багато проблем, які при традиційних представленнях структур нерозв'язні через значну розмірність таких представлень і складність операцій над ними.

Для проведення експериментів з порівняння ефективності застосування різних структур даних за затратами часу та пам'яті було розроблено програмну імітаційну модель спрощеної рекомендаційної системи, в якій було виділено три основні сутності – агент, сесія та предмет [1, 2]. Для формування рекомендацій було використано колаборативну фільтрацію типу user-based [3]. Експерименти проводилися на комп'ютері з процесором AMD Ryzen 5 3600 та 32 Гб оперативної пам'яті.

Програмна імітаційна модель рекомендаційнаої системи отримує наступні параметри: n_a – кількість агентів, n_s – кількість сесій, n_i – кількість предметів, n_{al} – максимальна кількість вподобань агента, n_{sl} – максимальний розмір сесії.

Рекомендаційна система у розробленій програмній імітаційній моделі працює за наступним алгоритмом:

1. Для кожного з n_a агентів випадковим чином генерується від 1 до n_{al} вподобань. При цьому унікальність вподобань не перевіряється, тому реальна кількість вподобань може виявитися менше.
2. Створюється n_s сесій. До кожної сесії закріплюється випадковим чином обраний агент. Потім серед вподобань цього агента випадковим чином обирається від 1 до $min(n_{al}, n_{sl})$ вподобань, які копіюються до сесії.
3. Випадковим чином обирається контрольна сесія, для якої буде сформовано рекомендацію.
4. Визначаються усі предмети, які належать до контрольної сесії.
5. Здійснюється пошук усіх сесій, вподобання яких мають перетин із вподобаннями контрольної сесії. На цьому етапі є можливість відфільтрувати сесії за розміром перетину.
6. Визначаються предмети, які буде рекомендовано. Здійснюється пошук усіх предметів, які належать хоча б одній з відібраних сесій, але не належать до контрольної сесії. На цьому етапі є можливість відфільтрувати предмети за кількістю закріплених сесій.

Найкращі результати показали методи зберігання даних з використанням розгорнутого та інвертованого розгорнутого з'язних списків. Тому було вирішено також провести додаткову серію

експериментів з цими структурами даних для різного розміру блоку списку. Розгорнутий список показав кращі результати за використовуваною пам'яттю в середньому в 1,54 рази та за часом генерації сесій в середньому в 1,68 разів. Інвертований розгорнутий список показав перевагу за часом генерації рекомендацій в середньому в 1,43 разів. Час генерації лайків обидва методи показали приблизно однаковим.

Література

1. Міхав В.В., Мелешко Є.В., Якименко М.С. (2020) Метод зберігання даних рекомендаційної системи на основі бінарних діаграм рішень. Системи управління, навігації та зв'язку. Збірник наукових праць. ПНТУ, Полтава. Т. 2 (60). С. 85-89.

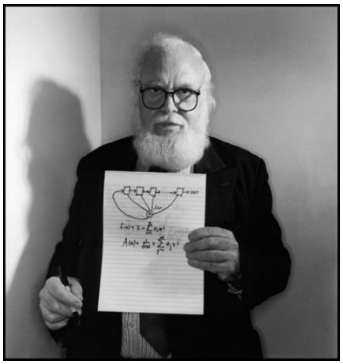
2.. Міхав В.В., Мелешко Є.В., Шимко С.В. (2021) Методи та структури даних для реалізації бази даних рекомендаційної системи соціальної мережі. Техніка в сільськогосподарському виробництві, галузеве машинобудування, автоматизація: збірник наукових праць Центрально-українського національного технічного університету. ЦНТУ, Кропивницький. Вип. 4(35). С. 8-16.

3. Recommender Systems Handbook (2010) Editors F. Ricci, L. Rokach, B. Shapira, P. B. Kantor, New York, NY, Springer-Verlag New York, Inc., USA. 842 p.

МОЖЛИВОСТІ ЗАСТОСУВАННЯ КОМБІНАТОРНИХ КОНФІГУРАЦІЙ ДЛЯ ОПТИМІЗАЦІЇ ІНФОРМАЦІЙНИХ СИСТЕМ

(ПАМ'ЯТІ С.ҐОЛОМБА)
Різник В.В.

Abstract. This report devotes to the 90- birthday of Solomon Wolf Golomb (1932-2016) - famous American mathematician, engineer, and professor of electrical engineering. The historical parallels of "Golomb rulers" and "Ideal Ring Bundles"(IRBs) as conceptual models for finding optimal solutions of scientific and technological problems examined. At present-day knowledge of information technologies as model of intelligent manifold system of axes, based on the idea of "perfect" combinatorial constructions with remarkable



properties is regarded. These properties provide competitive advantages of the system with respect to an ability to reproduce the maximum number of combinatorial varieties in the system, using IRBs, namely the concept of "perfect" relationships of complementary asymmetries implanted into rotational symmetry.

Доповідь присвячена пам'яті Соломона Вольф Ґоломба (31.05.1932-01.05.2016) – всесвітньо відомого американського вченого, спеціаліста в області телекомунікації, цифрової електроніки, кліткових автоматів і теорії інформації. Він був світовим лідером у застосуванні комбінаторної математики в теорії кодування та радіофізиці. С.Ґоломб був винахідником кодування Ґоломба, форми ентропійного кодування, лінійки Ґоломба, які використовуються в астрономії та шифруванні даних, також названі на його честь, як і один з основних методів генерації масивів Костаса, метод генерації Лемпеля-Ґоломба [1]. Лінійки Ґоломба вважаються «відкритими» В. Бебкоком у 1953 році [2], який досліджував інтермодуляційні спотворення в радіодіапазонах третього і п'ятого порядку, виявивши, що розміщення кожної пари каналів у частотному спектрі на певній відстані спотворення третього порядку було усунуто, а спотворення п'ятого порядку було значно зменшено. Гострий розум С.Ґоломба допоміг йому використати це явище для розробки математичного алгоритму, щоб поліпшити систему зв'язку і кодування, а його ентузіазм в області розважальної математики дозволив поширити цю ідею на популярні ігри. С. Ґоломб розгорнув ідею використання переваг багатобітових регістрів зсуву зі збалансованим числом 0 і 1, або 00, 01, 10, 11, виявивши в них відсутність автокореляції, що дало змогу вдосконалювати системи кодування - декодування сигналів з виправленням помилок за допомогою послідовностей, генерованих регістрами зсуву. С. Ґоломб застосував версії цих послідовностей (кодів Ріда - Соломона) для кодування відео зображень Марса, у коміркових телефонах CDMA (Code Division Multiple Access) з множинним доступом і кодовим розділенням каналів зв'язку. В 1956 році він поступив на роботу в аерокосмічну компанію Glenn L. Martin Company, яка згодом стала оборонним підрядником. Цим пояснюється вся його подальша наукова діяльність за умов строгої секретності, розробляючи військові та космічні комунікації. Продовжуючи

дослідження послідовностей на регістрах зсуву для вдосконалення систем радіо-керування ракетами, С. Ґоломб за допомогою 85-футової радіоантени Голстоун, розміщеної в пустелі Мохаве, уточнив дані про відстані Земля-Венера і Земля – Сонце. Генеровані регістрами зсуву послідовності дозволили уточнити відстань до Венери за допомогою системи радарів. У 1985 році С.Ґоломб отримав премію Шеннона Товариства теорії інформації IEEE, а згодом - медаль Агентства національної безпеки США. Він також був лауреатом медалі Ломоносова Російської академії наук і медалі Капіци Російської академії природничих наук. У 2000 році С.Ґоломб був нагороджений медаллю IEEE Річарда В. Хеммінга за його винятковий внесок в інформаційні науки та системотехніку. У 2016 році він був нагороджений медаллю Бенджаміна Франкліна в галузі електротехніки за новаторську роботу в галузі космічного зв'язку та обробки цифрових сигналів, безпечне пересилання даних, вдосконалення методів розшифрування криптографічних текстів, систем наведення ракет, стільникового зв'язку, радарів, гідролокаторів, GPS [1].

У 1996 році С. Ґоломб підтримав пропозицію Львівського державного політехнічного інституту про співпрацю з Південно - Каліфорнійським університетом (провідний дослідник проф. С. Ґоломб) за проектом ”Researches and Applications of the Combinatorial Configurations for Innovative Devices and Process Engineering“, який був надісланий до Фонду цивільних досліджень та розвитку США Civilian Research & Development Foundation (CRDF). Ця некомерційна організація була заснована в 1995 році Національним науковим фондом США згідно з рішенням Конгресу США з метою сприяння міжнародному науково-технічному співробітництву з наданням грантів, технічних ресурсів, проведення навчання для науковців та дослідників. Відбиток титульної сторінки цього проекту наведено на рис. 1. На цій сторінці С. Ґоломб власноручним підписом засвідчив дату свого дня народження 31.05.1932, а не 30.05.1932, як вказано у Вікіпедії [1]. С. Ґоломб відредагував текст проекту, мотивуючи переваги запропонованих комбінаторних конфігурацій з кільцевою топологічною структурою над ланцюговими послідовностями. Для порівняння обох конфігурацій в описі проекту були наведені формули для обчислення максимальної кількості способів послідовного додавання елементів. У лінійці і кільці з n елементами кількість всіх таких можливих способів визначається за формулами (1) і (2) відповідно:

$$C(n) = 1+ 2+ \ldots + n = n\,(n + 1)/2\,, \qquad (1)$$

$$S(n) = n\,(n - 1) + 1 \qquad (2)$$

Порівнюючи (1) і (2), легко бачити, що в кільцевій схемі кількість отриманих сум майже вдвічі більша для такого ж числа n елементів у ланцюговій послідовності.

Наприклад, ідеальне числове кільце (Ideal Numerical Ring) з п'яти (n=5) елементів генерує всі числа натурального ряду від 1 до 21 на кільцевій послідовності (1, 3, 10, 2, 5): 1=1, 2=2, 3=3, 4=1+3, 5=5, 6=5+1, 7=2+5, …, 21=1+3+10+2+5.

Якщо ці числа продовжувати додавати, обходячи кільце більше одного разу, числовий ряд можна продовжувати: 22=1+3+10+2+5+1, 23=2+5+1+3+10+2, і т.д.

В загальному випадку ідеальне числове кільце генерує кожне число натурального ряду від 1 до M точно R разів, за винятком значення M+1 (сума всіх n чисел):

$$M = n\,(n - 1) / R \qquad (3)$$

Наприклад, для n= 4 і R=2, M=6 існує ідеальне числове кільце (1, 1, 2, 3), для якої всі утворені суми є такими: 1, 1, 2, 3; 1 + 1 = 2, 1 + 2 = 3, 2 + 3 = 5, 3 + 1 = 4;

1 + 1 + 2 = 4, 1 + 2 + 3 = 6, 2 + 3 + 1 = 6, 3 + 1 + 1 = 5, 1 + 1 + 2 = 4, 1 + 1 + 2 + 3 = 7.

Тут кожна «кільцева сума» від 1 до 6 зустрічається точно двічі, а ця конфігурація описується параметрами n=4, R=2. Згідно висновку проекту запропоновані комбінаторні конфігурації окреслюють, по-суті, концептуально нові моделі технічних систем, властивості і структурна досконалість яких дає змогу відтворювати максимальну різноманітність системи з обмеженим числом елементів і зв'язків, що дає змогу поліпшувати якісні показники технічних пристроїв і технологічних процесів для їх застосування в різних галузях науки і передових технологіях, відтак досконалість, краса і гармонія існує не лише в абстрактних моделях, але також й у реальному світі [3].

За пропозицією С.Голомба українській групі дослідників у цьому проекті доручалися такі завдання: теоретичне дослідження ідеальних комбінаторних конфігурацій (ідеальних числових в'язанок), як модифікацій комбінаторних блок-схем (існування, перелік, класифікація); циклічні різницеві множини, циклічні групи полів Галуа; конструювання BIB-схем з використанням конфігурацій Ideal Numerical Ring (INR) та лінійок Ґоломба, а також зворотні задачі; програмне рішення побудови BIB-схем з використанням ІКВ; компіляція загального

каталогу INR та лінійок Ґоломба; розроблення схем для високопродуктивних інформаційних технологій кодування і модуляції; прикладні дослідження та інженерне проектування конкретних інноваційних пристроїв та технологій на основі комбінаторної теорії, і деякі проблеми гармонії та оптимізації систем.

COVER SHEET FOR COOPERATIVE GRANTS PROGRAM APPLICATION

I. INFORMATION TO BE PROVIDED BY EITHER THE U.S. OR FCU CO-INVESTIGATOR

A. Amound Requested from CRDF: Year One $24700 Year Two $21600 Total $46300
B. General Scientific Field of Proposed Activity: Mathematics, Systems Ingeneering
C. Brief descriptive title for your proposal:
Researches and Applications of the Combinatorial Configurations for Innovative Devices and Process Engineering

II. INFORMATION ON CO-INVESTIGATOR FROM THE FORMER SOVIET UNION
A. Name
Last **Riznyk** First **Volodymyr** Patronymic **Vasyljovych**

B. Sex M	C. Date of Birth (Month/Day/Year) May/21/1940

D. D.Sc., Professor
Field Combinatorics in Systems Engineering Year Awarded 1995
E. Office Phone +38(0322)398793, +38(0322)744300 Fax +380(322)744300
E-mail rvv@polynet.lviv.ua
F. Professional Title/Position Lvivska Polytechnika State University
Department of Control Systems Automation

G. Professional Affiliation and Address	H. Mailing Address(if different then G)
12,Bandera Str., 290646, Lviv-13, Ukraine	

I. Total number of FSU based investigators (including principal investigator):Three persons

Signature Date 17.02.96

III. INFORMATION ON U.S. CO-INVESTIGATOR
A. Name
Last **Golomb** First **Solomon** Middle **W.**

B. Sex M	C. Date of Birth (Month/Day/Year) 5/31/32

D. Highest Degree Earned Ph.D. Field Mathematics Year Awarded 1957
E. Office Phone (213)740-7333 Fax (213) 740-8729
E-mail milly@mizar.usc.edu
F. Professional Title/Position University of Southern California, "University Professor" (EE & Math.) University Park Campus

G. Professional Affiliation and Address	H. Malling Address (if different that G)
Los Angeles, CA 90089-2565, US	

I. Total number of U.S. based investigators (including principal investigator): one (1)

Signature Solomon W. Golomb Date March 5, 1996

Рис.1 *Титульна сторінка проекту "Researches and Applications of the Combinatorial Configurations for Innovative Devices and Process Engineering"*

Групі США належало здійснити теоретичне дослідження лінійок Ґоломба і їх зв'язок з різницевими множинами (існування, перелік, класифікація); конструювання різницевих множин і лінійок Ґоломба; розроблення загального каталогу лінійок Ґоломба; розроблення схем для високопродуктивних технологій для кодування інформації і комунікаційних сигналів; прикладні дослідження та інженерне проектування інноваційних пристроїв і технологій на основі комбінаторної теорії; використання комбінаторних розробок для отримання послідовностей з добрими кореляційними властивостями.

Для виконання цього проекту вимагалося поєднати досягнення теорії лінійок Ґоломба і теорії ідеальних в'язанок для розширення сфери практичних застосувань з метою досягнення комерціалізації продуктів.

Група українських дослідників повинна була розвинути основи теорії в'язанок як нову модифікацію комбінаторних конфігурацій на графах, включно з теорією лінійок Ґоломба, щоб запропонувати новий підхід до синтезу технічних пристроїв та перспективної інженерної технології. Група США повинна була розвинути основи теорії лінійок Ґоломба і запропонувати сфери можливих застосувань.

Група українських дослідників і група дослідників США повинні були координувати впровадження проекту щомісячно протягом двох років, використовуючи факс і електронну пошту.

Співпраця розпочалася з публікації ще в серпні 1996 року спільної статті «Комбінаторна теорія оптимізації перетворення сигналів векторних даних і опрацювання сигналів» з інтернаціональним складом авторів [4]. Однак вже восени того ж року С.Ґоломбу й мені надійшли листи від CRDF про відмову фінансової підтримки нашого проекту.

Історія винайдення ІКВ пов'язана з вирішенням інженерно-технічної задачі про розширення діапазону часових затримок в блоках запізнення, призначених для моделювання САР на електронних аналогових машинах. Математична задача полягала в закріпленні на роторі кільцевого комутатора 4-х контактних щіток з різними відносними кутовими зсувами в площині їх обертання з можливістю отримання якомога ширшого діапазону часових затримок на множині комбінаційних варіантів обрання відповідної пари щіток, одна з яких повинна підводити текуче значення напруги до запам'ятовувального елемента, а друга - зчитувати це саме значення з затримкою в часі після повороту ротора комутатора на відповідний кут [5].

Відомо, що на відміну від трьох наявних оптимальних і досконалих лінійок Ґоломба, серед яких нетривіальна – тільки єдина лінійка 4-го порядку довжиною 6 [6], існує апріорі нескінченно багато множин ідеальних кільцевих в'язанок (ІКВ), потужність сімей яких наближається апріорі до нескінченності зі збільшенням їх порядку n відповідно до значення функції Ейлера $\varphi(S)$ так, що вже для ІКВ 30-го порядку існує 132, для 270-го – 10800, для 984-го – точно 159630 різних варіантів послідовно впорядкованих чисел ІКВ, і т.д. [7], породжених просторовим полем обертової симетрії [8].

Таким чином, моделі оптимальних систем з кільцевою структурою вигідно відрізняються від лінійок Ґоломба нескінченно великим числом елементів досконало організованих ідеальних комбінаторних конфігурацій, багатоманітністю інваріантів, унікальністю ансамблів тонкої структури ІКВ, а також можливістю взаємно однозначного перетворення одновимірних ІКВ в ідеальні конфігурації вищих вимірів. Фундаментальні та прикладні дослідження векторних комбінаторних конфігурацій доцільно продовжувати ґрунтуючись на реально існуючому законі «елегантного» розбиття обертової симетрії на дві асиметричні частини, що становить інформаційний базис для розбудови взаємопов'язаних обертовою симетрією S–го порядку комплементарних ІКВ, породжених цією симетрією. У свою чергу, кожна з цих частин становить базис для конструювання векторних ІКВ, перетворюючись у множини сімей дво- чи багатовимірних просторових комбінаторних конфігурацій у вигляді тору чи многовиду відповідної розмірності зі заданими розмірами сітки координат, яка окриває їхню поверхню з спільною точкою відліку [8]. Опрацювання масивів даних у таких системах координат одночасно за кількома категоріями атрибутів відкриває перспективи для розвитку векторних інформаційних технологій. Зі збільшенням порядку обертової симетрії стрімко зростає кількість породжених цією симетрією елегантних ансамблів просторових комбінаторних конфігурацій, наповнюючи грандіозною красою, досконалістю і гармонією чарівний світ багатовимірної комбінаторики. Теоретично існує нескінченно багато базисів, які породжують численні різновиди множин координат багатовимірних просторів, що відкриває перспективи для розвитку векторних інформаційних технологій на основі мінімізації базової структури систем опрацювання потоків інформації [9].

Література

1. https://uk.wikisko.ru/wiki/Golomb_coding

2. Babcock, W.C. (1953). Intermodulation interference in radio systems, Bell Systems Technical Journal, 63-73

3. Riznyk, V.V., & Golomb, S.W. (1996). Researches and Applications of the Combinatorial Configurations for Innovative Devices and Process Engineering / V.V. Riznyk // *CRDF Cooperative Grants Program*, Los Angeles, CA 90089-2565, US, March 5, 1996, 10 p.

4. S.Golomb, P.Osmera, V.Riznyk. Combinatorial Sequencing Theory for Optimisation of Signal Data Vectors Converting and Signal Processing. Proc.All-European Workshop on Design Methodologies for Signal Processing, Zakopane, Poland, 1996, pp.43-44.

5. Ризнык В.В. Запоминающее устройство. А.с. 347788 СССР, № 1618266/18-24,заявл. 25.01.1971, опубл. 10.08.1972, бюл. № 24, 1972.

6. https://www.wiki.uk- ua.nina.az/ Лінійка_Голомба.html

7. Різник В.В. Синтез оптимальних комбінаторних систем: монографія. Львів. «Вища школа». 1989. 168 с.

8. Різник В.В. Комбінаторна оптимізація багатовимірних систем. Моделі багатовимірних систем інтелекту: монографія. Львів. Видавництво Львівської політехніки. 2019. 168 с.

9. Різник В.В., Талан М.В. Спосіб опрацювання інформації. Патент України 119781 від 12.08.2019, бюл. № 25, 2019.

АЛГОРИТМ ПРЫГАЮЩИХ ЛЯГУШЕК ДЛЯ ЗАДАЧИ ПРЯМОУГОЛЬНОГО РАСКРОЯ

Сардак О.В., Сабиров З.Р.
Запорожский национальный университет, Украина
Ургенчский государственный университет, Узбекистан

JUMPING FROG ALGORITHM FOR THE PROBLEM OF RECTANGULAR CUTTING

Abstract.. The problem of integer packing of rectangles into a rectangular matrix is considered. It is shown that this problem can be considered as an optimization problem on a fragmentary structure. A variant of the jumping frogs algorithm for this problem is proposed.

Задача целочисленного прямоугольного раскроя является одной из классических задач дискретной оптимизации. Задача относится к числу NP-трудных задач. Поэтому для этой задачи оправдано применение разного рода метаэвристик. Существует множество методов поиска

субоптимальных решений этой задачи на основе различных подходов [1]. Рассмотрим метод, который является комбинацией фрагментарного алгоритма и алгоритма перемешанных прыгающих лягушек.

Фрагментарной структурой [2] (X, E) на конечном множестве *X* [2] называется семейство его подмножеств $E = \{E_1, E_2, ..., E_n\}$ такое, что $\forall E_i \in E,\ E \neq \varnothing \quad \exists e \in E_i : E_i \setminus \{e\} \in E$. Элементы из множества *E* будем называть допустимыми фрагментами.

Таким образом, для любого допустимого фрагмента E_i существует нумерация его элементов $E_i = \{e_{i1}, e_{i2}, ...e_{is_i}\}$ такая, что $\forall k = 1, 2, ...s_i \quad \{e_{i1}, e_{i2}, ...e_{ik}\} \in E$. Одноэлементные множества, которые являются допустимыми фрагментами, будем называть элементарными фрагментами. Фрагмент, будем называть максимальным, если он не является подмножеством никакого другого фрагмента.

Максимальный фрагмент может быть построен с помощью следующего "жадного" (фрагментарного) алгоритма:

а) элементы множества *X* линейно упорядочиваются;

б) на начальном шаге выбирается пустое множество $X_0 = \varnothing$;

в) на шаге с номером *k* + 1 выбирается первый по порядку элемент $x \in X \setminus X_k$, такой, что $X_k \bigcup \{x\} \in E$;

г) алгоритм заканчивает работу, если на очередном шаге не удалось найти элемент $x \in X \setminus X_k$ с требуемым свойством.

Результат применения фрагментарного алгоритма определяется заданным линейным порядком на множестве *X*. Таким образом, любой максимальный фрагмент может быть описан некоторой перестановкой элементов множества *X*.

Задачу целочисленного прямоугольного раскроя будем рассматривать в следующей постановке:

Задана прямоугольная матрица-основа размерности $H \times W$, где *H*,*W* – целые положительные числа – высота и ширина матрицы основы. Задано множество прямоугольных заготовок, стороны которых также выражаются целыми числами. Требуется разместить прямоугольные заготовки на матрице-основе таким образом, чтобы площадь части основы, не занятой заготовками была минимальной. Заготовки могут пересекаться лишь по границе. Пример такого размещения приведен на рис1. Серым цветом выделена незанятая часть основы. Пусть множество

заготовок задано в виде набора пар $\{(h_i, w_i)\}$, где h_i, w_i - высота и ширина соответствующей заготовки, i=1,2,,,N, где N – число заготовок. Допустимое решение задачи можно представить набором троек $\{(\delta_i, t_i, l_i)\}_{i=1}^{N}$. Здесь

$$\delta_i = \begin{cases} 1, \text{ если } i\text{-я заготовка входит в покрытие} \\ 0, \text{ если } i\text{-я заготовка не входит в покрытие} \end{cases},$$

t_i,l_i – соответственно координаты левого верхнего угла заготовки относительно матрицы основы.

Задача плоского прямоугольного раскроя может быть сформулирована как задача оптимизации на фрагментарной структуре. Каждый элементарный фрагмент представляется заготовкой и координатами ее левого верхнего угла на матрице основе. Условие присоединения – размещаемая на матрице основе заготовка не принадлежит к множеству уже выбранных заготовок и не пересекается с уже размещенными на матрице заготовками.

Таким образом, задача прямоугольного раскроя может быть сведена к задаче оптимизации на перестановках элементарных фрагментов.

Существуют примеры, когда метод размещения заготовок в свободном верхнем левом угле матрицы основы не приводит к оптимальному решению при любом выборе перестановки элементарных фрагментов. Однако этот недостаток можно исправить путем добавления к множеству фрагментов нескольких виртуальных клеток [3].

Для поиска субоптимальных решений задачи был применен вариант перемешанных прыгающих лягушек на перестановках [4].

Результаты численных экспериментов на большой базе тестовых примеров показали, что такой подход может эффективно использоваться как эвристический алгоритм при решении оптимизационных задач целочисленного прямоугольного раскроя. Причем качество алгоритма будет возрастать при увеличении ряда параметров метода прыгающих лягушек таких как, величина популяции, число групп, количество прыжков.

Литература

1. Lodi A., Martello S., Monaci M. Two-dimensional packing problems: *A survey. European Journal of Operation Research*. 2002. Vol. 141. P. 241-252.

2. 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

3. Kozin, I.V., Batovskyi, S.E. Fragmentary Structures in a Two-Dimensional Strip Packing Problem. *Cybern Syst Anal* **55,** 943–948 (2019) doi:10.1007/s10559-019-00204-w

4. Narimani M.R. (2011) A New Modified Shuffle Frog Leaping Algorithm for NonSmooth Economic Dispath. World Applied Sciences Journal. P. 803–814

ПРО ОДИН ЗІ СПОСОБІВ ГЕНЕРАЦІЇ ГРАЦІОЗНИХ ДЕРЕВ З ДЕРЕВ МЕНШИХ ПОРЯДКІВ

Семенюта М.Ф.

Льотна академія Національного авіаційного університету

ABOUT ONE OF THE WAYS OF GENERATION OF GRACIOUS TREES FROM TREES OF SMALLER ORDERS

Abstract. We have presented a new method of constructing a graceful tree from isomorphic trees of smaller orders.

Засновником теорії розміток вважається А. Роса, який в 1967 році запропонував декілька типів розміток, як інструмент для розкладу повного графа на ізоморфні підграфи. За допомогою цих розміток досягнуто значних результатів у розв'язку задач існування, побудови, переліку розкладів графів. А. Роса прийшов до гіпотези, що всі дерева граціозні [1]. Ця гипотеза відкрита і надихає на створення нових методів побудови граціозних дерев.

Складність побудови граціозної розмітки довільного графа полягає в тому, що не кожній його вершині можна поставити у відповідність мітку 0. Інакше багато класів графів породжувалися б індуктивно з графів менших порядків. Це призвело до виникненню нетривіальних методів генерації граціозних графів з графів менших порядків.

В даній роботі представлено модифікований методом Δ-побудови граціозної розмітки для певного класа дерев, оснований на методі, запропанованому у [2].

Нехай T –дерево порядку p.

1. Побудуємо ліс: $nT=\bigcup_{i=1}^{n} T_i$, де T_i – ізоморфні копії дерева T, i=1,2, …, n.

2. Обираємо довільним чином вершину v в дереві T. Вершина v_i в дереві T_i є ізоморфним образом v з T.

2. Додаємо до графа nT, ребра v_1v_2, v_1v_3, …, v_1v_n. Новий граф $\bigcup_{i=1}^{n} T_i$ $+v_1v_2+v_1v_3+\ldots+v_1v_n$ позначимо T^*, він є деревом порядку np.

Опишемо метод знаходження вершинних міток для граціозної розмітки T^*.

1. Нехай дерево T порядку p має граціозну розмітку f з $f(v)=p-1$, де $v\in V(T)$. Розглядаємо T як двочастковий граф. Для цього виконаємо розбиття множини його вершин на дві підмножини A, B наступним чином:

1.1) знайдемо відстань від вершини v до всіх інших вершин u_s дерева T;

1.2) якщо $d(u_s)$ – дорівнює нулю або парне число, то $u_s\in A$, інакше $u_s\in B$.

2. Побудуємо ліс $nT=\bigcup_{i=1}^{n} T_i$ з дерев T_i, що є ізоморфними образами дерева T, де i=1,2, …, n. Вершини u_s^i, $v_i\in V(T^*)$ є ізоморфними образами вершин u_s, $v\in V(T)$, відповідно, і $u_s^i\neq v_i$ для будь-яких i=1,2, …, n, $s\in\{1, 2, \ldots, p\}$.

3. Для міток вершин $\bigcup_{i=1}^{n} T_i$ використаємо розмітку φ, що визначається з формул:

$$\varphi(u_s^i)=\begin{cases} i\cdot p-1-f(u_s), & якщо\ u_s\in A,\\ (n+1-i)p-1-f(u_s), & якщо\ u_s\in B,\end{cases} \qquad (1)$$

$$\varphi(v_i)=(i-1)p. \qquad (2)$$

4. Згідно теореми 2 [3] φ є граціозною розміткою дерева $T^*=\bigcup_{i=1}^{n} T_i$ $+v_1v_2+v_1v_3+\ldots+v_1v_n$.

Роботу наведеного методу продемонстровано в прикладі 1.

Приклад 1. Розглянемо дерево S, зображене на рис. 1 з граціозною розміткою f, у якого $f(v) = p - 1 = 6$. Вершини ототожнюємо з їх мітками. Граф $\bigcup_{i=1}^{3} S_i$ є диз'юнктивним об'єднанням ізоморфних образів S_i дерева S. Нехай вершина $v_i \in V(S_i)$ є ізоморфним образом $v \in V(S)$, *i*=1, 2, 3.

Побудуємо граф $S^* = \bigcup_{i=1}^{3} S_i + v_1v_2 + v_2v_3 + v_3v_4$ (рис. 1). Його граціозна розмітка знайдена з формул 1, 2 і представлена на рис. 1.

Отримані результати розширюють клас граціозних графів.

Література

1. 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.
2. Stanton R., Zarnke C. Labeling of balanced trees. // Proc. 4th Southeast Conf. Combin., Graph Theory, Comput. - 1973. - P. 479 - 495.
3. Семенюта М.Ф. Частинні випадки задачі граціозності графів. Комп'ютерна математика. 2015. №2. С. 96 - 102.

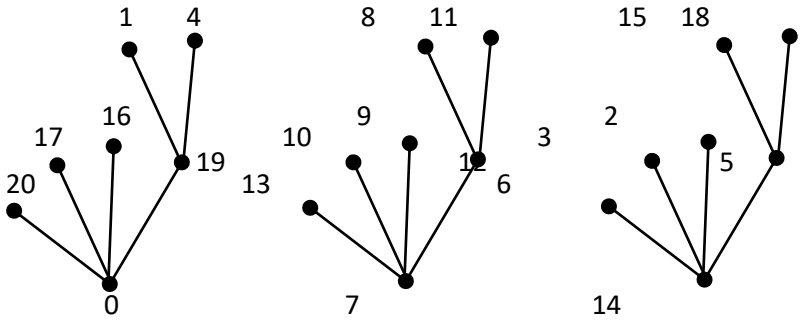


Рис.1. Граціозна розмітка дерева S^*

НЕЙРОМЕРЕЖЕВА МОДЕЛЬ ДЛЯ СУМІСНОГО ПРИЙНЯТТЯ РІШЕНЬ АВІАЦІЙНИМ ПЕРСОНАЛОМ ПРИ ВИБОРІ ЗАПАСНОГО АЕРОДРОМУ В ПОЗАШТАТНІЙ СИТУАЦІЇ

Сікірда Ю.В., Шмельова Т.Ф.
Льотна академія Національного авіаційного університету
Національний авіаційний університет

NETWORK MODEL FOR COLLABORATIVE DECISION MAKING BY THE AVIATION PERSONNEL DURING CHOOSING AN ALTERNATE AERODROME IN EMERGENCY

Abstract. For improving Collaborative Decision Making by the aviation personnel during choosing an alternate aerodrome in emergency, a five-layer recurrent neural network with biases was developed. The use of neural network will ensure the synchronization of the aviation specialists' decisions, the prompt exchange of correct information between them, the effective balance between safety and cost in consolidated decisions.

Вступ. Під час планування та виконання польотів забезпеченням безпеки займаються такі фахівці, як льотні диспетчери, льотні екіпажі, авіадиспетчери, обслуговуючий персонал, наземний персонал тощо. Кожен з них відіграє важливу роль на різних етапах, оскільки безпечний політ починається не тільки з вильоту літаків. Авіаційні спеціалісти суворо дотримуються посібників та правових документів, затверджених у сфері їх професійної діяльності. Дуже часто складність, зміст, особливості документів, які регулюють діяльність кожного авіаційного фахівця, різні, що не дозволяє розробити загальний алгоритм дій для всього авіаційного персоналу для конкретних умов, особливо в позаштатних ситуаціях, коли мають місце невизначеність, брак інформації та часу на прийняття рішень (ПР). Тому виникає конфлікт між рішеннями та діями залученого персоналу, який спільно приймає рішення [1].

Враховуючи взаємодію авіаційного персоналу, необхідно розуміти його ролі та обов'язки. Льотний диспетчер відповідає за планування польоту, пілот безпосередньо відповідає за безпечне виконання цього польоту, авіадиспетчер забезпечує обслуговування повітряного руху, інформування та допомогу екіпажам. Тому, приймаючи рішення, вони аналізують загальну групу факторів, оскільки основною метою є виконання завдання (політ з пункту А в пункт В), але з різних точок зору. Але остаточне рішення в польоті приймає пілот. Для раціонального сумісного прийняття рішень (Collaborative Decision Making – CDM)

кожний оператор повинен проаналізувати та врахувати поточну ситуацію. Тому актуальним завданням є розробка сучасних методів та адекватних моделей для удосконалення CDM авіаційним персоналом в позаштатних ситуаціях, застосування яких у складі інтелектуальних систем підтримки ПР дозволить підвищити ефективність його діяльності в умовах неповноти та невизначеності інформації. Авторами розроблено об'єктивно-суб'єктивний метод сумісного ПР у разі невизначеності та ризику та інтеграції стохастичних і детермінованих моделей ПР при неоднозначності початкових рішень учасників [2].

Пропонується нейромережева модель для отримання спільного, інтегрованого рішення всіх учасників, наприклад таких, як пілот, льотний диспетчер та авіадиспетчер, на прийнятному рівні ефективності у разі ПР в позаштатних ситуаціях.

Основна частина. Для удосконалення CDM авіаційним персоналом при виборі запасного аеродрому (ЗА) в позаштатній ситуації (наприклад, неспроможності продовжити політ до аеродрому призначення через погіршення погодних умов за маршрутом) розроблено багатошарову рекурентну штучну нейронну мережу (ШНМ) зі зсувами, яка має п'ять шарів, три з яких сховані (рис. 1). Багатошарова ШНС може апроксимувати будь-яку функціональну залежність за рахунок схованих шарів нейронів та здатна до навчання. Динамічність рекурентних мереж є важливою властивістю для складної соціотехнічної аеронавігаційної системи, так як завдяки зворотнім зв'язкам змінюються входи нейронів, що призводить до зміни стану ШНМ [3].

Перший шар ШНМ (вхідний). Входи $s_1, s_2, \ldots, s_5$ відповідають виду польотної ситуації, яка потребує відходу повітряного судна на ЗА/посадковий майданчик ($\overline{S}$): нормальна, ускладнена, складна, аварійна, катастрофічна.

Другий, третій та четвертий шари ШНМ (сховані). Оцінки впливу об'єктивних факторів на ПР щодо вибору ЗА пілотом ($\overline{F}$), авіадиспетчером ($\overline{L}$) та льотним диспетчером ($\overline{\lambda}$) відповідно, $f, l, \lambda = \overline{1,8}$: f_1, l_1, λ_1 – запас палива на борту літака (завжди контрольований); f_2, l_2, λ_2 – метеорологічна ситуація (на аеродромах вильоту, призначення, запасних та по маршруту); f_3, l_3, λ_3 – можливості літака (наявне обладнання на борту, особливості MEL, існуючі експлуатаційні обмеження); f_4, l_4, λ_4 – можливості аеродрому (радіо- та світлотехнічні системи заходу на посадку, технічні характеристики злітно-посадкових смуг та руліжних доріжок, обмеження робочого часу, категорія аеродрому, категорії пожежної та пошуково-рятувальної служби, аварійних служб); f_5, l_5, λ_5 – можливості екіпажу (мінімуми екіпажу, час роботи екіпажу); f_6, l_6, λ_6 – розташування перешкод у

секторах заходу, відходу на друге коло та вильоту; f_7, l_7, λ_7 – повітряна обстановка (інтенсивність сектору управління повітряним рухом, перевантаження радіочастот); f_8, l_8, $\lambda 8$ – комерційні фактори (аеропортові збори, відстань від аеродрому призначення, наявність контрактів з хендлінговими службами, наявність митної служби, служби прикордонного та міграційного контролю тощо).

Додатковий вхід Bias $\overline{\theta}$ характеризує вплив суб'єктивних факторів.

П'ятий шар ШНМ (вихідний). Результати $u_1, u_2, \ldots, u_5$ сумісного ПР щодо вибору ЗА пілотом, авіадиспетчером, льотним диспетчером (потенційний збиток/очікуваний ризик) $(\overline{U})$: дуже малий, малий, середній, великий, дуже великий потенційний збиток.

Вихідні вектори другого, третього та четвертого шарів ШНМ (1):

$$\overline{F},\ \overline{L},\ \overline{\lambda} = f\left(\overline{\text{net}} - \overline{\theta}\right) = f(\overline{W}\{\overline{S},\ \overline{F},\ \overline{L}\} - \overline{\theta}), \qquad (1)$$

де $\overline{W}$ – вагові коефіцієнти $\overline{W} = \{w_{ij}\}$; $\overline{\theta}$ – зсув оцінок об'єктивних факторів завдяки впливу суб'єктивних факторів.

Вихідний вектор п'ятого (вихідного) шару (2):

$$\overline{U} = f(\overline{W}, \overline{\lambda}). \qquad (2)$$

Задаються наступні вихідні сигнали векторів шарів нейронів (3):

$$\overline{F}, \overline{L}, \overline{\lambda}, \overline{U} = \begin{cases} 1; if\ f\left(\overline{W}\{\overline{S}, \overline{F}, \overline{L}\} - \overline{\theta}\right), f\left(\overline{W}, \overline{\lambda}\right) > 0 \\ 0; if\ f\left(\overline{W}\{\overline{S}, \overline{F}, \overline{L}\} - \overline{\theta}\right), f\left(\overline{W}, \overline{\lambda}\right) \le 0 \end{cases}, \qquad (3)$$

де f – нелінійна функція активації.

За допомогою нейропакету NeuroSolutions версії 7.1.1.1 (розробка NeuroDimension, Inc.) був побудований та навчений з вчителем процедурою оберненого поширення помилки багатошаровий прямонаправлений персептрон зі зсувами для CDM авіаційним персоналом при виборі ЗА в позаштатній ситуації.

Вхідні, проміжні та вихідні компоненти ШНМ задано згідно статистичним даним за попередній 10-річний період та експертним оцінкам авіаційних фахівців, додаткові входи – зсуви $\overline{\theta}$ умовно прийнято рівними одиниці. Навчання ШНМ відбувалось шляхом модифікації вагових коефіцієнтів зв'язків між нейронами до моменту, коли помилка досягає мінімального значення й перестає зменшуватись. Достатнім виявилось 1000 циклів навчання.

Тестування ШНМ на прикладах, які не входили до навчальної вибірки, показало високу точність визначення збитку (помилка Δ між дійсною та отриманою за допомогою нейромережі оцінкою потенційного

збитку складає не більше 5% від діапазону зміни його значень), що підтверджує достовірність запропонованої моделі.

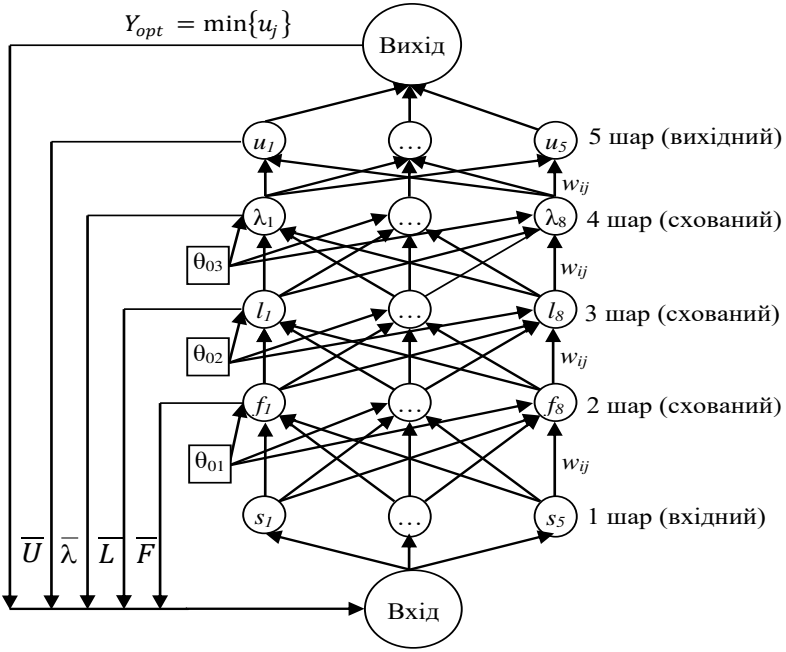


Рис. 1. *Нейромережева модель для CDM авіаційним персоналом при виборі ЗА в позаштатній ситуації*

Висновок. У позаштатних ситуаціях важливо забезпечити можливість спільного, інтегрованого рішення всіх учасників, наприклад таких, як пілот, льотний диспетчер та авіадиспетчер, на прийнятному рівні ефективності. Це досягається повнотою і точністю наявної інформації, а також злагодженою взаємодією авіаційного персоналу, його чітким і правильним розумінням службових обов'язків та своєї ролі в процесі виконання спільного завдання. Використання ШНМ у невизначених (позаштатних) ситуаціях забезпечить синхронізацію CDM авіаційними фахівцями, оперативний обмін коректною інформацією між ними, ефективне балансування між безпекою та вартістю в консолідованих рішеннях.

Література

1. Socio-Technical Decision Support in Air Navigation Systems: Emerging Research and Opportunities / Eds. T. Shmelova, Yu. Sikirda, N. Rizun, A.-B. M. Salem, Yu. Kovalyov. USA, Hershey: IGI Global Publ., 2018. 305 p.

2. Shmelova T., Sikirda Yu. Collaborative Decision-Making Models for UAV Operator's Intelligent Decision Support System in Emergencies. *Artificial Intelligence and Information Systems (ICAIIS 2021)*: Proceedings of the 2nd International Conference, Chongqing, China, May 28-30, 2021. Beijing, China: Association for Computing Machinery (ACM), 2021. P. 1–7.

3. Kumar S. Neural networks – a classroom approach. 2nd ed. U.S., New York: McGraw Hill Education, 2017. 730 p.

ОПТИМАЛЬНА ТЕХНОЛОГІЯ ПОБУДОВИ ХМАРНОЇ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНОЇ СИСТЕМИ АВТОМАТИЗАЦІЇ ВИРОБНИЧИХ ПРОЦЕСІВ

Смірнова Т.В., Смірнов С.А., Смірнов О.А.
Центральноукраїнський національний технічний університет

OPTIMAL TECHNOLOGY OF CONSTRUCTION OF CLOUD INFORMATION AND COMMUNICATION SYSTEM OF AUTOMATION OF PRODUCTION PROCESSES

Abstract. This work is devoted to the improvement of existing and development of new methods of building a modern cloud information and communication infrastructure of enterprises in order to further use it to automate production processes

Науково обґрунтоване планування й оптимізація інформаційних мереж, які забезпечують надання запитуваних послуг із заданими показниками ефективності функціонування, є дуже складною науково-технічною й економічною проблемою, без вирішення якої неможливе створення хмарної інформаційної інфраструктури підприємства, що відповідає всім потребам та сформованим вимогам [1-3]. Тому, дана робота присвячена саме удосконаленню існуючих та розробці нових методів побудови сучасної хмарної інформаційно-комунікаційної інфраструктури підприємств з метою її подальшого використання для автоматизації виробничих процесів.

Ефективність інформаційно-комунікаційної мережі – це її здатність досягати поставленої мети у заданих умовах функціонування з певним рівнем якості, тобто це характеристика, що відображає ступінь

відповідності мережі своєму призначенню, технічну досконаліть та економічну доцільність [4]. Поняття ефективності пов'язані з отриманням деякого корисного результату – ефекту використання інформаційно-комунікаційних мереж, який досягається ціною витрат певних ресурсів, тому ефективність мережі можна розглядати як співвідношення між ефектом (виграшем) і витратами.

Будемо визначати показник ефективності інформаційної мережі процесом її функціонування та відповідно функціоналом від цього процесу. Загалом $W = W(t, L_к, L_{ТП}, L_А, L_Д, L_У)$, де W – множина показників ефективності мережі, t – час; $L_к$, $L_{ТП}$, $L_А$, $L_Д$, $L_У$ – множина параметрів відповідно до вхідних потоків запитів на обслуговування користувачів ($L_к$), технічних та програмних засобів мережі ($L_{ТП}$), алгоритмів обробки та передачі інформації в мережі ($L_А$), діяльності користувачів ($L_Д$), умов функціонування мережі ($L_У$). В свою чергу – $L_Д = \{L_Т, L_{ТМ}, L_Н\}$, де $L_Т$, $L_{ТМ}$, $L_Н$ – множина вихідних показників діяльності вузлів інформаційної мережі відповідно до точнісних ($L_Т$), часових ($L_{ТМ}$), надійнісних ($L_Н$). Значення компонентів множин $L_Т$, $L_{ТМ}$, $L_Н$ визначаються конкретними процесами діяльності вузлів у розглянутій інформаційно-комунікаційній мережі, засобами, які є в їхньому розпорядженні для виконання своїх функцій та умовами роботи.

Відповідно до конкретизації поняття ефективності, показники множини можна розділити на три групи: $W = \{W_Ц, W_Т, W_Е\}$, де $W_Ц$ – показники цільової ефективності інформаційно-комунікаційної мережі або ефективності використання (цільового застосування) цієї мережі, це кількісний показник відповідності мережі своєму призначенню; $W_Т$ – показники технічної ефективності інформаційно-комунікаційної мережі, це кількісна міра, що відображає технічну досконаліть мережі; $W_Е$ – показники економічної ефективності інформаційно-комунікаційної мережі, це кількісний показник економічної доцільності розгортання мережі.

Виходячи з результатів багатокритеріального аналізу показників ефективності оцінюваних технологій, можна зробити висновки про те, що сфера використання технологій сучасних безпроводових мереж є досить фрагментованою. При цьому, лідируючу позицію посідає технологія 5G, але жодна технологія підключення не має повсюдне покриття та здатна вирішити всі проблеми для всіх потенційних вертикальних випадків використання. Кожен варіант підключення до мережі передбачає компроміс між споживаною потужністю, діапазоном та смугою пропускання.

В подальшому технологія 5G, яка націлена на реалізацію додатків з низькою затримкою і високою пропускною спроможністю,

становитимуть більшу частину величезного простору IoT, очолюваного операторами мобільного зв'язку, операторами неліцензованого спектру та підприємствами через власні приватні мережі.

При цьому, однією з ключових технологій, що може бути використана в мережах 5G для забезпечення виробничих процесів є MEC (Mobile Edge Computing). MEC [4] передбачає розміщення IT-функціоналу та пов'язаного функціоналу хмарних обчислень.

Периферійні сервери можуть підвищити продуктивність та зменшити затримку в будь-якому контексті, максимально наблизивши обчислювальні та інші ресурси до місця, де вони потрібні. Оскільки 5G – це висока швидкість та низька затримка, ці дві технології добре поєднуються. Більше того, взаємозв'язок між ними набагато глибший: широка доступність мереж 5G, величезна кількість обладнання та ПЗ, яке підключатиметься до цих мереж, не обійдуться без MEC.

У поєднанні з архітектурою периферійних обчислень 5G зазвичай розглядається як технологія, яка вирішить одну з найнагальніших проблем мережної затримки при одночасному забезпеченні високої пропускної здатності.

Таким чином, застосувавши багатокритеріальний аналіз до даних, а також проаналізувавши потреби підприємств, на яких необхідно проводити підтримку процесів автоматизації виробничих процесів, можна зробити висновок, що оптимальним рішенням для великих підприємств є розгортання мережі 5G.

Кожне виробниче підприємство повинно очікувати отримання максимального ефекту від вкладання власних коштів у свої власні проекти інформатизації. Величину цього ефекту можна відобразити на величині прибутку підприємства від реалізації проекту інформатизації. Виходячи із цих тверджень, можна проводити вибір оптимального проектного рішення, використовуючи критерій оптимальності Байеса-Лапласа (BL-критерій). Доцільно також обґрунтувати використання, як оціночної функції оптимальності – чистого приведеного ефекту за проектом інформатизації, як інтегрального показника ефективності проектних рішень.

За аналізом результатів можна зробити висновок про необхідність розгортання на підприємствах саме мікростільникової структури мережі із використанням радіочастот у діапазонах 700, 800 або 2600 МГц.

Висновки. Дослідження, проведені в даній роботі дозволили провести аналіз та здійснити відповідний вибір оптимальної технології для використання її з метою оптимізації виробничих процесів. Для цього було розроблено модель багатокритеріальної оптимізації, визначено пул комунікаційних технологій, які можуть бути використані для оптимізації

виробничих процесів. В результаті, проведений багатокритеріальний аналіз дозволив впевнитись в доцільності розгортання стільникових мереж 5G для автоматизації виробничих процесів. Також було запропоновано методику вибору оптимальних проектних рішень технологій мобільного зв'язку 5G.

Література

1. Смірнова Т.В., Солових Є.К., Смірнов О.А., Дрєєв О.М., «Побудова хмарних інформаційних технологій оптимізації технологічного процесу відновлення та зміцнення поверхонь деталей», Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 184-194, 2019.

2. Смірнова, Т.В., Смірнов, С.А., Минайленко, Р.М., Доренський, О.П., Сисоєнко С.В. «Хмарна автоматизована система інтелектуальної підтримки прийняття рішень для технологічних процесів». Вісник Черкаського державного технологічного університету. Технічні науки. №4, 2020, С. 84-92.

3. Смірнова Т.В., Столяренко М.П., Янков М.О., Грудік В.В., Моторін Ю.Ю. «Модель реалізації структури технологічного процесу у хмарному сервісі». Збірник наукових праць Харківського національного університету Повітряних Сил. 2021. № 4(70). С. 132-142.

4.https://www.it.ua/knowledge-base/technology-innovation/internet-veschej-internet-of-things-iot

ПРО ВИКОРИСТАННЯ СОЛВЕРА BARON ДЛЯ ЗАДАЧІ ЗБАЛАНСОВАНОЇ УПАКОВКИ КРУГІВ[1]

[1]Стецюк П.І., [2]Романова Т.Є, [1]Хом'як О.М.

[1]Інститут кібернетики імені В.М. Глушкова НАН України

[2]Інститут проблем машинобудування імені А.М. Підгорного НАН України

ON USING BARON SOLVER FOR BALANCED CIRCULAR PACKING PROBLEM

Abstract. The paper addresses a packing problem of unequal circles in a larger circle of minimum radius so that the gravity center of the family of

[1] Робота підтримана Volkswagen Foundation (грант № 97775)

circles coincides with the outer circle center. Two formulations of the balanced circular packing problem as multiextremum nonlinear programming models are provided. The first formulation uses quadratic constraints while the second one employs radical functions for describing the non-overlapping of circles and their containment into the outer circle. The global minima of both problems were found by solver BARON for the benchmark instance with five unequal circles.

Нехай задано сімейство кругів S_i з радіусами r_i і вагами w_i, $i = 1,\ldots,m$. Центр тяжіння круга S_i знаходиться в його центрі. Збалансованою упаковкою сімейства кругів S_i, $i = 1,\ldots,m$, в круг S називають таку їх упаковку, щоб радіус круга S був мінімальним і центр тяжіння сімейства кругів S_i, $i = 1,\ldots,m$, збігався з центром зовнішнього круга S, який знаходиться в початку системи координат. Позначимо (x_i, y_i) – невідомий центр круга S_i, r – невідомий радіус зовнішнього круга S. Зокрема визначимо $\lambda_i = w_i / \sum_{j=1}^{m} w_j$, $i = 1,\ldots,m$, $r_{low} = \max\limits_{i=1,\ldots,m} r_i$.

Задачі збалансованої упаковки сімейства кругів S_i, $i = 1,\ldots,m$, відповідають дві багатоекстремальні задачі нелінійного програмування [1, 2]. Перша з них (задача **A**) має такий вигляд:

$$r^* = \min_{r,x,y} r \qquad (1a)$$

при обмеженнях

$$x_i^2 + y_i^2 \le (r - r_i)^2, \quad i = 1,\ldots,m, \qquad (2a)$$

$$(x_i - x_j)^2 + (y_i - y_j)^2 \ge (r_i + r_j)^2, \quad 1 \le i < j \le m, \qquad (3a)$$

$$\sum_{i=1}^{m} \lambda_i x_i = 0, \quad \sum_{i=1}^{m} \lambda_i y_i = 0, \qquad (4a)$$

$$r \ge r_{low}. \qquad (5a)$$

Другу задачу (задача **B**) сформульовано у такий спосіб:

$$r^* = \min_{r,x,y} r \qquad (1b)$$

при обмеженнях

$$\sqrt{x_i^2 + y_i^2} \le r - r_i, \quad i = 1,\dots,m, \qquad (2b)$$

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \ge r_i + r_j, \quad 1 \le i < j \le m, \qquad (3b)$$

$$\sum_{i=1}^{m} \lambda_i x_i = 0, \quad \sum_{i=1}^{m} \lambda_i y_i = 0. \qquad (4b)$$

Цільові функції (1a), (1b) є лінійними функціями і пов'язані з мінімізацією радіуса зовнішнього круга S. Квадратичні обмеження (2a) та обмеження (2b), що містять радикали, означають, що кожний круг S_i, $i = 1,\dots,m$, належить кругу S. Квадратичні обмеження (3a) та обмеження з радикалами (3b) гарантують, що ніякі два круги із сімейства кругів S_i, $i = 1,\dots,m$ не перетинаються. Лінійні обмеження (4a), (4b) означають, що центр ваги кругів S_i, $i = 1,\dots,m$, знаходиться в початку координат. Обмеження (5a) забезпечує додатнє значення r, потрібне для коректності обмежень (2a). Задача **B** вільна від аналога для обмеження (5a), тому що обмеження (2b) автоматично враховує додатність змінної r.

За допомогою лагранжевих двоїстих оцінок для квадратичної моделі задачі **A** в статті [2] обґрунтовано оптимальну збалансовану упаковку для $m = 5$, $r_1 = 0.1$, $r_2 = 0.2$, $r_3 = 0.3$, $r_4 = 0.5$, $r_5 = 0.8$, $w_1 = 0.0785$, $w_2 = 0.314$, $w_3 = 0.7065$, $w_4 = 1.9625$, $w_5 = 5.024$. Розглянемо результати розв'язання цієї ж задачі солвером BARON (**B**ranch-**A**nd-**R**educe **O**ptimization **N**avigator), який дозволяє знайти глобальні екстремуми задач нелінійного програмування [3, 4].

На рис. 1 наведено оптимальні збалансовані упаковки п'яти кругів для задачі **A** (рис. 1*б* та 1*д*) та задачі **B** (рис.1*в* та 1*е*), отримані з двох стартових точок (рис.1*а* та 1*г* у масштабі 1.31623:10). Їм відповідає мінімальний радіус $r^* = 1.31623$.

Глобальні мінімуми задач **A** та **B** для першої стартової точки (рис. 1*а,* 1*б* та 1*в*) наведено в табл. 1. Кожен із них можна отримати з іншого за допомогою повороту останнього на деякий кут відносно центра круга S

. Цього неможливо зробити з глобальними мінімумами задач **A** та **B**, отриманими для другої стартової точки.

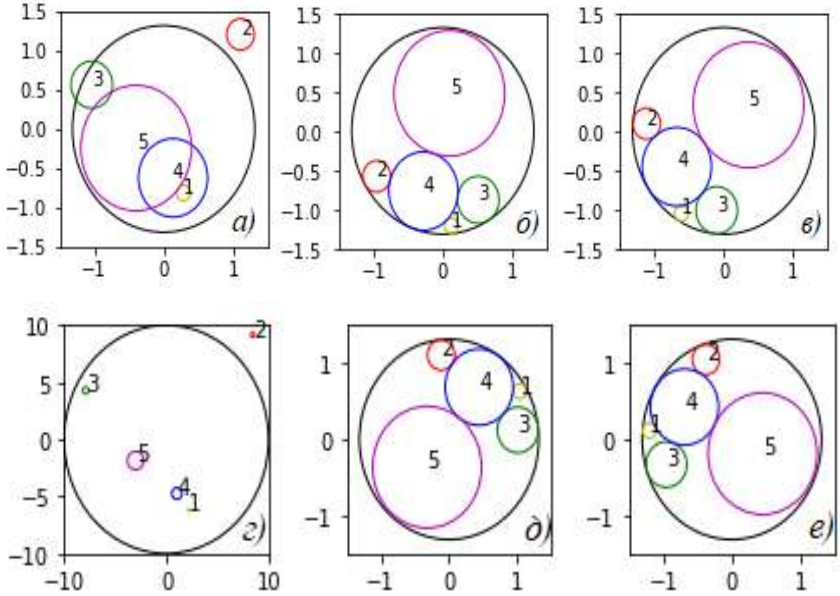


Рис. 1. BARON: глобальні мінімуми для двох стартових точок

Таблиця 1 – Розв'язки задач – стартова точка $r_0 = r^* = 1.31623$

	Стартова точка		Розв'язок задачі A		Розв'язок задачі B	
i	x_i	y_i	x_i	y_i	x_i	y_i
1	0.28749	-0.8164	0.125172	-1.20975	-0.610319	-1.05201
2	1.11061	1.20344	-0.954181	-0.579238	-1.11239	0.09249
3	-1.03791	0.56363	0.512519	-0.877528	-0.101387	-1.01116
4	0.13566	-0.62354	-0.279779	-0.766786	-0.677155	-0.455748
5	-0.39591	-0.24417	0.094896	0.478047	0.357832	0.330878

В табл.2 наведено затрати солвера BARON для розв'язання задач **A** та **B**: кількість ітерацій і час (в секундах) для нижніх границь на

мінімальний радіус r^*. Із таблиці видно, що для задачі **A** точна нижня границя для $r^* = 1.31623$ досягається вдвічі швидше для першої стартової точки, ніж для другої стартової точки.

Таблиця 2 – Результати роботи BARONa: верхня границя **1.31623**

Задача A (перша стартова точка)			Задача B (перша стартова точка)		
Iteration	Time (s)	Lower bound	Iteration	Time (s)	Lower bound
19442	105.47	0.800000	17795	124.64	1.07820
39507	225.48	0.881865	31588	224.62	1.24300
60531	315.50	1.18031	44142	314.63	1.26828
87983	435.51	1.27354	60428	434.63	1.28566
107956	**540.60**	**1.31623**	77108	554.66	1.29411
			97593	**708.28**	**1.31623**
Задача A (друга стартова точка)			Задача B (друга стартова точка)		
Iteration	Time (s)	Lower bound	Iteration	Time (s)	Lower bound
24431	215.19	0.800000	26506	211.30	1.02077
43623	432.20	0.885228	53778	421.35	1.17284
60285	616.76	1.12001	80561	631.40	1.26184
77345	831.48	1.24381	107086	841.42	1.29292
88026	985.51	1.27344	**116293**	**911.78**	**1.31623**
99580	**1148.45**	**1.31623**			

Література

1. Stetsyuk P., Romanova T., Scheithauer G. On the global minimum in a balanced circular packing problem. Optimization Letters. 2016. № 10. P. 1347–1360.

2. Стецюк П.И., Романова Т.Е., Лиховид А.П. О двойственной оценке Шора для задачи равновесной упаковки кругов // Матеріали Двадцять першого Міжнародного науково-практичного семінару "*Комбінаторні конфігурації та їх застосування*"' (Кропивницький, 17–18 травня 2019 року) / За ред. Г.П. Донця. Кропивницький, 2019. С. 147–151.

3. Tawarmalani M., Sahinidis N.V. A polyhedral branch-and-cut approach to global optimization. Mathematical Programming. 2005. 103(2). P. 225–249.

4. Sahinidis N.V., BARON 21.1.13: Global Optimization of Mixed-Integer Nonlinear Programs, User's manual, 2021.

ДИНАМІКА РОЗГОРТАННЯ ЛОГАРИФМІЧНИХ СПІРАЛЕЙ У ПРИРОДІ

Тимофієва Н. К.

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

DYNAMICS OF LOGARITHMIC SPIRAL DEVELOPMENT IN NATURE

Abstract. Representation of biological space as a sign combinatorial allows to explain various phenomena in nature, in particular the formation of logarithmic spirals during the growth of shells of some species of mollusks, galaxy sleeves, spirals of rose petals, etc. The dynamics of their formation is explained by the finite sequences that take place during the deployment of the agreed spaces and which are presented geometrically using polar coordinates.

Вступ. При розгортанні знакових комбінаторних (біологічних) просторів, утворюються комбінаторні числа, зокрема числа Фібоначчі [1], через які в живій природі можна пояснити наявність логарифмічних спіралей. Ці спіралі утворюються завдяки скінченним послідовностям, які мають місце при розгортанні оговорених просторів і які подаються геометрично з використанням полярних координат.

Постановка задачі. Утворення логарифмічної спіралі геометрично подається через “золотий прямокутник”, в якого одна сторона довша в 1.618 разів («золоте» число або золотий перетин) [2]. Присутність золотого перетину в природі проявляється через числа Фібоначчі, які утворюються з арифметичного трикутника. Останній з’являється при розгортанні комбінаторних просторів із утворених скінченних послідовностей. Завдяки цим послідовностям можна прослідкувати динаміку утворення логарифмічних спіралей.

Підхід, що пропонується. Використовуючи скінченні послідовності, які утворюються при розгортанні знакових комбінаторних

просторів та представлення їхніх елементів у полярних координатах, подано динаміку утворення логарифмічних спіралей у природі.

Аналіз останніх досліджень та публікацій за темою. Описані в літературі комбінаторні простори, як правило, зводять до метричних. Деякі автори вважають, що точками комбінаторного простору є рекурсивні функції. В літературі також описано евклідові комбінаторні простори. Але особливістю знакових комбінаторних просторів є не просто існування заданої множини точок комбінаторного характеру, між якими уведено віддаль, а утворення їх з елементів однієї або кількох базових множин з використанням певної системи правил. Для задання комбінаторного простору достатньо увести одну або кілька базових множин, із елементів яких формуються його точки, тип комбінаторної конфігурації та систему правил, за допомогою яких він розгортається.

Утворення скінченних послідовностей при розгортанні знакових комбінаторних просторів. Характерною особливістю знакових комбінаторних просторів є утворення заданої множини точок комбінаторного характеру із елементів однієї або кількох базових множин з використанням певної системи правил [1]. Для задання знакового комбінаторного простору достатньо увести одну або кілька базових множин, із елементів яких формуються його точки, тип комбінаторної конфігурації та систему правил, за допомогою яких він розгортається. Точками цих просторів є комбінаторні конфігурації різних типів.

В [1] наведено аксіоми знакових комбінаторних просторів і показано, що природні простори мають комбінаторну природу, якщо для них виконуються ці аксіоми. В біології існують явища, пов'язані з комбінаторними числами. При формуванні суцвіття деяких квітів, луски шишок, розміщенні листя дерев та інших рослин утворюються правильні спіралі, число рядів яких збігається з числами Фібоначчі, послідовність яких має такий вигляд: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55,…. Це говорить про те що біологічні простори мають комбінаторну природу.

При генеруванні ж множини розбиттів числа або множини розбиттів n-елементної множини на підмножини з використанням властивості періодичності одержані числові послідовності, які задають у них кількість комбінаторних конфігурацій, містять числа Фібоначчі. Наприклад, для розбиття натурального числа для $n=7$ утворена скінченна послідовність, яка задає кількість розбиттів у їхній множині, має вигляд 1, 3, 4, 3, 2, 1, 1, де останні чотири цифри – числа Фібоначчі.

При розгортанні знакового комбінаторного простору з використанням властивості періодичності [3], точкою якого є сполучення без повторень або розбиття n-елементної множини на підмножини,

одержані числові послідовності, які задають у них кількість комбінаторних конфігурацій, утворюють комбінаторні числа та являють собою біноміальні коефіцієнти, що утворюють арифметичний трикутник. Сформулюємо такі теореми.

Теорема 1. Значення послідовності, які задають кількість сполучень без повторень w у їхній множині W, що упорядкована з використанням рекурентно-періодичного методу генерування комбінаторних конфігурацій, утворюють арифметичний трикутник та є фігурними числами.

Підмножину $W_\eta \subset W$ назвемо підмножиною ізоморфних комбінаторних конфігурацій $w = (w_1, ..., w_\eta)$, якщо її елементи – ізоморфні комбінаторні конфігурації; $\eta \in \{1, ..., n\}$ – кількість елементів у w або підмножин $w_j \subset w$.

Теорема 2. Значення послідовностей, які задають кількість розбиттів n-елементної множини на підмножини у їхній підмножині W_η для $\eta = 2$, що упорядкована з використанням рекурентно-періодичного методу генерування комбінаторних конфігурацій, утворюють арифметичний трикутник та є фігурними числами.

Доведення теорем 1, 2 наведено в [1].

В результаті доведення теорем отримуємо такі послідовності: 1, 2, 3, 4, 5, 6, ...; 1, 3, 6, 10, 15, 21, ...; 1, 4, 10, 20, 35, 56,….

Існують роботи, в яких елементи скінченних послідовностей чисел подано через полярні координати [4]. Їхнє представлення на поверхні утворює логарифмічні спіралі. Із наведених теорем видно, що при розгортанні знакових комбінаторних просторів утворюються скінченні послідовності. Представлення елементів цих послідовностей в полярних координатах пояснює, як утворюються логарифмічні спіралі в природі.

Для елементів послідовності 1, 2, 3, 4, 5, 6,... визначимо полярні координати: для 1: (1,1), де перше число – відстань від початку координат до заданої точки, друге – кутова координата в радіанах; тобто дорівнює 1 радіан, для 2, 3, 4, 5, 6 відповідно координати: (2,2), (3,3), (4,4), (5,5), (5,5). Якщо нанести точки з цими координатами на площину проти годинникової стрілки та з'єднати їх лінією, то отримаємо спіраль Архімеда.

Послідовності 1, 3, 6, 10, 15, 21, ...; та 1, 4, 10, 20, 35, 56,..., задамо у полярних координатах: (1, 1), (3,2), (6,3), (10, 4), (15,5), (21, 6), та (1, 1), (4,2), (10, 3), (20, 4), (35,5), (56, 6).Якщо побудувати криві на площині по

годинниковій стрілці, то вони наближаються до логарифмічної спіралі.

Висновок. Отже, при розгортанні знакових комбінаторних просторів, зокрема біологічних, із згорнутого утворюються скінченні послідовності, елементи яких задають кількість комбінаторних конфігурацій у їхніх множинах (або підмножинах ізоморфних комбінаторних конфігурацій). Динаміка утворення логарифмічних спіралей прослідковується завдяки цим послідовностям, елементи яких подано в полярних координатах.

Література

1. Тимофієва Н.К. Знакові комбінаторні простори та штучний інтелект. *Штучний інтелект*. 2015. 1–2(67–68). С.180 –189.

2. Фернандо Корбалан. Золотое сечение. Математический язык красоты. Мир математики: в 40 т. Пер. с англ. М.: Де Агостини, 2014. Т 1. 160 с.

3. Тимофієва Н.К. Теоретико-числові методи розв'язання задач комбінаторної оптимізації. Автореф. дис... докт. техн. наук. – Ін-т кібернетики ім. В.М. Глушкова НАН України, Київ. – 2007. – 32 с.

4. Почему простые числа образуют спирали? https://www.youtube.com/watch?v=DxntHp7-wbg (дата звернення: 5.05.2021).

ПАРАЛЕЛЬНІ УПОРЯДКУВАННЯ ДЛЯ ПОВНИХ ДВОДОЛЬНИХ ГРАФІВ

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

PARALLEL SEQUENCINGS FOR COMPLETE BIPARTITE GRAPHS

Abstract. This research considers the problems of parallel sequencing for graphs of special structure, known as complete bipartite graphs. The cases where interruptions can significantly affect the solutions' optimality were investigated. The dependence between the given parameters and graphs' structure for cases when such influence is insignificant was received. Issues that need to be investigated further are presented.

Задачі паралельного упорядкування вершин орграфів в класичній постановці мають два формулювання: або по заданому графу $G=(V,U)$ та

заданій ширині упорядкування h побудувати паралельне упорядкування S мінімальної довжини l, або по заданій довжині упорядкування l побудувати упорядкування мінімальної ширини h [1]. Розглянемо першу задачу для випадку повних дводольних графів, тобто графів $K_{m,r}$, $m,r \in N$, де m та r — кількість вершин у першій та другій долях відповідно. Вважатимемо, що ваги вершин графа рівні та приймемо їх за 1. Розглянемо залежність довжини оптимального упорядкування від кількості вершин в долях графа та ширини. Якщо $m = r$ (тобто граф $K_{m,m}$), а $h = m$-1, то $l(S^*) = 4$.

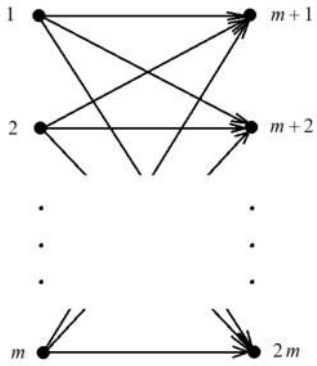


Рис.1. *Граф* $K_{m,m}$

Дійсно, в цьому випадку схематично оптимальне упорядкування можна представити у наступному вигляді:

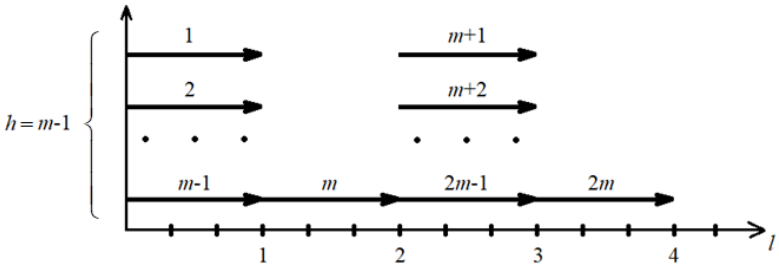


Рис.2. *Схема розподілу вершин графа* $K_{m,m}$ *без переривань*

Відомо, що в задачах упорядкування дослідження впливу переривань на оптимальність заслуговують особливої уваги [2,3].

Виникає питання, а чи не можна зменшити довжину, дозволивши переривання при розподілі вершин, тобто виходити з припущення, що виконання тих робіт, яким у відповідність поставлені вершини, можуть перериватися із подальшим поновленням їх виконання. Цілком зрозуміло, що перш за все такі переривання повинні стосуватися вершин першої долі дводольного графа, а саме чи можна зменшити за рахунок переривань кількість місць, які вони займають. Розглянемо частковий випадок дводольного графа, а саме граф $K_{4,4}$ та $h = 3$.

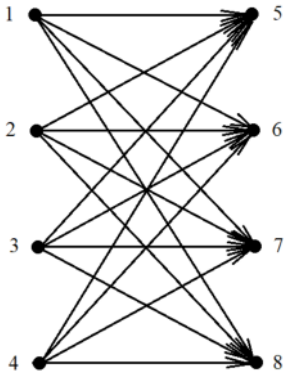


Рис.3. *Граф* $K_{4,4}$

Для цього графа, як видно із рис.2, довжина оптимального упорядкування $l = 4$.

Дозволивши переривання при виконанні 2-ої та 4-ої роботи, яким відповідають вершини 2 і 4 першої долі, довжина тієї частини упорядкування, на якій розміщуються вершини першої долі, може бути скорочена на 2/3. Таке ж скорочення ми отримаємо і при розташуванні вершин другої долі.

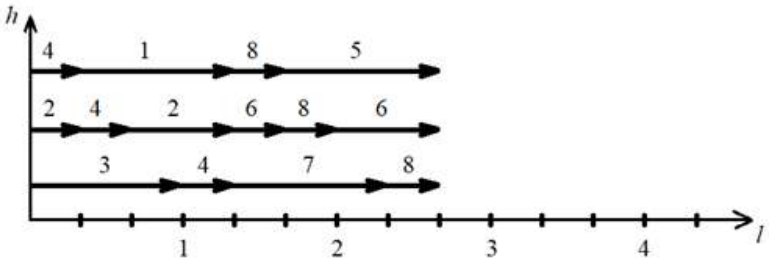


Рис.4. *Схема розподілу вершин графа $K_{4,4}$ з перериваннями*

Позначимо l^* довжину оптимального упорядкування без переривань а $l_{П}^*$ довжину оптимального упорядкування з перериваннями. Зрозуміло, що $l_{П}^* \le l^*$. Тоді виграш від переривань можна оцінити наступним чином:

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

Для графа $K_{4,4}$ отримуємо виграш $W \approx 33\%$, для $K_{5,5}$ та $h = 4$ виграш 38%. У загальному випадку для графів $K_{m,m}$ ($m > 2$) отримуємо:

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

Аналізуючи залежність виграшів від значень m, можна відмітити, що з певного кроку відбувається стабілізація цих змін.

Графік, наведений на рис. 5, ілюструє як проходить ця стабілізація.

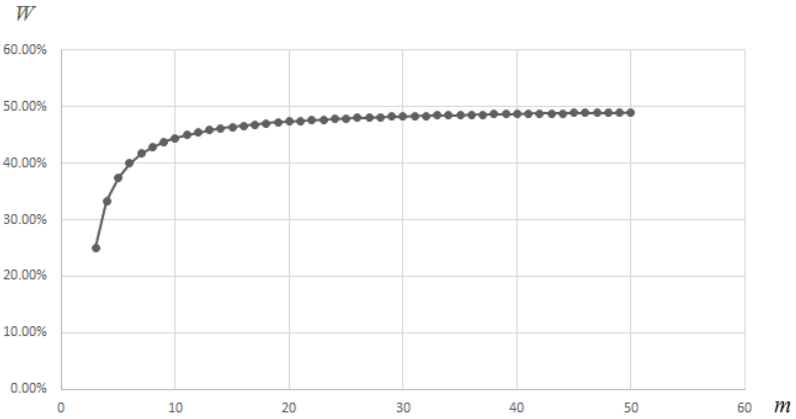


Рис.5. *Залежність виграшу W від кількості вершин при h=m-1*

Аналізуючи виграші, які можна досягти дозволивши переривання, виникають наступні питання:

1. Які виграші будуть давати переривання при $m > r$ або $m < r$?
2. Як залежить виграш від ширини упорядкування h при довільних m та r?
3. Як можна мінімізувати кількість переривань не погіршуючи значення цільової функції?

Література.

1. Коваленко Є. О., Турчина В. А. Аналіз впливу структури графів на оптимальність розв'язку задач паралельного упорядкування з перериваннями. Питання прикладної математики і математичного моделювання: зб. наук. пр. – Дніпро, 2021. – Вип. 21. – С. 130-137.
2. P. Brucker, S. Heitmann, and J. Hurink. How useful are preemptive schedules? Operations Research Letters 31, 31:129-136, 2003.
3. P. Baptiste On preemption redundancy. Technical Report, IBM T.J. Watson ResearchCenter, Yorktown Heights, NY, 2002.

СТАБІЛІЗАЦІЯ ДОВЖИНИ УПОРЯДКУВАННЯ ТА АНАЛІЗ ВПЛИВУ ЗМІН ПОЧАТКОВИХ ДАНИХ НА ОПТИМАЛЬНІСТЬ

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

STABILIZATION OF ORDER LENGTH AND ANALYSIS OF THE INFLUENCE OF CHANGES IN INITIAL DATA ON THE OPTIMALITY

The paper considers the known problems of parallel ordering of vertices of oriented graphs for the case of certain restrictions. The concept of stabilization of length and conditions of its occurrence is introduced. For stabilized optimal ordering, the issues of influencing the optimality of changes in the initial data related to the simplification of technological constraints and the reduction of time constraints are investigated. It is shown that in these cases the so-called anomalies also occur.

Ряд практичних задачах, що пов'язані з визначенням оптимального розподілу частково впорядкованих робіт між фіксованою кількістю ресурсів, зводиться до відомих оптимізаційних задач на графах. Одна із таких задач [1] розглядається в даній роботі.

Задано орграф $G=(V,U)$, ширина упорядкування h, список пріоритетів $L=(i_1, \ldots, i_n)$, ваги вершин $T=(\tau_1, \ldots, \tau_n)$ [2]. Необхідно побудувати паралельне упорядкування ширини h, яке задовольняє список пріоритетів та має мінімальну довжину [3].

Зрозуміло, що мінімальне значення довжини упорядкування не може бути менше довжини критичного шляху у графі.

Означення. Будемо казати, що довжина оптимального упорядкування стабілізована, якщо вона дорівнює довжині критичного шляху.

Розглянемо як впливає значення ширини на стабілізацію довжини.

Приклад 1. Нехай задано граф G (рис.1), $h=3$ $L=(1,2,3,4,5,6,7,8,9,10)$, $T=(5,3,3,2,8,8,7,4,2,9)$.

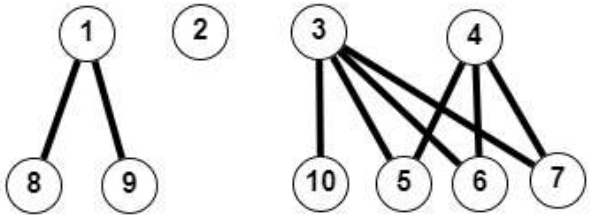


Рис.1 *Граф G*

Знайдемо оптимальне упорядкування та його довжину.

Таблиця 1.
Оптимальне упорядкування вершин графа G при $h=3$

1	1	1	1	1	5	5	5	5	5	5	5	5	8	8	8	8		
2	2	2	4	4	6	6	6	6	6	6	6	6	9	9				
3	3	3	10	10	10	10	10	10	10	10	10	7	7	7	7	7	7	7

Довжина $l=19$. Збільшимо кількість виконавців. Нехай $h=5$. Отримаємо упорядкування (табл. 2) з довжиною $l=12$.

Таблиця 2.
Оптимальне упорядкування вершин графа G при $h=5$

1	1	1	1	1	8	8	8	8	9	9	
2	2	2	5	5	5	5	5	5	5	5	

3	3	3	6	6	6	6	6	6	6	6	
4	4		7	7	7	7	7	7	7		
			10	10	10	10	10	10	10	10	10

Аналізуючи результати, отримані при збільшенні ширини упорядкування, можна зробити висновок, що на деякому кроці відбувається стабілізація, тобто, отримуємо оптимальне упорядкування, довжина якого дорівнює довжині критичного шляху.

Постає питання чи будуть мати місце аномалії після стабілізації довжини упорядкування. Розглянемо на прикладі 1 покращення інших умов. Послабимо обмеження, що стосується технологічного порядку. Отримаємо граф *G'* (рис. 2).

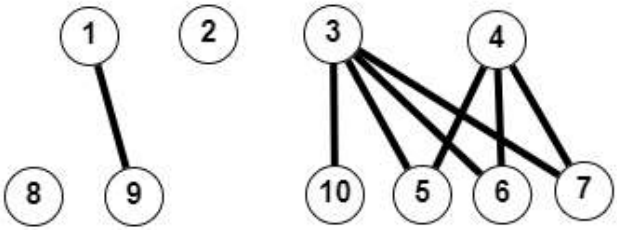


Рис.2 *Граф G'*

Замість очікуваної довжини l=12, як видно із табл. 3, довжина l=13. Тобто, послаблення технологічних обмежень погіршило значення цільової функції.

Таблиця 3.
Оптимальне упорядкування вершин графа *G'* при h=5

1	1	1	1	1	9	9						
2	2	2	5	5	5	5	5	5	5	5		
3	3	3	6	6	6	6	6	6	6	6		
4	4		7	7	7	7	7	7	7			
8	8	8	8	10	10	10	10	10	10	10	10	10

Погіршення можемо отримати і якщо зменшимо час виконання робіт. При τ_1'=3 отримаємо оптимальне упорядкування (табл. 4), довжина якого l=14.

Таблиця 4.

Оптимальне упорядкування вершин графа G при h=5, τ_1'=3

1	1	1	5	5	5	5	5	5	5	5			
2	2	2	6	6	6	6	6	6	6	6			
3	3	3	7	7	7	7	7	7	7				
4	4		8	8	8	8							
			9	9	10	10	10	10	10	10	10	10	10

Інтуїтивне покращення початкових умов знову призвело до погіршення результату на 16,7%.

Таким чином, приходимо до висновку, що і у випадках стабілізації довжини оптимального упорядкування, аномалії мають місце.

Подальшого дослідження потребують питання, пов'язані з отриманням умов, при яких відбувається стабілізація, та встановленням аналітичних залежностей, які будуть гарантувати відсутність аномалій для упорядкувань стабілізованої довжини.

Література

1. Бурдюк В.Я. Алгоритмы параллельного упорядочения: Учебное пособие / В.Я. Бурдюк, В.А. Турчина. – Д.: ДГУ, 1985. – 84с.

2. Челпанова О.О. Узагальнення аномальних випадків у задачах упорядкування. / О.О. Челпанова, В.А. Турчина. // Питання прикладної математики та математичного моделювання: зб. наук. пр. Дніпро, 2021. – Вип. 21. – С. 220-226.

3. Graham R. L. Bounds on multiprocessing timing anomalies // SIAM J. Appl. Math., 1969. V. 17. P. 416–429.

ЧАСТИНА ІІ

OPEN SCIENCE AND REPRODUCIBLE RESEARCH

Olexandr Konovalov
University of St Andrews

OPEN SCIENCE AND REPRODUCIBLE RESEARCH

Abstract. I will talk about of open science practices in computational research and present some novel ways of sharing reproducible computational experiments using Jupyter notebooks. Furthermore, there will be a discussion of associated technical skills, needed for modern collaborative research, and of training opportunities offered by the Carpentries (https://carpentries.org/).

Відкрита наука та відтворювані дослідження

(Ця доповідь розповість про деякі аспекти відкритої науки, які мають відношення до досліджень за допомогою обчислювальних методів. Мова піде про сучасні практики розробки та розповсюдження наукового програмного забезпечення з відкритим кодом, які сприяють науковому співробітництву та відтворюваності досліджень, а також про те, як їми оволодіти. Приклади будуть використовувати ноутбуки Jupyter з використанням Python та R, але попереднє знайомство з ними мати не обов'язково.

Literature

1. "Україна приєдналась до країн ЄС, що мають затверджений план реалізації принципів відкритої науки". Міністерство освіти і науки України, 08.10.2022. https://www.kmu.gov.ua/news/ukraina-pryiednalas-do-krain-ies-shcho-maiut-zatverdzhenyi-plan-realizatsii-pryntsypiv-vidkrytoi-nauky
2. M. Beg *et al*., "Using Jupyter for Reproducible Scientific Workflows," in *Computing in Science & Engineering*, vol. 23, no. 2, pp. 36-46, 1 March-April 2021. https://doi.org/10.1109/MCSE.2021.3052101
3. O. Berezko, L.M.P. Medina, G. Malaguarnera et al. Perspectives on Open Science and Scholarly Publishing: a Survey Study Focusing on Early Career Researchers in Europe. F1000Research 2021, 10:1306. https://doi.org/10.12688/f1000research.74831.1

4. O. Konovalov. Template for publishing reproducible experiments in Python Jupyter notebooks runnable on Binder, v0.2.0, 2023. Zenodo. https://doi.org/10.5281/zenodo.7703173
5. O. Konovalov. Towards Open Science training for Ukraine: translating Software Carpentry lessons, 2023. https://olexandr-konovalov.github.io/posts/2023/03/28/carpentries-translation/
6. National Open Science Plan. Approved by the Decree of the Cabinet of Ministers of Ukraine on October 8, 2022, No 892-p. https://mon.gov.ua/storage/app/media/nauka/2023/01/26/National-Open-Science-Plan-Ukraine.pdf
7. Understanding open science. UNESCO, SC-PBS-STIP/2022/OST/1, 10.54677/UTCD9302, 2022. https://doi.org/10.54677/UTCD9302
8. Y. Yehudi, S. Aragon S and O. Konovalov. Code is science manifesto. *F1000Research* 2018, 7 (ISCB Comm J):1582 (poster). https://doi.org/10.7490/f1000research.1116166.1

GAMMA-TETA ALGORITHM FOR GRAPHS PLANARIZATION

Sergey Sgadov
National University "Zaporizhzhia Polytechnic"

Abstract . In this paper, one considers a way for modifying the thread method algorithm for checking the graph for planarity. The system of graph isometric cycles obtained as a result of the algorithm induces the rotation of vertices to describe the topological pattern of a planar graph for the purposes of subsequent visualization. The presented algorithm is based on the rearrangement of the reference cycle and the construction of a list of reverse routes. The basis of the calculation is the construction of a matrix of fundamental cycles based on the construction of the spanning tree of the graph by the depth-first search method. Based on the matrix of fundamental cycles, the chains that form the routes are selected. To obtain a system of isometric cycles, routes are embedded using a modified gamma algorithm. An adapted version of the gamma algorithm in the application of the thread method makes it possible to embed path-forming chains into graph cycles and simplifies the task of finding suitable cycles for this. Adaptation of the gamma algorithm includes an additional predictive index to determine the possibility of embedding subsequent paths in a given cycle. Graph planarization is the most important subtask in solving many relevant applied problems, such as

designing complex products and systems, flat constructs, analyzing social networks, etc.

Introduction. Placing planar graphs on the plane and checking graphs for planarity - an important subtask that arises when solving applied problems in many areas. In the work [1] presented a method of controlling the planarity of graphs by simultaneously constructing mathematical structures describing topological patterns of graphs for visualization purposes. The method is based on the construction of isometric cycles [2] by embedding chains in the DFS support cycle -of the graph tree [1], rotating the vertices of the graph [3]. In [5], a modified string method for determining planarity and building a vertex rotation list with a computational complexity comparable to the Hopcroft-Tarian algorithm [4] was investigated. It is based on the method of reshaping DFS -trees and backpaths, as well as on the recursive method of constructing hoops. This paper proposes a modification of the thread method [1] using the principles underlying the gamma algorithm [6] for embedding chains into cycles.

The original version of the gamma algorithm [6] is that a set of segments is built first;

for each segment, a value $\Gamma(S)$, that equals to the number of cycles in which all vertices of the segment S placed, is calculated;

then the segment with the minimum number $\Gamma(S) > 0$ must be selected;

in selected segment, the chain between two contact vertices is selected – so,this chain fits into any face containing this segment.

The disadvantage of the gamma -algorithm in this form is, apparently, the fact that, unlike the thread method, the segment can be not only a chain, but also a much larger component of connectivity, which also separately arises to planarize.

Problem formulation. Let G be a given arbitrary graph. Sequentially reviewing all the vertices of the graph, we will remove the "hanging" loops. Let it be inseparable - i.e. such that it is a connected undirected graph without loops and multiple edges, without bridges and points of articulation, without vertices with a local degree less than or equal to two. In the future, we will consider non-separable undirected graphs. A topological pattern of a planar graph is the equally directed rotation of all its vertices, which induces a set of isometric cycles of the graph that satisfies the McLain planarity criterion [5].

Thus, the task is to obtain an algorithm that will check a given inseparable graph G and construct its topological drawing, first in the form of a system of isometric cycles, and then it will be possible to obtain a list of rotations of vertices from them.

Description of the algorithm. As a basis, one will take the thread method algorithm given in [5] and modify it as shown below.

1. Construction of the DFS -tree of the graph and formation of the matrix of fundamental cycles (MFC).

2. We select the cycle of the maximum length of the reference cycle from the MFC -and extend it.

3. Construction of reverse routes. The edges of the cycles that are not included in the reference cycle and the corresponding chords are writen out from the MFC. Then one leave the longest ones unchanged and process the list so that there will be anyone common edges.

4. Embedding reverse routes. Using the gamma -algorithm approach, we consistently search for suitable routes, simultaneously determine the cycles on which the ends of the routes lie and embed them in these cycles.

5. If the embedding of the reverse routes was successful, it is concluded that the graph is planar, and then a list of rotations is built according to the obtained system of isometric cycles.

In [5] the operation of including an edge (route) in a loop was introduced when the content of the loop is divided into a part where the end vertices of the edge participate in the formation of routes. Then, following the sequence, arcs are joined to the selected parts.

Using the concept of the operation of including a route in a cycle, we will build an algorithm for embedding reverse routes from the list **W** into graph cycles, modifying a similar one [1, 5]:

1. Let the list **C** be a list of isometric cycles.

Because the reference cycle divides the surface R^2 into inner and outer regions, for initialization we will add the reference cycle in the forward and reverse directions to the list **C.**

2. Let's review the list **W** and for each i calculate $\Gamma(W_i)$on the set **C** . Where value $\Gamma(S)$equal to the number of faces (cycles) in which all vertices of the segment lie S. Let's find i_{min}something that corresponds to $\min \Gamma(W_i)$.

3. If $\min \Gamma(W_i) = 1$, then the path is found $W_{i_{min}}$and there is an element of the cycle list C_j, such that it includes the initial and final vertices of the route $W_{i_{min}}$. Then we include the route $W_{i_{min}}$in the loop C_j, which generates the loops $C_j'$and C_j''. We replace the cycle $C_j$with these two new cycles $C_j'$and C_j''.

4. If $\min \Gamma(W_i) \geq 2$, then this means that the route claims more than one face and there exists a set of cycles $\boldsymbol{\Theta} \in C$such that $W_{i_{min}}$can be embedded in any -cycle from this set. Then we introduce an additional criterion

$$\theta_{ij} = \theta$$

for the route W_i– the number of elements of the list **W** , the ends of which lie both on $W_i$and on the given cycle $\Theta_j \in \boldsymbol{\Theta}$**.**

If $\theta_{ij} = 0$, then it is clear that -it is impossible to insert more routes into j and the cycle. Therefore, let's calculate $\theta_{ij}$for all elements$\Theta_j \in \mathbf{\Theta}$ and find j such that θ_{ij}it is maximal. Then, using this criterion for, $W_{i_{min}}$we will find the cycle into which we can mount the route, without fearing that in subsequent iterations of the algorithm, a situation will arise when it will not be possible to mount routes that lie on the given $W_{i_{min}}$.

5. We successively embed the routes from the **W** list until they are completely exhausted. If it turns out that it is impossible to achieve the exhaustion of the list **W** , that is, if at some step a segment S was encountered for which there is no containing face, then the graph is nonplanar

If the graph G is planar, then the algorithm constructs its planar arrangement in the form of a system of isometric cycles **C** .

6. Based on the system of isometric cycles **C,** we will construct lists of rotations for this graph using data algorithms, for example, in work [8].

Conclusions. The paper presents an algorithm obtained by a modification of the thread method [5], which uses an improved way of embedding routes with the use of additional criteria, which allows avoiding incorrect selection of cycles at the final stage of the algorithm. An approach to modifying the gamma algorithm for chain embedding is also obtained. Thus, it is proposed to evaluate using not only the gamma -criterion for paths, but the theta criterion to allow situations where a route claims multiple cycles. The obtained system of isometric cycles of the graph induces the rotation of the vertices to describe the topological pattern of the planar graph. Obtaining the rotation of the vertices of the graph immediately solves the two most important tasks of graph theory: the task of checking the graph for planarity and the task of constructing a topological drawing of a planar graph.

Literature

1. S. V. Kurapov, M. V. Davydovsky. Checking planarity and constructing a topological drawing of a planar graph (by depth-first search) // PDM. 2016. No. 2(32). P.100–115.

2. S. V. Kurapov, M. V. Davydovsky., Sgadov Extraction of the flat part of the SEP graph. Matroids and algebra of structural numbers *Components and technologies.* 2018. No. 9. P. 56–59

3. G. Ringel. Map coloring theorem. / H. Ringel - M.: Myr. - 1977. - 126 p.;

4. J.E. Hopcroft, R.E. Taryan. Isomorphism of planar graphs / J.E. Hopcroft, R.E. Taryan // In the book: Cybernetic collection. New series. issue 12. – 1975. – P.39–61;

5. S.V. Kurapov, M.V. Davydovsky, A.V. Tolok Modified algorithm for checking the planarity of a graph and constructing a topological pattern.

Thread method. // Scientific visualization, 2018, volume 10, number 4, pages 53 – 74

6. Hrebeniuk B. Modification of the analytical gamma-algorithm for the flat layout of the graph. *Computer Science & Software Engineering* , 2018. Vol. 2292. P. 46 -54

7. Mc-Lane S. Combinatory condition for flat graphs / S. Mc-Lane // In the book: Cybernetic collection. New series. – 1970. Issue 7. – P.133–144.

8. S. V. Kurapov, M. V. Davydovsky. Algorithms for constructing topological and geometric drawings of the SEP graph. *Components and technologies.* 2017. No. 6. P. 128–132

GENERAL TECHNIQUE OF DECISION-MAKING BY THE OPERATORS IN FLIGHT EMERGENCY

Shmelova T., Sikirda Yu., Yatsko M.
National Aviation University
Flight Academy of the National Aviation University

Abstract. The general technique of decision-making by the operators in flight emergencies is developed. It includes: analysis of flight emergency as a complex situation, construction of the algorithm of the pilot actions in flight emergency, modeling of decision-making by the pilot in flight emergency, modeling and synchronization of decision-making for all collaborative decision-making participants in flight emergency, and evaluation of the effectiveness of the decisions. The designed deterministic, stochastic, and non-stochastic models can be used both for the informational support and professional training of the air navigation system operators.

Introduction. Aviation is the safest mode of transport. This is a generally accepted fact, which is confirmed by statistics. In 2014-2019, there were 107 accidents in the world, during which 3245 people died. Whereas in 2018 alone, airlines around the world carried nearly 4.5 billion passengers on about 45 million flights [1]. 2017 was the safest year in the history of commercial airlines: a total of 10 crashes were registered, of which only half were passenger aircraft (ACFT). In 2018, according to the Aviation Safety Network [2], the number of accidents rose sharply to 18, killing 561 people.

ACFT crashes are very rare, about 200 times less common than car accidents. Civil aviation statistics over the past six decades show a downward trend in tragic events and increased security. But taking into account the

registered accidents in 2019, these indicators are above the average for the last five years [1; 2].

The reasons for aviation accidents are human factors (68%), technical factors (18%), and environmental factors (14%) [3].

According to a Boeing study [4], 11% of aviation accidents with human casualties occur during a flight at cruising altitude, 2% – during the descent phase, 2% – during the initial approach to landing, 29% – at the stage of the final approach to landing, 24% – during landing. At the beginning of the flight, according to statistics, there are fewer problems: 12% of ACFT crashes occur during take-off and initial climbing (before removing the flaps), 13% – during climbing and another 7% – on the ground during towing, taxiing, loading / unloading, etc.

The most frequent incident is the announcement of an emergency (39%), in the second place – the risk of collision / violation of the intervals between ACFT (18%). A significant share is occupied by engine failure (13%), the smallest share – in collisions between aircraft (1%) [5].

The distribution of aviation accidents and incidents that occurred on the territory of Ukraine in the period from 2013 to 2017 with civilian Ukrainian and foreign aircraft by category [6] indicates that incidents most often occur due to technical, bird collisions, and engine failures, and these trends do not change significantly over the years.

In the work [7] is provided a fragment of the network graph describing the collaborative work of the ACFT crew (pilot-in-command – co-pilot) from the moment of engine failure during take-off to the issuance by the captain to continue or reject take-off. The critical time of actions of the ACFT crew and performance of works by the air traffic controller (ATCO) in case of engine failure during take-off in deterministic and stochastic conditions is obtained. With the help of network planning the analysis of joint actions of the ACFT crew (Pilot Flying and Pilot Monitoring) in the case of flight emergency (FE) "Power supply problems" is conducted, the time for operational procedures with using the method of expert assessments is determined, structurally-time table and network graph are built, a critical time of work by two pilots (Pilot Flying and Pilot Monitoring) is obtained [7]. Deterministic, stochastic, non-stochastic, and neural network models of the collaborative decision-making (CDM) by ACFT pilot/unmanned aerial vehicle's remote pilot and ATCO in FE for maximum synchronization of operators' technological procedures are developed [8; 9].

The purposes of this work is developing the general technique of decision-making (DM) by the operators in FE.

Main part. The general technique of DM by the operators in FE is presented in Fig. 1.

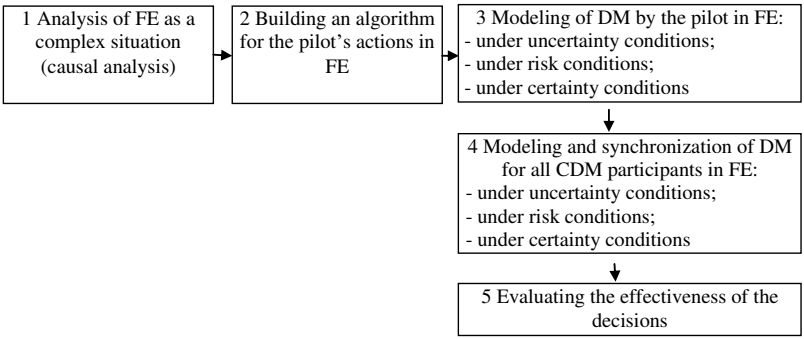


Fig. 1 – The general technique of DM by the operators in FE

The general technique of DM by the operators in FE includes next steps.

1. Analysis of situation as a complex situation: identification of causal relationships.

2. Building an algorithm for the pilot's actions in FE.

3. Modeling of DM by the pilot in the case of rejected take-off as an emergency.

3.1. Models of DM under uncertainty conditions: determination of the alternatives $\{A\}$ and factors $\{F\}$ that influence the choice of the optimal solution (tool – decision matrix) (Table 1).

Table 1

Decision-making matrix in uncertainty

	$\{A\}$	Factors influencing decision-making in emergency					
		f_1	f_2	…	f_j	…	f_n
Alternative solutions	A_1	U_{11}	U_{12}	…	U_{1j}	…	U_{1n}
	A_2	U_{21}	U_{22}	…	U_{2j}	…	U_{2n}
	…	…	…	…	…	…	….
	A_i	U_{i1}	U_{i2}	…	U_{ij}	…	U_{in}
	…	…	…	…	…	…	…
	A_m	U_{m1}	U_{m2}	…	U_{mj}	…	U_{mn}

3.2. Models of DM under risk conditions: evaluation of risk R for different solutions (tool – decision tree). Each stage of DM is characterized by solutions ($A = \{A_1; A_2; \ldots, A_n\}$), a time t of situation development on some stage, and additional value β, that depends on the stage of the situation development and DM in time for parry a situation (Figure 6). When solving the problem of

minimizing risks at each stage, additional risks arise ($+\beta_k$), the threats are increasing with time t (1):

$$R_k = t_k \sum_{i=1}^{n} p_i u_i \pm \beta_k, \quad (1)$$

where t_i – is a time of stage k;

β_k – is an additional risk on stage k;

p_i – are the probabilities of situation development, $\sum_{i=1}^{n} p_i = 1$;

u_i – are the expected outcomes (losses/profit).

The model of DM under risk is shown in Fig. 2. Step-by-step correction of the decision matrix is carried out in risk assessment.

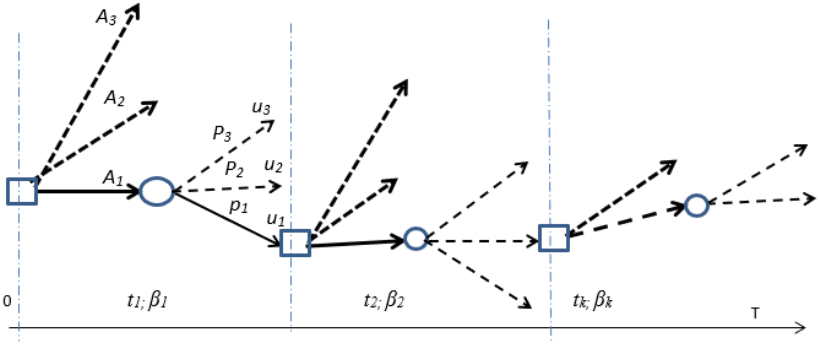


Fig. 2 – *The stages of situation development and DM in the decision tree*

3.3. Models of DM under certainty conditions: determination of the optimal solution by the criterion of minimizing the critical time of pilot actions in FE *T*, development of instructions for the pilot actions in the FE (tool – network planning).

4. Modeling and synchronization of DM for all CDM participants in FE (ACFT crew, ATCO, ground handling agents, rescue service, aerodrome service, production and dispatch service, etc.).

4.1. Under uncertainty conditions: determination of the alternatives $\{A\}$ and factors $\{F\}$ that influence the choice, determination of the optimal solution by the criterion of minimizing potential loss U (tool – decision matrix).

4.2. Under risk conditions: determination of alternatives *A* and probabilities of influence the factors *P(F)*, determination of the optimal solution by the criterion of minimizing potential risk *R* (tool – decision tree);

4.3. Under certainty conditions: determination of the optimal solution by the criterion of minimizing the critical time of collaborative actions in the FE *T*, development of instructions for joint actions of the operators in the FE (tool – network planning).

So, for example, stochastic and non-stochastic uncertainty, neural, and dynamic models can be integrated into deterministic models. When analyzing a critical situation in a team decision (A_1, A_2, A_3), each operator determines his actions to solve the problem (S_1, S_2, and S_3). In a deterministic model, some actions are ambiguous, multi-alternative (S_1, S_2, and S_3). For ambiguous actions, optimal solutions are found using stochastic DM models under risk or uncertainty conditions (Fig. 3).

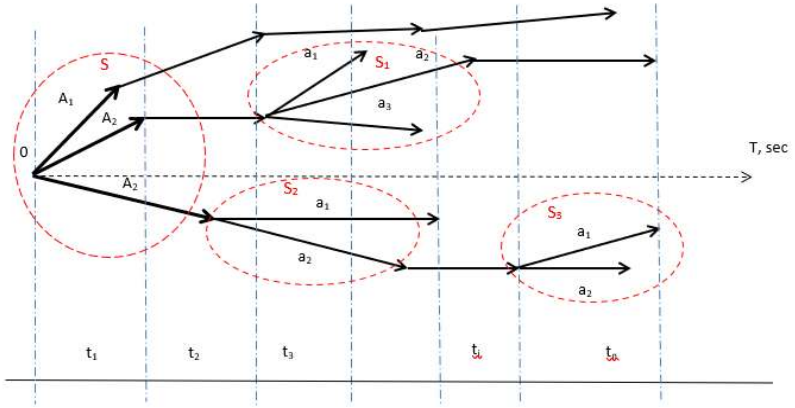


Fig. 3 – *The deterministic models with ambiguous actions (S_1 and S_2) of operators (A_1, A_2, A_3)*

After determining the minimum risks and maximum safety an integrated simplified model (S_1, S_2, S_3) is an aggregated deterministic model with included stochastic models (Fig. 4).

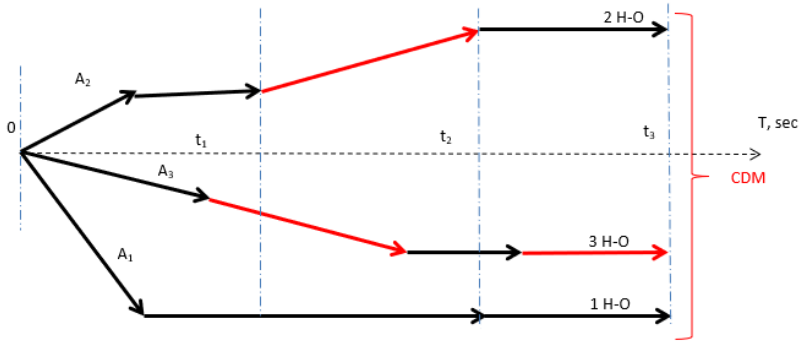


Fig. 4 – *The deterministic models with decisions A_1, A_2, A_3*

When analyzing and synthesis of situations emergency by several operators each operator determines his actions to solve problems of ensuring the safety of flights. For example, when need to build the CDM models for the pilot, air traffic controller, flight dispatcher, and technical personal, for choosing optimal actions and synchronization actions of operators in the case of rejected take-off.

5. Evaluating the effectiveness of the decisions.

Currently, the concept of Airport CDM (A-CDM) implements specific solutions that can unite the interests of partners (airport operators, aircraft operators, ground handling agents, and air traffic services) in joint work, to create the basis for effective DM through more accurate and timely information that provides all partners at the airport a single operational picture of air traffic [10]. The A-CDM system is expected to increase situational awareness and reduce the risks of unauthorized ground maneuvering, and economically improve punctuality and reduce operating costs by reducing land delays and thus saving fuel by reducing taxiing time.

Conclusion. The general technique of DM by operators in FE is developed. It includes: analysis of FE as a complex situation, construction of the algorithm of the pilot actions in FE, modeling of DM by the pilot in FE, modeling and synchronization of DM for all CDM participants in FE, and evaluation of the effectiveness of the decisions.

Step-by-step correction of the decision matrix with the help of computational systems / information technologies is carried out in risk assessment. After determining the minimum risks and maximum safety, an integrated simplified model is an aggregated deterministic model with included stochastic models. The integration of stochastic and non-stochastic models of DM to deterministic models based on a posteriori analysis of FE development will serve to correct existing and develop new instructions for pilot actions. The designed deterministic, stochastic, and non-stochastic models can be used both for the informational support and professional training of the air navigation system operators.

The direction of further research is working out models of DM for all CDM participants within the Airport CDM (A-CDM) concept that can unite the interests of partners (airport operators, aircraft operators, ground handling agents, and air traffic services) in joint work, to create the basis for effective DM through more accurate and timely information that provides all partners at the airport a single operational picture of air traffic.

Literature

1. Mori E. Fly without fear. We answer the most common questions about airplane crashes, 2021. URL: https://suspilne.media/7904-letiti-bez-strahu-

vidpovidaemo-na-najposirenisi-zapitanna-pro-aviakatastrofi/. (date of request: 17.05.2023).

2. ASN Wikibase. *Aviation Safety Network*. URL: https://aviation-safety.net/wikibase/ (date of request: 17.05.2023).

3. J. A. Wise, V. D. Hopkin, D. J. Garland (Eds.), Handbook of aviation human factors, 2nd ed., CRC Press, Florida, USA, 2016. doi:10.1201/b10401.

4. Statistical Summary of Commercial Jet Airplane Accidents Worldwide Operations 1959-2018. 50th ed. USA, Boeing, 2019. 15 p.

5. Statistical summary: Air transportation occurrences in 2017, Transportation Safety Board of Canada, 2021. URL: https://www.bst-tsb.gc.ca/eng/stats/aviation/2017/ssea-ssao-2017.html. (date of request: 28.05.2023).

6. Analysis of the state of safety of flights with civil aircraft of Ukraine based on the results of the investigation of aviation accidents and incidents in 2013-2017. Kyiv: National Bureau for the Investigation of Aviation Accidents and Incidents with Civil Aircraft, 2019. 53 p.

7. Socio-Technical Decision Support in Air Navigation Systems: Emerging Research and Opportunities / Eds. T. Shmelova, Yu. Sikirda, N. Rizun, A.-B. M. Salem, Yu. Kovalyov. USA, Hershey: IGI Global Publ., 2018. 305 p.

8. Shmelova T., Sikirda Yu., Scarponi C., Chialastri A. Deterministic and Stochastic Models of Decision Making in Air Navigation Socio-Technical System. *CEUR Workshop Proceedings*. Vol-2104. 2018. P. 649–656.

9. Shmelova T., Sikirda Yu. Collaborative Decision-Making Models for UAV Operator's Intelligent Decision Support System in Emergencies. *Artificial Intelligence and Information Systems (ICAIIS 2021)*: Proceedings of the 2nd International Conference, Chongqing, China, May 28-30, 2021. Beijing, China: Association for Computing Machinery (ACM), 2021. P. 1–7.

10. Airport-Collaborative Decision Making: IATA Recommendations. Canada, Montreal: IATA, 2018. 20 p.

USING AN EVOLUTIONARY ALGORITHM IN THE PROBLEM OF OPTIMIZATION OF THE UKRAINIAN POWER GRID

postgraduate student of Zaporizhzhia national university, Ukraine,
Roibul Pavlo

At this time, a war is going on in Ukraine, the enemy is striking at critical infrastructure, the energy system of which is the main goal. After strikes, it is necessary to restore the power system as soon as possible, and then carry out rolling blackouts for repairs, and in these cases it is optimal to use

metaheuristics. Modern network structures are complex territorially extensive systems with a heterogeneous structure. As a result of the complexity and multidimensionality of modern network structures, which in turn are multivariate and multicriteria, and have different options when choosing solutions, the problem of justifying the optimization of network structures in the form of a general research task turns out to be very cumbersome and from a practical point of view has no solution. Given the complexity of the problem, it is advisable to consider its solution with the help of fragmentary models and evolutionary algorithms.

The developed genetic algorithm has fast convergence (about 12 iterations for a scheme of 17 nodes). Differs in high accuracy (found the solution differs from the global optimum by 8.1%. Also works correctly with various network schemes, generation and load structures. Besides, it allows you to determine the optimal network topology, the level of total loss in which is 31% lower than in the original topology. The algorithm can be used in conjunction with RES generation forecasting systems for the day ahead when planning operating modes power associations in order to minimize the costs of covering losses electricity and improve the quality of supplied electricity.

REFERENCES

1. Zhu Y., Guo X., Li J. (2009) An Improved Genetic Algorithm for Power Grid. 2009 Fifth International Conference on Information Assurance and Security. Ottawa. 455–458. https://doi.org/10.1109/IAS.2009.86.

2. Dongmei Z., Pei W., Xu Z. (2014) Reactive Power Optimization by Genetic Algorithm Integrated with Reduced Gradient Method. 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA), 838–841. https://doi.org/10.1109/WARTIA.2014.6976403.

3. Zhang J., Huang T., Zhang H. (2005) The Reactive Power Optimization of Distribution Network Based on an Improved Genetic Algorithm. 2005 IEEE/PES Transmission & Distribution Conference & Exposition: Asia and Pacific. Dalian, 1–4. https://doi.org/10.1109/ TDC.2005.1546960

ФОРМУВАННЯ ЗАПИТІВ КОМПЕТЕНЦІЇ ДО ОНТОЛОГІЇ

Анна Бакурова, Станислав Ведмедєв, Еліна Терещенко
Національний університет «Запорізька політехніка

FORMATION OF COMPETENCE REQUESTS TO ONTOLOGY

Abstract. The problem of forming competence requests to an ontology forms the basis of any methodology for constructing an ontological model, and

is also one of the means of checking the adequacy of an ontology and the purpose of its creation as a knowledge base. The subjectivity of the ontology design process creates problems of formalizing competence requests and checking the completeness of their set. The paper considers an example of generating queries for the ontology of the decision-making process on the application of fertilizers for sunflower cultivation.

Згідно методології проєктування онтології за Грунінгером та Фоксом [1] одним з важливих етапів є формування неформальних запитів компетенції, які формулюються експертами предметної області. При цьому до предметної онтології висувається вимога представляти запити компетенції за термінологією даної предметної області, а відповіді на запитання – за аксіомами і визначеннями. Також система запитів дає семантичну оцінку якості побудованої онтології. Оцінка онтології необхідна для виявлення та виключення помилкових визначень та логічних протиріч.

Неформальні запити формалізуються за допомогою дескриптивної логіки, а аксіоми визначають семантику базових термінів. На завершення побудови онтології в даній методології рекомендовано обґрунтування системи запитів за допомогою теорем про її повноту. В них визначаються умови задоволеності відповідей на запити компетенції. Грунінгер та Фокс [1] сформулювали критерії повноти множини запитів компетенції, що полягають в наступному.

1. Множина запитів є коректною, якщо кожен запит відноситься до певних особливостей онтології.

2. Система запитів вважається повною, якщо експерти стверджують, що їм невідомі додаткові запити.

3. Множина запитів внутрішньо несуперечлива, якщо між ними відсутні конфлікти.

4. Множина запитів піддається перевірці, якщо існує скінчений процес перевірки, чи задовольняє онтологія кожен запит, з розумною вартістю.

5. Система запитів є зрозумілою, якщо кожен з них є зрозумілим користувачам.

6. Множина запитів є однозначною, якщо кожен запит не допускає сумнівів і непорозумінь.

7. Множина запитів є короткою, якщо кожен запит є релевантним та не існує дублюючих запитів.

Такий підхід здається надійним, але розробка онтології є ітеративним процесом, отже система запитів компетенцій може постійно змінюватися. Це є одною з причин того, шо на практиці такий суворий

формалізм використовують не часто, наприклад, в роботі [2] обмежуються формалізацією запитів та підрахунком тих запитів, на які онтологія змогла дати правильні відповіді на думку експертів.

Крім того існують різноманітні інструменти для оцінки онтологій, наприклад, для редактора Protégé існує плагін Ontology Evaluation. Стандартом методологічного супроводу для Protégé є «101 Ontology» [3]. Для побудови запитів в Protégé використовують віджет Queries.

Розглянемо фрагмент формування запитів на прикладі моделі онтології HELIANTHUS [4], яка була розроблена авторами на основі інформації, що надана експертами з Інституту Олійних культур НААН України. Фрагмент онтології показаний на рис.1. Метою створення онтології є підтримка бази знань, що зібрані науковцями в галузі вирощування соняшників за багато років практики та подальше застосування в системах підтримки прийняття рішень.

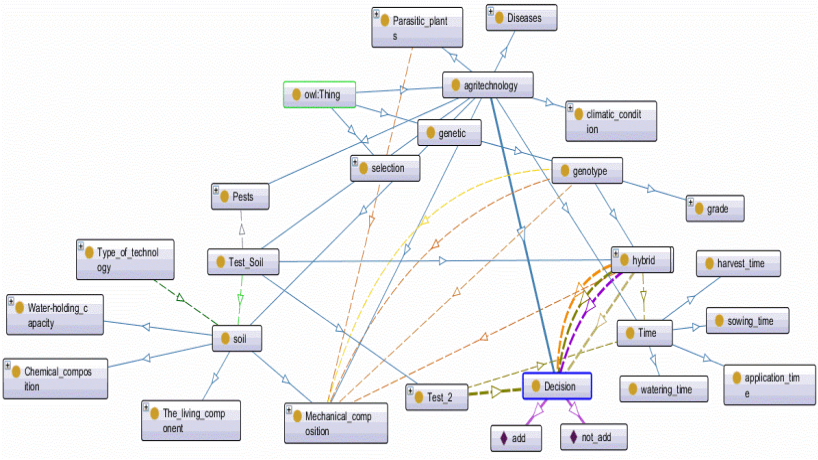


Рис.1. – фрагмент онтології HELIANTHUS

Для побудови онтології HELIANTHUS експертами подано було багато запитів компетенції на природній мові, з яких розберемо два, що пов'язані між собою концептом «стан ґрунту» та відповідають процесу прийняття рішення про внесення добрив:

1. Як можна визначити стан родючості ґрунту під урожай соняшнику?

2. Скільки і коли вносити добрив на гектар соняшнику?

Для формалізації запитів на мові дескриптивної логіки на множині класів онтології введемо наступні позначення:

y – певна методика тестування ґрунту,

x – певний зразок ґрунту, на якому виконується тестування,

t – час забору зразка ґрунту,

d – певна рекомендація,

f – кількість певного добрива на гектар соняшнику.

Класи онтології, що приймають участь у запитах компетенцій прикладу:

SoilTest – методики тестування ґрунту,

Chemical_Composition – хімічний склад ґрунту,

Water-holding_Capacity – водо утримуюча здатність ґрунту,

Living Component – живі компоненти ґрунту,

Mechanical Composition –механічний склад ґрунту,

Decision – рекомендоване рішення,

Genotype – генотип соняшнику (сорт або гібрид),

Amount() – кількість() на гектар соняшнику.

Перелік бінарних відношень між класами онтології у наведених запитах компетенцій:

isSoilTest(y,x) – для зразка ґрунту x використовується метод тестування y;

isTestedChemical_Composition (x,y) – тестування хімічного складу зразка ґрунту x методом тестування y;

isTestedWater-holding_Capacity(x,y) - тестування водо утримуючої здатності зразка ґрунту x методом тестування y;

isTestedMechanical_Composition(x, sand, clay, humus) - тестування механічного складу зразка ґрунту x методом тестування y;

isDecision(Application, Time t) – прийняття рішення про внесення добрив (так, чи ні) в час забору зразка ґрунту t;

isNitrogenRich(f, y) – тест y виявив достатність азоту в добриві f;

isBoronRich(f, y) - тест y виявив достатність бору в добриві f;

isBacteriaRich(f, y) - тест y виявив достатність бактерій в добриві f;

isMostlySuitableFor(f, y) - тест y виявив достатню водо утримуючу здатність ґрунту для добрива f.

Тоді формалізований перший запит компетенцій має вигляд:

$\exists t\ \exists y$ (1m–1.5m) $\forall x$ AND ChemicalComposition(NPK OR minor elements (boron, zinc, magnesium, selenium, etc.)) AND Water-holding Capacity ((aggregate) state, colloidality) OR LivingComponentSoil (bacteria,

worms, symbionts, etc.) OR Mechanical Composition of soils (sand, clay, humus, etc.)→isDecision (Application (YES/NO) AND Time t).

Формалізований запис другого запиту компетенцій:

∃d ∃f ∀Genotype (∃y (1m–1.5m) ∀x (Chemical Composition(NPK OR minor elements (boron, zinc, magnesium, selenium, etc.)) AND LivingComponentSoil (bacteria, worms, symbionts, etc.) → Amount(f; litter OR NPK OR urea).

Отже, онтологія є результатом довільної формалізації суб'єктивних знань фахівця в предметній області. Тому верифікація онтології є необхідною, але система запитів не може бути вичерпною. Формування запитів компетенцій відбувається як на етапі розробки онтології (визначення її структури та обсягу), так і в процесі прийняття рішень. Одним з перспективних напрямів розвитку методології проєктування онтологій та формування запитів компетенцій є графи знань [5], що є метою подальших досліджень.

Література

1. Methodology for the Design and Evaluation of Ontologies: Proceedings of the Workshop on basic Ontological Issues in Knowledge Sharing–IJCAI-95* M Gruninger, M Fox - Montreal, Canada, 1995

2. E. F. Aminu, I. O. Oyefolahan, M. B. Abdullahi, M. T. Salaudeen, MaCOnto: A robust maize crop ontology based on soils, fertilizers and irrigation knowledge, Intelligent Systems with Applications, Volume 16(2022). https://doi.org/10.1016/j.iswa.2022.200125.

3. Natalya F. Noy and Deborah L. McGuinness. Ontology Development 101: A Guide to Creating Your First Ontology. // Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, 2001.

4. Ontological Model of Helianthus Cultivation in Ukrainian Conditions / Anna Bakurova, Kateryna Vedmedeva, Stanyslav Vedmedev, Elina Tereschenko https://ceur-ws.org/Vol-3396/paper10.pdf.

5. Nicolas Hubert, Armelle Brun, Davy Monticolo. New Ontology and Knowledge Graph for University Curriculum Recommendation. ISWC 2022 - The 21st International Semantic Web Conference, Oct 2022, Hangzhou / Virtual, China. ffhal-03768154v2f https://climbtheladder.com/knowledge-graph-interview-questions/

ЗАХИСТ НЕСТРУКТУРОВАНОЇ ІНФОРМАЦІЇ НА МОБІЛЬНОМУ ПРИСТРОЇ

Євген Бровченко
Відкритий міжнародний університет розвитку людини «Україна», Київ

PROTECTION OF UNSTRUCTURED INFORMATION ON MOBILE

Abstract. It is difficult to imagine a modern person without a mobile phone or tablet, which has almost continuous and unlimited access to the Internet or the worldwide global network. The nature and importance of this information may vary, but protection is a priority. The future of protecting unstructured information on mobile devices will be based on a combination of technological innovations and increased user awareness of cybersecurity issues. Some areas of development include the use of artificial intelligence and machine learning; biometric authentication, improvements in biometric authentication technologies, such as facial recognition, fingerprints, can provide more reliable protection of unstructured information; application of blockchain technologies; development of quantum-resistant encryption algorithms; privacy by design; user education and awareness; development of the legislation and regulatory framework, improvement of the legislation and regulatory framework in the field of cyber security will contribute to the creation of a reliable environment for the protection of unstructured information. Securing unstructured information on mobile devices will rely on a combination of technological advances, user education, and collaboration between various parties such as developers, service providers, and lawmakers.

Інформаційні технології (ІТ) та обчислювальні системи з кожним днем все більше стають невід'ємною частиною життя сучасної людини, незамінними помічниками. Розвиток ІТ відбувається зі швидкістю блискавки та має великий вплив на наше суспільство, кожний день ми маємо можливість спостерігати нові технічні та інженерні виноходи. Причому це може бути як певна інтеграція у робочий процес – автоматизація повсякденної рутини, так і частина дозвілля – спілкування та розваги, тощо. Весь комплекс технологій та систем поліпшують та спрошують виконання повсякденних задач, відкривають нові можливості та прибирають кордони. Важко уявити собі сучасну людину без мобільно телефону чи планшету, що має майже безперервний та безлімітний доступом до Інтернету чи всесвітньо глобальної мережі. Сьогодні мобільний пристрій має певний сакральний сенс, він може містити багато різноманітної інформації чи даних. Природа та важливість цієї інформації може бути різною, але захист має пріоритетне значення. А

враховуючи динаміку розвиту та невизначеність деяких процесів, перед інженерами та розробниками стоїть важке та важливе завдавання по захисту інформаційних систем та забезпеченню надійної роботи. До певної мірі мобільний пристрій можна вважати найбільш проблематичною складовою. Цей пристрій має ряд переваг та особливостей – власне це й роблять його уразливим перед потенційними зловмисниками та й дії користувача можуть спричинити ряд проблем. Доповідь присвячена проблематиці використання мобільних пристроїв та захисту неструктурованої інформації. Захист неструктурованої інформації на мобільному пристрої є важливим завданням з точки зору кібербезпеки. Неструктурована інформація може бути у вигляді текстових документів, електронних листів, фотографій, відео, записів голосу та іншого контенту, який зберігається на мобільних пристроях. До основних проблем можна віднести: втрата або крадіжка мобільного пристрою; вразливості операційної системи та програмного забезпечення; недостатній рівень шифрування та недостатній рівень складності паролів; нехтування кібергігієною. Існуючи інженерно-архітектурні рішення не завжди виконують поставлену задачу та можуть мати проблеми з інтеграцією. Не слід забувати й про економічну складову. Аналіз прикладної області та результати дослідження говорять про те що захист інформації в цілому є важливим та є попит на швидкі та ефективні рішення щодо безпеки інформаційних систем в цілому та мобільного пристрою зокрема. Лише комплексний підхід може забезпечити ефективне рішення.

Неструктурована інформація - це дані, які не мають чіткої структури або не відповідають певному формату, що ускладнює їх обробку та аналіз. Вона може включати текстові документи, електронні листи, аудіо- та відеофайли, зображення, веб-сторінки, соціальні мережі та інші форми даних. У порівнянні зі структурованою інформацією, яка зазвичай зберігається у реляційних базах даних або таблицях, неструктуровані дані не підпорядковуються певному схемі або моделі. Однак важливим аспектом є захист неструктурованої інформації, оскільки ці дані можуть містити конфіденційну або чутливу інформацію, яка повинна залишатися в безпеці. Зокрема, на мобільних пристроях неструктурована інформація може бути особливо вразливою до крадіжки та злому через відносно низький рівень безпеки цих пристроїв порівняно зі стаціонарними комп'ютерами. Тому розробка та впровадження ефективних методів захисту є дуже важливою.

До основних загроз можна віднести: втрата пристрою, злом акаунтів, шкідливе програмне забезпечення, небезпечні мережі Wi-Fi, фізичний доступ, віддалений доступ, сумнівні хмарні сервіси.

Методи захисту: шифрування даних, автентифікація користувачів, резервне копіювання і відновлення даних, оновлення програмного забезпечення, встановлення антивірусного ПЗ, блокування небезпечних додатків, використання захищених мереж, управління доступом до додатків, застосування політик безпеки, віддалення управлянні даними. Також є важливим використовувати програмні рішення для захисту неструктурованої інформації: антивіруси, файрволи, програми для шифрування та управління паролями. До можливих прикладів можна віднести наступне: шифрування даних: програми для шифрування даних, такі як VeraCrypt, AxCrypt, або BitLocker, дозволяють шифрувати файли та теки на пристрої, забезпечуючи захист інформації від несанкціонованого доступу; віртуальні приватні мережі (VPN): використання VPN-служб, таких як ExpressVPN, NordVPN або CyberGhost, дозволяє забезпечити анонімність онлайн-активності, шифрувати передачу даних та обмежити доступ до чутливої інформації; захист від вірусів та шкідливого ПЗ: антивірусні програми, такі як Norton або McAfee, забезпечують захист від вірусів, троянських коней, шпигунського та шкідливого програмного забезпечення; менеджери паролів: менеджери паролів, такі як LastPass, Dashlane або 1Password, допомагають зберігати та генерувати складні паролі, забезпечуючи захист різних аккаунтів користувача; резервне копіювання даних: рішення для резервного копіювання даних, такі як Acronis, Veeam або Carbonite, дозволяють створювати резервні копії важливої інформації, забезпечуючи відновлення даних в разі втрати або пошкодження; управління доступом та співпраця: рішення для управління доступом та співпраці, такі як Microsoft Azure Information Protection, Box або Google Workspace, дозволяють контролювати доступ до чутливої інформації, налаштовувати дозволи та співпрацювати над документами в безпечному середовищі . Ці програмні рішення допомагають забезпечити захист неструктурованої інформації на мобільних пристроях та комп'ютерах, а також допомагають користувачам контролювати доступ до своїх даних. Вибір відповідного рішення для захисту інформації залежить від конкретних потреб користувача та його середовища роботи.

Не слід забувати про роль користувача: розуміння важливості захисту даних, використання сильних паролів, регулярне оновлення ПЗ а головне освіта з питань безпеки.

Майбутнє захисту неструктурованої інформації на мобільних пристроях буде засноване на поєднанні технологічних нововведень та підвищенні обізнаності користувачів з питань кібербезпеки. Деякі напрямки розвитку включають: Застосування штучного інтелекту (AI) та машинного навчання: AI може допомогти в розробці розширених

механізмів захисту, що автоматично виявляють аномальні або підозрілі дії та блокують потенційні загрози. Біометрична автентифікація: покращення технологій біометричної автентифікації, таких як розпізнавання обличчя, відбитків пальців, можуть забезпечити більш надійний захист неструктурованої інформації. Застосування блокчейн-технологій: блокчейн може відігравати важливу роль у забезпеченні безпеки інформації, оскільки він дозволяє зберігати дані розподіллено та захищено від змін. Розвиток квантово-стійких алгоритмів шифрування: з розвитком квантових комп'ютерів створюється потреба в нових алгоритмах шифрування, які можуть витримати потужність квантового злому. Приватність за замовчуванням (Privacy by Design): проектування мобільних пристроїв та застосунків з врахуванням приватності користувачів сприятиме захисту неструктурованої інформації на мобільних пристроях. Освіта та обізнаність: підвищення обізнаності користувачів щодо кібербезпеки та навчання їх ефективним методам захисту інформації на мобільних пристроях допоможе зменшити ризик виникнення інцидентів, пов'язаних з безпекою. Розвиток законодавства та нормативної бази: вдосконалення законодавства та нормативної бази у галузі кібербезпеки сприятиме створенню надійного середовища для захисту неструктурованої інформації. Захист неструктурованої інформації на мобільних пристроях базується на комбінації технологічного прогресу, навчання користувачів та співпраці між різними сторонами, такими як розробники, постачальники послуг та законодавці. Забезпечення безпеки та приватності даних в цифровому світі залишається пріоритетом, оскільки кількість мобільних пристроїв та їх використання продовжують зростати. Сучасній користувач мобільних пристроїв має певні вимоги що спарюються на ергономіку та рівень інформаційної обізнаності. Розвиток галузі в цілому вимагає швидких, надійних та ефективних рішень що потрапляють не лише в економічну площину а й площину кібербезпеки. Використання мобільних пристроїв може адаптуватися до швидких сучасних потреб безпеки організації та може бути легко налаштований відповідно до конкретних вимог. Мобільний пристрій є зручним і доступним для багатьох користувачів, він зазвичай знаходиться поруч з користувачем і може використовуватися як додатковий елемент безпеки.

Література

1.Бровченко, Є. М., Самарай, В. П., Даценко, І. П., Павленко, В. І. (2023). Мобільний пристрій як частина адаптивної кейс-менеджмент системи. Інфокомунікаційні та комп’ютерні технології, 2(04), 157-163.

https://doi.org/10.36994/2788-5518-2022-02-04-18 (дата звернення: 25.05.2023)

2.Gámez, N., García-Sánchez, P., & Pérez, J. (2020). User authentication using mobile devices: A review. Information Fusion, 55, 153-170.

3.Kang, Y., Zhang, N., Zhu, X., & Huang, Y. (2021). A novel authentication mechanism based on dynamic cognitive keystroke on mobile devices. IEEE Access, 9, 32098-32107.

4.Yavuz, A. G., Akgu¨n, M. A mobile-based multi-factor authentication system using biometric and location information. International Journal of Computer Science and Network Security, 17(5), 181-187, (2017).

ВПЛИВ ТРАНЗИТИВНИХ ДУГ ГРАФА НА ОПТИМАЛЬНІСТЬ В УЗАГАЛЬНЕНІЙ ЗАДАЧІ ПАРАЛЕЛЬНОГО УПОРЯДКУВАННЯ

Поліна Василькович, Валентина Турчина
Дніпропетровський національний університет імені Олеся Гончара

INFLUENCE OF TRANSITIVE GRAPH ARCS ON OPTIMALITY IN THE GENERALIZED PARALLEL ARRANGEMENT PROBLEM

Abstract. The work considers one of the well-known problems of parallel ordering of the vertices of a digraph. These tasks emerge with determining the optimal order of execution of a finite set of tasks, when certain technological restrictions are imposed on the order. Some cases are considered for the well-known distribution algorithm when the presence of transitive arcs affects the accuracy of the obtained result. Also, the rule for adding such arcs to the graph is proposed.

В даній роботі розглядється одна з відомих задач, що виникають при призначанні працівників на скінчену множину робіт за наявності технологічних обмежень на порядок їх виконання так, щоб загальний час виконання був мінімальним. Зручними інструментами, що задають частковий порядок, можуть бути орієнтовані ациклічні графи.

При розв’язанні задач в класичній постановці виходять з припущення, що будь-яке завдання може виконувати будь-який виконавець, а час виконання кожного завдання однаковий [1].

Розглядається наступне узагальнення однієї із класичних задач побудови оптимальних упорядкувань [2].

По заданому графу $G = \{V, U\}$ і вектору $h = (h_1, ..., h_r)$, $r \le n$, побудувати паралельне упорядкування S мінімальної довжини таке, що $|S[i]| \le h_i$, $i = 1,r$, і не порушувались технологічні обмеження.

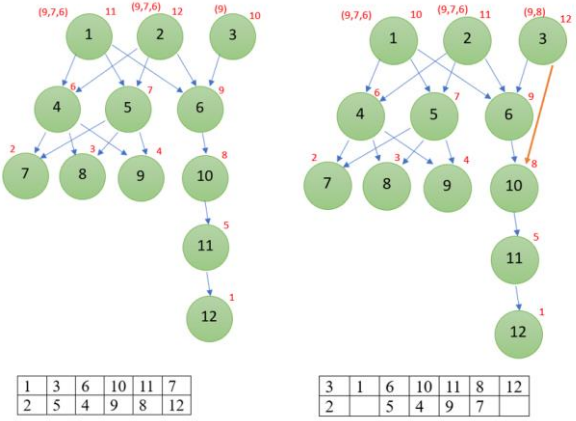
Ця задача, як і класичні задачі, відноситься до класу NP-повних [3].

Серед точних методів, що мають поліномальну складність, для класичної задачі, що відповідає випадку, коли всі значення $h_i = 2$ ефективним є алгоритм заснований на лексикографічному помічуванні вершин [4]. Але в цьому випадку при наявності транзитивних дуг в графі розв'язок може бути наближеним. Це проілюстровано на наступному прикладі з рис. 1.

Тобто ми бачимо, що вилучення транзитивної дуги, яка для даної постановки задачі не є інформативною, оскільки технологічні обмеження на порядок не порушуються, впливає на точність розв'язку.

Виникає питання чи будуть транзитивні дуги впливати на оптимальність розв'язку і для узагальненої задачі, яку ми розглядаємо.

Наявність транзитивних дуг не впливає на порядок, але може впливати на мітки, що отримують вершини графу.



1	3	6	10	11	7
2	5	4	9	8	12

3	1	6	10	11	8	12
2		5	4	9	7	

Рис. 1 – *Приклад 1. Неоптимальне упорядкування після додаванні однієї транзитивної дуги при h = 2*

Іноді у графах на етапі розставлення поміток, виникає неоднозначність вибору для декількох вершин одного рівня. Як виявилося, для узагальненої задачі додавання транзитивних дуг дозволяє позбутися цієї неоднозначності та отримати оптимальне упорядкування при тій самій стратегії помічування вершин.

Розглянемо граф із рис. 2 при $h = (1,3,1,1,1)$.

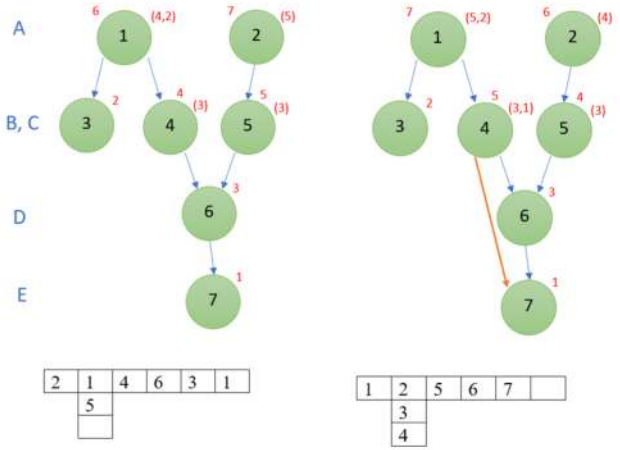


Рис. 2 – *Приклад 2. Побудова оптимального упорядкування при додаванні однієї транзитивної дуги*

Спробуємо прослідкувати залежність у таких випадках.

На прикладі 2 (рис. 2) $h_2 = 3$. Призначити три вершини (2, 3, 4) на друге місце можна тільки якщо на перше місце встановити вершину під номером 1, але у графі без транзитивних дуг на це місце стновиться вершина 2, оскільки має більшу помітку. Така ситуація виникає через те, що вершини 4 та 5 мають однакові лексикографічні послідовності. Додавання однієї транзитивної дуги з вершини 4 до вершини 7 дозволяє уникнути неоднозначності вибору поміток.

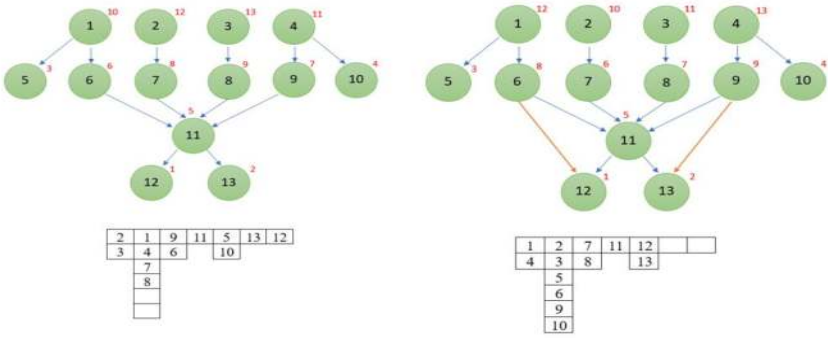


Рис. 3 - *Приклад 3. Додавання двох транзитивних дуг приводить до оптимального упорядкування*

Розглянемо ще один приклад такого впливу транзитивних дуг. На рис. 3 можна побачити, що додавання однієї транзитивної дуги було б недостатнім для того, щоб позбутися неоднозначності. Тут $h_1 = 2$ і, щоб побудувати оптимальне упорядкування, необхідно поставити вершини 2 і 3 на перше місце, тому необхідно додати дві транзитивні дуги, що виходять з вершин 6 і 9.

Тепер, виходячи з цих прикладів, наведемо деякі спостереження та узагальнемо їх у вигляді правила.

Під рівнем будемо вважати умовну лінію, що визначає положення множини вершин, на ній розташованих, відносно критичних шляхів графу.

У графі повинно бути не менше 4 рівнів. Перший з них будемо позначати літерою *A,* а V_1 – множина вершин на ньому. На ньому розташовані вершини, порядок призначення на місця яких буде суттєво впливати на оптимальність упорядкування. Другий рівень *B* іде одразу за першим, V_2 – множина вершин рівня. На рівні *B* окрім вершин, що лежать на критичних шляхах, повинно бути $|V_2| - |V_1|$ вершин, які не мають вихідних дуг і в які входять дуги з вершин рівня *A*. Третій – *C*, це рівень на якому розташовуються вершини, для яких виникає неоднозначність вибору поміток. Зауважимо, що рівні *B* і *C* можуть співпадати. На четвертому рівні *D* повинна бути вершина, у яку входять дуги з вершин рівня *C*, для яких існує неоднозначність вибору. На останньому рівні *E* стоять вершини, у які входять транзитивні дуги. Також зауважимо, що між *B* і *C* та *D* і *E* можуть стояти ще декілька інших рівнів. Із цього випливає наступне правило.

Правило 1.

Нехай в упорядкуванні на місце i потрібно поставити h_i вершин з рівня *A* ($h_i < |V_1|$, $h_i < h_{i+1}$), а на $i+1$ місце $|V_1| - h_i$ вершин рівня *A*, $|V_2| - |V_1|$ вершин рівня *B*, що не мають вихідних дуг, та h_i вершин рівня *B*, що стоять на критичних шляхах, ($h_{i+1} = |V_1| - h_i + |V_2| - |V_1| + h_i = |V_2|$). За таких умов, може виникнути ситуація, коли h_i вершин рівня *A* повинні мати пріоритетність над іншими вершинами цього рівня, щоб відкрити шляхи для $|V_2| - |V_1| + h_i$ вершин рівня *B*. Цю пріоритетність їм забезпечить додавання h_i транзитивних дуг з вершин рівня *C* до вершин рівня *E*, що стоять на тих самих шляхах, що і вершини, порядок призначення на місця яких буде суттєво впливати на оптимальність упорядкування.

Повернемось до прикладу 2. На рівні *A* стоять вершини 1 і 2, $i = 1$, $h_i = 1$, $|V_1| = 2$, $|V_2| = 3$, $h_{i+1} = 3$. Рівні *B* та *C* співпадають. Для того, щоб відкрити шлях до вершини 3, що не має вихідних дуг, та вершини 4 рівня *B*, що лежить на критичному шляху, треба забезпечити пріоритетність

вершини 1, забезпечивши їй перше місце. Це можна зробити додавши транзитивну дугу з вершини рівня *C*, що стоїть на одному шляху з вершиною 1, до рівня *E*. Такою вершиною є вершина 4.

У прикладі 3 аналогічно пріоритетність вершин 1 і 4 встановлюється додаванням дуг з вершин 6 і 9, що стоять на тих самих шляхах.

В наступному прикладі ситуація є менш прозорою.

На прикладі 4 (рис. 4) можна спостерігати, як для встановлення пріоритетності вершин довелося застосувати правило 1 два рази. Маємо наступний вектор h = (2, 5, 1, 4, 1, 1, 1, 1). Так при i = 1, h_1 = 2, встановлюється пріоритетність двох вершин цього рівня, однією з яких обов'язково є вершина 3. Для i = 3, h_3 = 1, встановлюється пріоритетність вершини 5.

Зауважимо, що для кожної транзитивної дуги рівні визначаються окремо. Головне, на що треба звернути увагу, є те, що кількість доданих транзитивних дуг у графі буде рівною сумі h_i, які обираються при кожному застосуванні правила 1 ($h_1 + h_3 = 3$).

Таким чином, справедливе наступне твердження:

Якщо в упорядкуванні на місце i повинно стати h_i вершин з рівня A ($h_i < |V_1|$, $h_i < h_{i+1}$), а на $i+1$ місце $|V_1| - h_i$ вершин рівня A, $|V_2| - |V_1|$ вершин рівня B, що не мають вихідних дуг, та h_i вершин рівня B, що стоять на критичних шляхах, то існує таке додавання транзитивних дуг між відповідними рівнями, яке вплине на пріоритетність вершин попередніх рівнів та відповідно на оптимальність упорядкування.

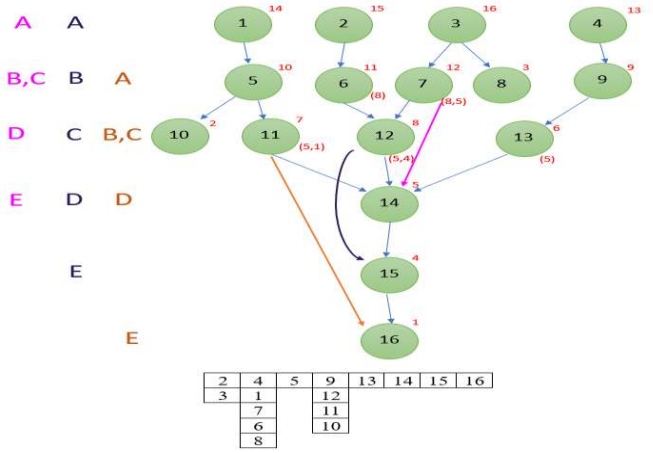


Рис. 4 - *Приклад 4. Застосування правила 1 забезпечує побудову оптимального упорядкування*

У подальшому потребує досліждення питання впливу властивостей обмежуючої послідовності h на кількість транзитивних дуг, які покращують наближені розв'язки.

Література.
1. Василькович П. О., Турчина В. А. Алгоритми побудови паралельних упорядкувань, засновані на аналізі структури графів. *International scientific journal «Grail of Science»*. 2022. №12-13. С. 300–306.
2. Бурдюк, В. Я., Турчина В. А. Алгоритмы параллельного упорядочения: Учебное пособие. Днепропетровск: ДГУ, 1985. 85 с.
3. Ullman J. D. NP-complete scheduling problems. *J. Comput. Systems Sci*., 1975. P. 384– 393.
4. Computer and job-shop scheduling theory / ed. by E.G. Coffman. John Wiley & Sons, 1976. 336 p.

ПРО ЗВ'ЯЗНІСТЬ КВАЗІВИПАДКОВИХ ГРАФІВ

Глухов О.Д.
glukhov07@gmail.com,

Національний авіаційний університет

ON THE CONNECTIVITY OF QUASI-RANDOM GRAPHS

Abstract. In this paper, we consider a new approach to estimating the connectivity of quasi-random graphs based on the concept of a multispanner and of graph

Квазівипадкові графи.

Квазівипадкові графи є моделями дискретних систем, структура яких може змінюватись внаслідок випадкового розриву частини зв'язків [1-2].

Означення 1. Нехай $G = G_n$ - зв'язний граф на n вершинах з множиною вершин G^0 і множиною ребер G_n^1 , $|G^0| = n$, $|G^1| = m$, квазівипадковим графом на основі графа G називається граф $G(p)$ з

множиною $G^0(p) = G^0$ вершин і з випадковою множиною $U = G^1(p)$ ребер для якої виконуються умови:

$$Prob(u \in U) = p, \text{ якщо } u \in G^1;$$
$$Prob(u \in U) = 0, \text{ якщо } u \notin G^1.$$

Будемо розглядати задачу оцінки ймовірності зв'язності квазівипадкового графа $G(p)$ побудованого на основі - 3-реберно зв'язний графа G.

Мультикаркаси графа і С-поліноми.

Нехай G - зв'язний граф, $U \subseteq G^1$ - деяка множина його ребер, через $G[U]$ будемо позначати мінімальний підграф графа G з множиною ребер $(G[U])^1 = U$.

Означення 2. Мультикаркасом графа G будемо називати сімейство множин $\{E_j\}_{j=1}^s$, яке задовольняє наступним умовам:

(1) $\forall j \; E_j \subseteq G^1$

(2) Якщо множина W ребер графа G задовольняє умові $\forall j \; W \cap E_j \neq \varnothing$, то граф $G[W]$ буде зв'язним факторграфом графа G (зв'язним підграфом, який містить усі вершини даного графа).

Лема1. Сімейство множин $M = \{E_j\}_{j=1}^s$ буде мультикаркасом графа G тоді і тільки тоді, коли для будь-якого реберного розріза U знайдеться таке E_k, що $E_j \subseteq U$.

Означення 3. Нехай $M = \{E_j\}_{j=1}^s$ - деякий мультикаркас графа G, $\xi_k(M) = \left| E \in M : \left| E \right| = k \right|$. С-поліномом або зв'язністним поліномом мультикаркаса графа G назвемо поліном, визначений наступним чином:

$$C(G,M,x)=\sum_k \xi_k x^k .$$

Приклади мультикаркасів.

Нехай G -зв'язний граф, T - його остовне дерево (каркас) з множиною ребер $T^1=\{u_k\}_1^{n-1}$ $(n=|G^0|)$. Тоді очевидно, що $M=\{E_k\}_1^{n-1}$, $E_k=\{u_k\}$, є мультикаркасом графа G.

Інший приклад мультикаркаса. $M=\{U_k\}_1^s$, де $\{U_k\}_1^s$ множина усіх його бондів (мінімільних по включенню реберних розрізів) зв'зного графа G .

Важливість C - полінома випливає з наступного твердження.

Теорема 1. Якщо G зв'язний граф, M - деякий його мультикаркас, то для ймовірності P зв'язності квазівипадкового графа $G(p)$ на основі графа G має місце наступна оцінка:

$$P\geq 1-C(G,M,q)\text{, де } q=1-p.$$

Регулярні вкладення графів в 2-многовиди.

Означення 4. 2-кліткове вкладення графа G в 2-многовид називається регулярним, якщо замикання кожної 2-клітки гомеоморфно замкненому диску, а замикання двох різних кліток не можуть мати більше одного спільного ребра графа. [3].

Нескладно довести наступні твердження.

Лема 2. 2-кліткове вкладення графа G в 2-многовид регулярне тоді і тільки тоді, коли дуальний граф G^* є простим графом (графом без петель і кратних ребер).

Зауважимо, що для зручності ми вважаємо, що $(G^*)^1=G^1$.

Лема 3. Кожному реберному розрізу U регулярно вкладеного графа G відповідає набір $\{Z_j\}_1^k$ простих циклів дуального графа G^* , який задовольняє умовам: $\bigcup_{j=1}^{k} Z_j^1=U$.

Лема 4. Нехай граф G регулярно вкладений в деякий 2-многовид , а G^* - його дуальний граф, $\{Z_k\}_1^s$ усіх простих циклів графа G^*. Тоді множина $M = \{Z_k^1\}_1^s$ є мультикаркасом графа G.

Лема 5 [4]. Якщо H - простий (без петель і кратних ребер) 3-зв'язний граф, $|H^1| = m$, то числа $c_k(H)$ k-циклів має місце наступна нерівність: $c_k(H) \le \frac{1}{2k}(2m)^{k/2}$.

Наступні твердження дозволяють оцінювати зв'язність широкого класу квазівипадкових графів, зокрема 3-зв'язних планарних графів.

Теорема 2. Якщо зв'язний граф G допускає регулярне вкладення в деякий 2-многовид , то для ймовірності P зв'язності квазівипадкового графа $G(p)$ на основі графа G має місце наступна оцінка:

$$P \ge 1 - \sum_{k=1}^{n} \frac{1}{2k}(2m)^{k/2} q^k, \text{ де } q = 1 - p.$$

Наслідок. Якщо зв'язний граф G допускає регулярне вкладення в деякий 2-многовид , а $G(p)$ квазівипадковий графа на основі графа G, $p = 1 - \alpha(m)/\sqrt{m}$, де $\alpha(m) = o(1)$, то граф $G(p)$ буде зв'язним з ймовірністю $1 - o(1)$.

Література

1. Diestel R. Graph Theory. –Springer-Verlag, 2000. –322 P.
2. Глухов А.Д. Квазислучайные графы и структурная устойчивость сложных дискретных систем. –Электрон. моделирование, 2016 , 38, №5, с.35–41.
3. Mohar B., Thomassen C. Graphs on Surfaces. – Baltimore: John Hopkins Univ. Press, 2001. – 291 P.
4. Глухов О.Д. Про зв'язність планарних pr- графів пуассонівського типу// II Український математичний конгрес, 27-29 серпня 2009 р.: тези доп. – К., 2009. – режим доступу: http://www.imath. kiev.ua/~congress2009.

ПРАКТИЧНІ КЕЙСИ ЗАСТОСУВАННЯ КЛАСТЕРНОГО АНАЛІЗУ

Інна Горват, Євген Кондрук, Владислав Нерода
Ужгородський національний університет

PRACTICAL CASES OF CLUSTER ANALYSIS

Abstract. The article discusses practical use cases of cluster analysis, which is a powerful tool for identifying similar groups of objects within a dataset. Cluster analysis is widely applied in various fields, including marketing, medicine, social sciences, and information technology.

Кластерний аналіз є потужним інструментом для організації та аналізу даних у багатьох сферах діяльності. Він дозволяє групувати схожі об'єкти разом і виділяти внутрішні закономірності та структуру даних. Практичне застосування кластерного аналізу охоплює широкий спектр сфер, включаючи науку, бізнес, медицину та соціальні науки. Метою роботи є розглянути деякі практичні кейси застосування кластерного аналізу і їх вплив на відповідні галузі.

Одним з популярних прикладів використання кластерного аналізу є сегментація споживачів у сфері маркетингу. Компанії використовують кластерний аналіз для групування клієнтів на основі спільних характеристик, таких як поведінка, інтереси, демографічні дані тощо. Це дозволяє компаніям створити більш ефективні маркетингові стратегії, налаштувати специфічні пропозиції для кожної групи споживачів та поліпшити задоволення клієнтів. У [1] проведено аналіз методичних підходів щодо застосування методів кластерного аналізу для виконання конкретних практичних завдань, що повсякденно стоять перед основними ринковими суб’єктами господарювання.

У сфері медицини кластерний аналіз може бути застосований для класифікації пацієнтів на основі медичних даних, таких як симптоми, лікування або генетична інформація. Це допомагає лікарям уточнити діагнози, розробити індивідуальні підходи до лікування та прогнозувати ризики. Наприклад, застосування кластерного аналізу в онкології може допомогти встановити підтипи раку, що дає змогу призначити більш ефективне та персоналізоване лікування для кожного пацієнта. Кластерний аналіз може бути використаний, також, для розробки системи підтримки прийняття рішень лікаря дієтолога. Шляхом аналізу даних про пацієнтів, таких як харчові звички, стан здоров'я, фізичні характеристики та інші фактори, можна виявити групи пацієнтів зі схожими потребами та ризиками. В [2] використали підхід автоматичного групування критеріїв

(цілей) введення дієти пацієнтам із подальшою редукцією критеріального простору задачі прийняття рішень. Даний підхід був використаний для реалізації системи підтримки прийняття рішень лікаря дієтолога.

Кластерний аналіз є важливим інструментом для аналізу соціальних мереж та поведінки користувачів в Інтернеті. Використовуючи цей метод, можна виділити групи користувачів з подібними інтересами, поведінкою та думками. Наприклад, у статті [3] використовувався кластерний аналіз та аналіз багатовимірного масштабування для дослідження основних аспектів Facebook. Це дозволило проаналізувати поведінку користувачів, вплив соціальних мереж, а також наслідки цього впливу на індивідів, групи, підприємства та суспільство в цілому.

Кластерний аналіз також може бути використаний для персоналізації рекламних пропозицій, рекомендацій та створення спільнот із спільними інтересами. Наприклад, платформи для стрімінгу музики та відео використовують кластерний аналіз для рекомендацій музики або фільмів на основі схожих смаків користувачів. Крім того, кластерний аналіз може бути застосований для аналізу різних опитувань з метою виділення груп учасників з подібними думками та досвідом. У статті [4] описується дослідження однієї загальноосвітньої школи у Фінляндії під час пандемії COVID-19. За допомогою описової статистики та кластерного аналізу були проаналізовані дані опитувань учнів і вчителів. Результати свідчать про успішне впровадження дистанційного навчання, але завдяки відкритим коментарям та кластерному аналізу було виявлено багато проблем.

Кластерний аналіз також знайшов своє застосування у сфері географічних досліджень та планування міського розвитку. Він дозволяє групувати географічні об'єкти, такі як райони міста або території, на основі подібних характеристик, таких як населення, інфраструктура, соціально-економічні показники та інше. Це допомагає у розумінні розподілу та характеристикам різних районів, розробці стратегій розвитку та плануванні міського середовища. Кластерний аналіз може бути використаний і для вивчення інтенсивності міграційного руху населення в регіонах України, дозволяючи ідентифікувати групи регіонів зі схожими характеристиками міграційного потоку. Зокрема, в [5] проводиться групування об'єктів на базі методу однорівневої кластеризації та довжинної міри подібності до дослідження інтенсивності міграційного руху населення в регіонах України. Використовуючи

отримані результати, було інтерпретовано стан міграційного руху у різних регіонах України.

Вагоме значення має кластерний аналіз і у геології. Під час геологічного пошуку корисних копалин та видобування рудних тіл регулярно збирають великі набори даних. Ці набори даних часто досягають великого розміру або складності, це ускладнює їх інтерпретацію. Застосування кластерного аналізу у геології розглянуто у [6].

Затори стали поширеною проблемою у містах по всьому світу внаслідок прискореного процесу урбанізації. У [7] розширюється концепція ситуації заторів, розглядаються дві аспекти: стан і тенденція. Пропонується метод визначення рівня стану та типу тенденції заторів на головних дорогах у містах з холодним кліматом на основі кластеризації щільності, ієрархічної кластеризації та нечіткої кластеризації K-середніх.

Методи кластеризації знайшли своє застосування і у криміналістиці. Для допомоги правоохоронним органам у розкритті злочинів було розроблено кілька систем профілювання злочинців, але методи, які використовуються в більшості систем, не дають можливості кластеризувати злочинців на основі їхніх поведінкових характеристик. У [8] розглядаються різні методи кластеризації, що використовуються в злочинному профілюванні, а потім вибирається один алгоритм нечіткої кластеризації і два алгоритми жорсткої кластеризації. Обрані алгоритми були розроблені та протестовані на реальних даних для створення "профілів" злочинної діяльності та поведінки злочинців.

Використання методів кластеризації у останні роки значно збільшилося. Головною метою дослідників було вивчення та оптимізація методів аналізу, спрямованих на вирішення астрономічних проблем. У [9] презентується застосування методу кластеризації до двох випадкових досліджень, отриманих з радіо- та гамма-астрономії. У першому випадку на меті вивчення природних групувань, у другому випадку - дослідити структури малого масштабу світлової кривої за допомогою фазових чисел. У обох випадках основою аналізу є побудова мінімального покриваючого дерева на основі евклідової відстані.

Одним із викликів, пов'язаних з кластерним аналізом, є вибір відповідного алгоритму кластеризації та налагодження параметрів. Різні алгоритми мають свої переваги та обмеження, і вибір підходящого алгоритму залежить від конкретної задачі та характеристик даних.

Крім того, при роботі з кластерним аналізом важливо враховувати якість та достовірність вхідних даних. Неправильні або неточні дані можуть призвести до некоректних результатів кластеризації. Тому перед застосуванням кластерного аналізу важливо провести попередню

підготовку даних, включаючи обробку пропущених значень, нормалізацію та видалення випадкових шумів.

Узагальнюючи, практичні кейси застосування кластерного аналізу охоплюють багато сфер діяльності і дозволяють отримати цінні інсайти з наборів даних. Він допомагає виявляти схожі групи об'єктів, розуміти внутрішні закономірності та залежності, робити усвідомлені рішення та покращувати результати у різних галузях, включаючи маркетинг, медицину, соціальні науки та бізнес.

Література

1. Соколова Л. В. Використання методів кластерного аналізу у практичній діяльності підприємства / Л. В. Соколова, Г. М. Верясова, О. Є. Соколов. // Вісник національного університету «Львівська політехніка». – 2011. – №720. – С. 240–247.

2. Кондрук Н. Е. Системи підтримки прийняття рішень для автоматизованого складання дієт / Н.Е. Кондрук // Управління розвитком складних систем. - 2015. - Вип. 23. - С. 92-96.

3. Shiau W. Examining the core knowledge on facebook / W. Shiau, Y. Dwivedi, H. Lai. // nternational Journal ofInformation Management. – 2018. – №43. – С. 52–63.

4. Niemi H. A case study of students' and teachers' perceptions in a Finnish high school during the COVID pandemic / H. Niemi, P. Kousa. // International Journal of Technology in Education and Science (IJTES). – 2020. – №4. – С. 352–369.

5. Кондрук, Н. Е. Використання довжинної міри подібності в задачах кластеризації /Н. Е. Кондрук //Радіоелектроніка, інформатика, управління. – 2018. – № 3 (46) – С. 98-105. DOI: 10.15588/1607-3274-2018-3-11.

6. Application of principal component analysis and cluster analysis to mineral exploration and mine geology / [M. Gazley, K. Collins, J. Roberston та ін.]. // AusIMM New Zealand Branch Annual Conference. – 2015. – С. 131–139.

7. Method for Identifying the Traffic Congestion Situation of the Main Road in Cold-Climate Cities Based on the Clustering Analysis Algorithm / [P. Yulong, C. Xiaoxi, L. Jie та ін.]. // Sustainability. – 2021. – №13. – С. 1941.

8. Adeyiga J. A comparative analysis of selected clustering algorithms for criminal profiling. / J. A. Adeyiga, S. O. Olabiyisi, E. O. Omidiora3. // Nigerian Journal of Technology (NIJOTECH). – 2020. – №39. – С. 464–471.

9. Application of cluster analysis to astronomical data / R. Buccheri, V. Gesu, M. Maccarone, B. Sacco // Centre de Données astronomiques de Strasbourg. – 2014.

НЕЧІТКА ПОСТАНОВКА ЗАДАЧІ СЕГМЕНТУВАННЯ РИНКУ ПРАЦІ

Леонід Гуляницький, Ірина Рясна
Інститут кібернетики імені В. М. Глушкова НАН України

FUZZY STATEMENT OF LABOR MARKET SEGMANTATION PROBLEM

Abstract. The problem of segmentation of a set of objects under quantitative and qualitative information is considered. Based on the theory of fuzzy sets, the problem is analyzed and its fuzzy model is proposed. The formal concepts of segment and segmentation of the first and second kind are introduced. An approach to constructing a labor market segmentation model based on the concepts of a fuzzy cluster and a fuzzy similarity relation, determined using the linguistic correlation coefficient and fuzzy combinatorial configurations, is proposed.

Сегментування ринків праці в економічній сфері – це процес розбиття множини суб'єктів ринків на складові сегменти, які визначаються стійкими характеристиками. За наявності вербального визначення значень загальних характеристик різних сегментів отримаємо ситуацію, коли суб'єкти ринків можуть одночасно належати кільком сегментам. При цьому виникає задача розбиття заданої множини суб'єктів на підмножини, що не перетинаються. Розрізняючись між собою, сегменти ринку праці мають загальні риси. Поглиблення відмінності в межах кожного сегменту, поява нових якісних і кількісних характеристик, схожі видозміни в різних сегментах неможливо пояснити на підставі дії лише одного чинника, який визначається, як правило, вербально і не має чисельного виміру [1]. Необхідно ураховувати низку параметрів. Спосіб включення послуг з праці суб'єктів ринку праці в сукупний процес виробничої діяльності може визначатись у багатьох альтернативних формах. Наприклад, у вигляді найму на умовах неповного робочого дня, часткової зайнятості, роботи за сумісництвом, тимчасових робіт тощо. Спосіб координації праці заснований на узгодженні трудових процесів шляхом їхнього підпорядкування

потребам капіталу і кінцевим запитам споживачів. Спосіб оцінки праці має багато форм та визначається за різними видами утворення трудових прибутків. Крім того, необхідно ураховувати неминучість модифікації сегментувань ринку праці у зв'язку з динамічністю умов соціально-економічного розвитку держави [2].

Відзначимо такі особливості задачі сегментування ринку праці:

– межі сегментів чітко не визначені;

– існує взаємозалежність сегментів;

– наявність у сегментів загальних і унікальних характеристик;

– формування сегментів як відносно окремих систем;

– кількісне і якісне визначення характеристик суб'єктів ринку праці;

– визначення характеристик за різними шкалами;

– взаємозалежність характеристик суб'єктів ринку праці;

– необхідність урахування кількох критеріїв сегментування;

– велика розмірність початкової множини даних;

– наявність ядра у сегментів, яке обумовлюється стратегією, що спрямована на захист свого сегменту та різними умовами для стабільної, гарантованої зайнятості й оплати праці в різних сегментах;

– наявність переміщень всередині сегментів і між ними.

Тому, для того, щоб не втрачати відмінності, які визначають сутність сегментів, необхідно працювати в багатовимірному просторі характеристик.

Визначити сегмент – це значить задати кількісні та якісні характеристики ядра цього сегменту. Ядро сегменту – це та частина сегменту, елементи якої характеризуються стійкими істотними ознаками. Нестійке положення суб'єктів на ринку праці визначають межі сегменту. Поняття меж сегменту за змістом співпадає з поняттям периферійного кільця, відомого з економічної теорії. Задачу сегментування ринку праці формулюємо як задачу визначення ядер і меж сегментів на заданій множині об'єктів при заданій необхідній мінімальній кількості сегментів та заданій множині значущих характеристик суб'єктів ринку.

Подамо задачу сегментування як задачу нечіткої кластеризації, специфіка якої в полягає у такому:

характеристики суб'єктів ринків визначаються за якісними та кількісними шкалами за класифікацією С. Стівенса (шкалах порядку, інтервалів, відношень та абсолютних шкалах);

для кластеризації використовується задана сукупність характеристик;

у задачі присутня апріорна інформація, а саме, відома мінімальна кількість сегментів;

описано істотні характеристики типових представників, яких може бути кілька у кожному сегменті.

Сегментами називаємо нечіткі підмножини, які мають певні спеціальні властивості. Ці сегменти, як правило, перетинаються через нечіткість характеристик елементів (суб'єктів ринків) і називаються нижче сегментами першого роду. Сегменти першого роду використовуємо в процедурі розбиття заданої множини на нечіткі підмножини, що не перетинаються, які називаємо сегментами другого роду. Наведемо формальні означення цих підмножин.

Нехай X – скінченна множина N елементів. На множині X з використанням коефіцієнту лінгвістичної кореляції [3] задаємо нечітке відношення схожості $\tilde{R}$ з функцією належності $f_{\tilde{R}}(x, y)$. Так визначене відношення є інваріантним до допустимих перетворень у шкалах за класифікацією С. Стівенса.

Нехай $\tilde{L}_y$ – нечіткий кластер з функцією належності $\mu_{\tilde{L}_y}(x) = f_{\tilde{R}}(x, y)$.

Позначимо

$\mathbf{C}_X^* = \left\{\tilde{C}_1^*, ..., \tilde{C}_{n_{\min}}^*\right\}$ – множину сегментів, мінімальна кількість $n_{\min}$ яких задана експертом, $n_{\min} \geq 2$;

$Ker\,\tilde{C}_i^* = \left\{z_{j_i}^i\right\}$ – елементи ядра сегмента $\tilde{C}_i^*$ (типові представники, задані експертом), $z_{j_i}^i \in X$, $j_i \in \{1, ..., k_i\}$, $k_i \geq 1$, $i \in \{1, ..., n_{\min}\}$.

Означення 1. Сегментуванням першого роду $\mathbf{C}_X = \left\{\tilde{C}_1, ..., \tilde{C}_n\right\}$ множини X назвемо сукупність $n_{\min} \leq n < N$ нечітких підмножин $\tilde{C}_i$, $i \in \{1, ..., n\}$, множини X, таких, що:

1. $\tilde{C}_i = \bigcup\limits_{y \in Ker\,\tilde{C}_i} \tilde{L}_y$,

$$\mu_{\tilde{C}_i}(x) = \max_{x \in X}\left\{\mu_{\tilde{L}_y}(x) \,\middle|\, y \in Ker\,\tilde{C}_i, (x, y) \in X \times X\right\};$$

2. $\forall x \in X \quad \exists \tilde{C}_i \in \mathbf{C}_X$ така, що $\mu_{\tilde{C}_i}(x) > 0$,

$\forall \tilde{C}_i \in \mathbf{C}_X \quad \exists x \in X$ такий, що $\mu_{\tilde{C}_i}(x) > 0$;

3. $\tilde{C}_X \supseteq \mathbf{C}_X^*$;

де $\mu_{\tilde{C}_i}(x)$ – функція належності кластера $\tilde{C}_i$, а також,

4. якщо $\tilde{C}_i \notin \mathbf{C}_X^*$, то $\forall x, y \in Ker\,\tilde{C}_i$ виконується умова $f_{\tilde{R}}(x, y) \geq \alpha_i,\ 0{,}5 < \alpha_i < 1.$

Значення α_i для сегмента $\tilde{C}_i \in \mathbf{C}_X^*$ визначається складом його ядра. Для сегментів $\tilde{C}_i \notin \mathbf{C}_X^*$ значення α_i визначається експертом і може змінюватися у процесі сегментування, якщо умова 2 не виконується.

Вочевидь, що $\bigcup_{i=1}^{n} Supp\ \tilde{C}_i = X.$

Елементи $\tilde{C}_i$, $i = 1,...,n$, сукупності нечітких підмножин $\mathbf{C}_X$ назвемо сегментами першого роду.

Означення 2. Сегментуванням другого роду $Segm\,X$, $Segm\,X = \{\tilde{S}_1,...,\tilde{S}_n\}$, множини X назвемо сукупність n нечітких підмножин $\tilde{S}_i$, таких, що задовольняють таким умовам:

$$Ker\ \tilde{S}_i = Ker\ \tilde{C}_i,\ i \in \{1,...,n\},$$

$$\bigcup_{i=1}^{n} Supp\ \tilde{S}_i = \bigcup_{i=1}^{n} Supp\ \tilde{C}_i,$$

$$Supp\ \tilde{S}_i \bigcap Supp\ \tilde{S}_j = \varnothing,\ i \neq j,\ \ i, j \in \{1,...,n\}.$$

Елементи $\tilde{S}_i$, $i = 1,...,n$, множини $Segm\,X$ назвемо сегментами другого роду.

Сегменти другого роду є нечіткими підмножинами, визначеними на множині X, функції належності яких визначають ступінь схожості елементів сегмента певному елементу ядра сегмента, а нечіткі ядра та межі сегментів другого роду не перетинаються.

Нехай $\mathbf{C}_X = \{\tilde{C}_1,...,\tilde{C}_n\}$ – деяке сегментування першого роду. Визначимо $\forall i \in \{1,...,n\}$ функцію належності сегмента $\tilde{S}_i$ так: $\forall x \in X$ $\mu_{\tilde{S}_i}(x) = \mu_{\tilde{C}_i}(x) \Leftrightarrow \mu_{\tilde{C}_i}(x) > \mu_{\tilde{C}_j}(x),\ \ j \in \{1,...,n\},\ i \neq j.$

Вочевидь, $\bigcup_{i=1}^{n} Supp\ \tilde{S}_i = X,$ однак $\bigcup_{i=1}^{n} \tilde{S}_i \subset X.$

Розбиття $Segm\,X$ визначаємо на основі нечітких комбінаторних конфігурацій (об'єктів) другого порядку першого типу [4], [5].

Комбінаторну задачу про розбиття на сегменти множини X формулюємо так: для заданого α, $0{,}5 < \alpha < 1$, та $n_{\min} \geq 2$ знайти розбиття $Segm\,X^* \in D \subseteq \tilde{M}$ таке, що

$$F\left(Segm\,X^*\right) = \underset{Segm\,X \in D \subseteq \tilde{M}}{ext} F\left(Segm\,X\right),$$

де $Segm\,X = \left\{\tilde{S}_1, ..., \tilde{S}_n\right\}$, D – допустима область, що визначається обмеженнями задачі, $\tilde{M}$ – множина всіх можливих сегментувань $Segm\,X$.

У залежності від специфіки певної задачі критерії сегментування обираємо так: $F_1\left(Segm\,X\right) = \left|\tilde{S}_X\right|$, де $\tilde{S}_X = \bigcup_{i=1}^{n} \tilde{S}_i$, тоді визначаємо $\tilde{S}_X^* = \max\limits_{Segm\,X \in D \subseteq \tilde{M}} \left|\tilde{S}_X\right|$; $F_2\left(Segm\,X\right) = \left|Segm\,X\right|$, тоді знаходимо $Segm\,X^* = \min\limits_{Segm\,X \in D \subseteq \tilde{M}} \left|Segm\,X\right|$.

Висновки. Сформульовано задачу сегментування ринків праці як нечітку задачу визначення ядер сегментів та меж сегментів на заданій множині об'єктів при заданій мінімальній кількості сегментів та визначеній множині значущих характеристик суб'єктів ринків. У випадку, коли використано критерій F_1 розв'язок задачі відповідає максимуму суми потужностей елементів розбиття (сегментів), тобто класів еквівалентності. Змістовна інтерпретація отриманого розв'язку задачі залежить від аналізу таких параметрів розв'язку як кількість сегментів, їхня потужність, а також наявності загальних характеристик елементів сегмента, частина яких може розглядатися як суттєві характеристики певного сегмента. При використанні критерію F_2 розв'язок задачі відповідає мінімуму кількості сегментів понад визначеним апріорі експертами значення $n_{\min}$. І в першому, і другому випадках суттєві властивості розбиття є такими.

1. Функції належності сегментів та міри схожості емпіричних елементів (суб'єктів ринку) інваріантні за наявності багатовимірних якісних та кількісних характеристик елементів, так як їхні значення обчислюються за допомогою коефіцієнта лінгвістичної кореляції. Це дозволяє проводити подальший аналіз властивостей отриманих класів,

зокрема, виявити суттєві характеристики з метою розробки заходів, спрямованих на покращення позицій суб'єктів ринку в кожному сегменті.

2. Отримані класи еквівалентності повністю відповідають неформальній постановці задачі нечіткого кластерного аналізу: елементи, які належать одному й тому ж сегменту більш схожі, ніж елементи з різних сегментів.

Для аналізу особливостей ринку праці має сенс одночасно використовувати обидва критерії, що може сприяти обґрунтованому визначенню сегментоутворюючих характеристик.

Література

1. Петрова І.Л. Ринок праці: процес сегментації. Київ: УДПУ. 1996. 176 с.

2. Стрембіцька Л.Л. Сегментація ринку праці: загрози економічній безпеці держави. Економіка і суспільство. 2017. 10. С. 547 – 551. https://economyandsociety.in.ua/journals/10_ukr/94.pdf

3. Hulianytskyi L., Riasna I. On Fuzzy Similarity Relations for Heterogeneous Fuzzy Sets. II International Scientific Symposium «Intelligent Solutions» IntSol-2021, September 28–30, 2021, Kyiv-Uzhhorod, Ukraine IntSol. P. 48 – 59. https://ceur-ws.org/Vol-3018/Paper_5.pdf

4. 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.

5. Hulianytskyi 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.

ПРО НЕОБХІДНІ ТА ДОСТАТНІ УМОВИ НАЯВНОСТІ АВТОМОРФІЗМУ У ПАРАЛЕЛЬНО-ПОСЛІДОВНИХ ГРАФАХ

Костянтин Караваєв, Валентина Турчина
Дніпровський національний університет імені Олеся Гончара

ON NECESSARY AND SUFFICIENT CONDITIONS FOR THE EXISTENCE OF AUTOMORPHISM IN SERIES–PARALLEL GRAPHS

Abstract. The paper investigates the prospects of applying the algorithm of non-isomorphic subgraph generation to series–parallel graphs in order to reduce the enumeration in the branch and bound method for the optimal sequencing problem. The necessary condition for the presence of automorphism which is true in the general case is given. It is proved that for this class of graphs, this condition is also sufficient.

Особливе місце серед дискретних оптимізаційних задач теорії розкладів займають задачі упорядкування вершин орграфів. Вони знайшли своє практичне застосування у багатьох сферах виробничої діяльності, оптимізації обчислень на ЕОМ та інших галузях, де на порядок виконання завдань накладаються деякі несуперечливі технологічні обмеження. Розв'язок задач з цього класу передбачає побудову такого розподілу завдань між виконавцями у кожний момент часу, який не суперечить заданим технологічним обмеженням та надає екстремальне значення деякому функціоналу якості.

Дослідження задачі показали, що у загальному випадку вона є NP-складною. Поліноміальні алгоритми були отримані лише для декількох спеціальних класів графів, що задають технологічні обмеження, та для задач з двома виконавцями. Тому особливу увагу науковці приділяють, як пошуку нових точних поліноміальних алгоритмів, так і пришвидшенню точних методів експоненційної складності, до яких належить метод гілок та меж.

Серед підходів до скорочення перебору методом гілок та меж можна виділити два основних: скорочення за рахунок уточнення оцінок функціоналу та за рахунок відсікання гілок, що відповідають ізоморфним підграфам.

Останній підхід досліджується у роботі [1]. У ній було запропоновано можливий алгоритм генерації неізоморфних підграфів. Дослідження цього алгоритму показали, що він не втрачає неізоморфні підграфи, проте серед результуючих графів можуть зустрітися й ізоморфні. Також у роботі наведені твердження, виконання яких дозволило б повністю виключити ізоморфні підграфи, та показано, що у загальному випадку вони є хибними.

Подальше дослідження з цієї теми було присвячено пошуку таких класів графів, для яких виконувалися б вищезгадані твердження. Одним з перспективних підкласів виявилися паралельно-послідовні графи [2].

Клас паралельно-послідовних графів визначається рекурсивно таким чином:

– Орграф $G = (\{i\}, \emptyset)$, який складається з єдиної вершини, належить цьому класу.

– Якщо графи $G_1 = (V_1, U_1)$ та $G_2 = (V_2, U_2)$, де $V_1 \cap V_2 = \emptyset$, належать цьому класу , тоді

$$G = G_1 S G_2 = (V_1 \cup V_2, U_1 \cup U_2 \cup T_1 \times S_2),$$

де T_1 – множина стоків графу G_1, S_2 – множина джерел графу G_2, також буде належати даному класу. В цьому випадку кажуть, що граф G є послідовною композицією графів G_1 та G_2.

– Якщо графи $G_1 = (V_1, U_1)$ та $G_2 = (V_2, U_2)$, де $V_1 \cap V_2 = \emptyset$, належать цьому класу, тоді граф

$$G = G_1 P G_2 = (V_1 \cup V_2, U_1 \cup U_2)$$

також буде належати даному класу і він є паралельною композицією графів G_1 та G_2.

– Тільки ті орграфи, що можуть бути отримані шляхом скінченої кількості застосувань попередніх двох побудов, належать до класу паралельно-послідовних графів.

Графи цього класу можна природним чином подати у вигляді бінарного дерева, в якому листи відповідають окремим вершинам графу, а внутрішні вузли визначають тип композиції, що була застосована при побудові до графів, які відповідають піддеревам з коренями у дочірніх вузлах. Таке бінарне дерево називають бінарним деревом декомпозиції. Якщо у бінарному дереві декомпозиції зв'язані групи однакових типів композиції стиснути в одну вершину з відповідним об'єднанням дочірніх вузлів, то отримане дерево називають канонічним деревом декомпозиції.

Наведемо терміни, що стосуються ізоморфізму графів з [1].

Два орієнтовні графи G та H є ізоморфними, якщо між множинами їх вершин існує взаємно однозначна відповідність, що зберігає суміжність вершин та орієнтацію дуг. Позначається $\mathrm{G} \cong H$ або $\mathrm{G} = H$.

Автоморфізмом орієнтовного графу G називається ізоморфізм G на себе, або автоморфізм є підстановкою множини вершин V, що зберігає суміжність та орієнтацію дуг. Дві вершини i та j, що переходять одна в одну під дією автоморфізму $f: V \to V$ $(i \in V, j \in V,\ f(i) = j)$ будемо називати взаємозамінними та позначати $i \sim j$.

У [1] були доведені необхідні умови взаємозамінності вершин для довільного графу.

Теорема 1. Якщо у графі G вершини i та j є взаємозамінними, то графи G_1 та G_2, утворені видаленням вершин i та j з G відповідно, будуть ізоморфними.

Для паралельно-послідовних графів ці умови виявилися також достатніми, що доводиться у наступних лемах та теоремі 2.

Лема 1. Графи G_1 та G_2 ізоморфні тоді і тільки тоді, коли графи $G_1 S G$ та $G_2 S G$ є ізоморфними.

Доведення. Необхідність. Нехай графи G_1 та G_2 ізоморфні, f – відповідний ізоморфізм, G – деякий граф.

Побудуємо дві матриці суміжності M_1 та M_2: в першій рядки і стовпці відповідають вершинам графу G_1 у порядку $1,2,\dots,n$ (n – кількість вершин у графах G_1 та G_2), в другій – графу G_2 у порядку $f(1), f(2), \dots, f(n)$. За означенням ізоморфізму ці матриці збігаються. Нехай M – матриця суміжності для графу G.

Розглянемо тепер блочні матриці $M_1' = \begin{pmatrix} M_1 & D_1 \\ 0 & M \end{pmatrix}$ та $M_2' = \begin{pmatrix} M_2 & D_2 \\ 0 & M \end{pmatrix}$, де D_1 та D_2 – матриці, в яких стоять одиниці на перетині рядків, що у графах G_1 та G_2 відповідають стокам, та стовпців, що у графі G відповідають джерелам. Матриці D_1 та D_2 збігаються, оскільки їх значення визначається лише наборами рядків і стовпців, а стоки ізоморфізм переводить у стоки, за властивістю, і матриця M є однаковою у M_1' та M_2', а отже і стовпці-джерела графу G. З цього випливає, що матриці співпадають за побудовою, а з іншого боку є матрицями суміжності графів G_1SG та G_2SG відповідно, тому вони є ізоморфними за означенням.

Достатність. Нехай графи G_1, G_2 та G – деякі графи, графи G_1SG та G_2SG ізоморфні, f – відповідний ізоморфізм.

Побудуємо матрицю суміжності графу G_1SG у формі $\begin{pmatrix} M_1 & D_1 \\ 0 & M \end{pmatrix}$, де блок M_1 відповідає вершинам графу G_1, блок M – вершинам графу G, D_1 – матриця, в якій стоять одиниці на перетині рядків, що у графі G_1 відповідають стокам, та стовпців, що у графі G відповідають джерелам. Це зможемо зробити, оскільки між вершинами графів G_1 та G дуги є лише між всіма стоками G_1 та джерелами G за означенням послідовної композиції. Тоді, якщо побудуємо матрицю суміжності для графу G_2SG з рядками та стовпцями у порядку $f(1), f(2), \dots, f(m)$ (m – кількість вершин у графі G_1SG), вона буде збігатися з матрицею суміжності для графу G_1SG, за означенням ізоморфізму.

Серед вершин $f(1), f(2), \dots, f(n)$ (n – кількість вершин у графі G_1) не може бути вершин графу G, оскільки ізоморфізм зберігає всі властивості графів та сума відстаней від будь-якої вершини графу G_1 до всіх стоків графу G строго більша за аналогічну суму для будь-якої вершини графу G, за властивістю послідовної композиції. Шляхом видалення з матриць суміжності рядків і стовпців $n+1, n+2, \dots, m$, що в обох графах відповідають вершинам графу G, отримаємо матриці M_1, що є матрицями суміжності графів G_1 та G_2, а отже вони є ізоморфними за означенням. ■

Наслідок 1. Графи G_1 та G_2 ізоморфні тоді і тільки тоді, коли графи GSG_1 та GSG_2 є ізоморфними.

Доведення. Аналогічно доведенню леми 1. ■

Лема 2. Графи G_1 та G_2 ізоморфні тоді і тільки тоді, коли графи G_1PG та G_2PG є ізоморфними.

Доведення. Необхідність. Нехай графи G_1 та G_2 ізоморфні, f – відповідний ізоморфізм, G – деякий граф.

Побудуємо дві матриці суміжності M_1 та M_2: в першій рядки і стовпці відповідають вершинам графу G_1 у порядку $1,2,\dots,n$ (n – кількість вершин у графах G_1 та G_2), в другій – графу G_2 у порядку $f(1), f(2), \dots, f(n)$. За означенням ізоморфізму ці матриці збігаються. Нехай M – матриця суміжності для графу G.

Розглянемо тепер блочні матриці $M_1' = \begin{pmatrix} M_1 & 0 \\ 0 & M \end{pmatrix}$ та $M_2' = \begin{pmatrix} M_2 & 0 \\ 0 & M \end{pmatrix}$. Вони співпадають за побудовою, а з іншого боку є матрицями суміжності графів G_1PG та G_2PG відповідно, з чого випливає, що вони ізоморфні за означенням.

Достатність. Нехай графи G_1, G_2 та G – деякі графи, графи G_1PG та G_2PG ізоморфні, f – відповідний ізоморфізм.

Побудуємо матрицю суміжності графу G_1PG у формі $\begin{pmatrix} M_1 & 0 \\ 0 & M \end{pmatrix}$, де блок M_1 відповідає вершинам графу G_1, блок M – вершинам графу G. Завжди зможемо це зробити, оскільки між вершинами графів G_1 та G немає дуг за означенням паралельної композиції. Тоді, якщо побудуємо матрицю суміжності для графу G_2PG з рядками та стовпцями у порядку $f(1), f(2), \dots, f(m)$ (m – кількість вершин у графі G_1PG), вона буде збігатися з матрицею суміжності для графу G_1PG, за означенням ізоморфізму.

Якщо серед вершин $f(1), f(2), \dots, f(n)$ (n – кількість вершин у графі G_1) немає вершин графу G, тоді шляхом видалення з матриць суміжності тих рядків і стовпців, що в обох графах відповідають вершинам G, отримаємо матриці M_1, які є матрицями суміжності графів G_1 та G_2, тому вони є ізоморфними за означенням.

Якщо серед вершин $f(1), f(2), \dots, f(n)$ знайдуться вершини графу G, тоді переставимо рядки і стовпці, які відповідають вершинам $f(1), f(2), \dots, f(n)$ та $1,2,\dots,n$, у матрицях суміжності так, щоб у матриці суміжності для графу G_2PG спочатку йшли вершини, що відповідають графу G_2, а потім G. Позначимо новий порядок вершин у першій матриці $i_1, i_2, \dots, i_k, i_{k+1}, \dots, i_n$, де k – кількість тих вершин з $f(1), f(2), \dots, f(n)$, що належать графу G_2. Помітимо, що при цьому зміниться лише матриця M_1, оскільки сусідні блоки нульові, позначимо змінену матрицю M_2.

Вона також матиме блочно-діагональний вигляд, оскільки між вершинами графів G_2 та G немає дуг за означенням паралельної композиції. Тоді вершини $i_{k+1}, \dots, i_n$ у графі G_1 та вершини $f(i_{k+1}), \dots, f(i_n)$ у графі G мають дуги лише між собою (всі інші значення у відповідних рядках та стовпцях дорівнюють нулю), а відповідні підграфи є ізоморфними (мають однакові матриці суміжності після видалення рядків і стовпців, що відповідають іншим вершинам).

Побудуємо відображення g графу G_1PG на себе, яке для вершини $i_{k+1}, \dots, i_n$ співпадає з f, для вершин $f(i_{k+1}), \dots, f(i_n)$ з f^{-1}, а для інших вершин є тотожнім. Таке відображення буде автоморфізмом графу G_1PG, що випливає з попереднього абзацу. Розглянемо композицію $f_1 = f \circ g$: вона є ізоморфізмом (за властивістю композиції ізоморфізму та автоморфізму), залишає вершини $f(i_{k+1}), \dots, f(i_n)$ на місці, оскільки $f\left(f^{-1}(f(i_l))\right) = f(i_l), l = \overline{k+1, n}$, відображає вершини $i_{k+1}, \dots, i_n$ у вершини $f(f(i_{k+1})), f(f(i_{k+2})), \dots, f(f(i_n))$ та не змінює відображення решти вершин (зокрема $i_1, i_2, \dots, i_k$ відображаються у ті ж вершини графу G_2).

Якщо тепер серед $f_1(1), f_1(2), \dots, f_1(n)$ немає вершин графу G, то твердження доведено з попередніх міркувань, інакше зможемо повторити усі наведені міркування та побудувати новий ізоморфізм. Тоді, оскільки граф є скінченним та кожен наступний ізоморфізм додає нові вершини, які він залишає на місці, та не впливає на відображення вершин, які залишали на місці попередні ізоморфізми, за побудовою, зрештою всі вершини G_1 будуть проектуватися на вершини G_2. ■

Наслідок 2. Графи G_1 та G_2 ізоморфні тоді і тільки тоді, коли графи, утворені деякою послідовністю паралельних та послідовних комбінацій цих графів з відповідними графами, є ізоморфними.

Доведення. Випливає з послідовного використання лем 1, 2 та наслідку 1. ■

Наслідок 3. Дві вершини є взаємозамінними у графі G тоді і тільки тоді, коли вони є взаємозамінними у графі, утвореному деякою послідовністю паралельних та послідовних комбінацій цього графу з деякими графами.

Доведення. Випливає безпосередньо з наслідку 2. ■

Теорема 2. Якщо для деяких відкритих вершин i та j паралельно-послідовного графу G, графи $G\backslash i$ та $G\backslash j$ є ізоморфними, то вершини i та j є взаємозамінними у графі G.

Доведення. Нехай T – канонічне дерево декомпозиції графу G. Розглянемо шляхи від вершин i та j до кореня T. Першою вершиною, що належить обом цим шляхам має бути вершина p, що відповідає

паралельній композиції. Справді, якщо б p відповідала послідовній композиції, то з цього випливало б, що існує орієнтовний шлях або з вершини i у вершину j, або у зворотному напрямку, за властивістю послідовної композиції, що в свою чергу суперечить тому, що обидві ці вершини є відкритими у G.

Розглянемо тепер граф G', що відповідає піддереву T з коренем p. Цей граф є паралельно-послідовним, оскільки є підграфом паралельно-послідовного графу G, та не є зв'язним, оскільки корінь p його дерева декомпозиції відповідає паралельній композиції. Вершини i та j належать графу G' за побудовою. З наслідку 2 та ізоморфності графів $G\backslash i$ й $G\backslash j$ випливає, що графи $G'\backslash i$ та $G'\backslash j$ також є ізоморфними.

Нехай тепер $\mathrm{G}'_1, \mathrm{G}'_2, \dots, \mathrm{G}'_k$ - система максимальних (за включенням) неорієнтовно зв'язних підграфів графу G' (набір послідовно-паралельних графів, що утворюють G' за допомогою паралельної композиції). Без втрати загальності можемо вважати, що вершина $i \in \mathrm{G}'_1$, а вершина $j \in \mathrm{G}'_2$, інакше p не була б першою загальною вершиною на шляхах до кореня T.

Розглянемо тепер графи $G'\backslash i$ та $G'\backslash j$. Граф $G'\backslash i$ утворено з підграфів $\mathrm{G}'_1\backslash i, \mathrm{G}'_2, \dots, \mathrm{G}'_k$, а граф $G'\backslash j$ – з підграфів $\mathrm{G}'_1, \mathrm{G}'_2\backslash j, \dots, \mathrm{G}'_k$. З ізоморфізму цих графів випливає, що знайдеться таке бієктивне відображення f, яке підграфу $G'\backslash i$ ставить у відповідність підграф $G'\backslash j$, ізоморфний йому. Відзначимо, що підграфи $\mathrm{G}'_1\backslash i$ та G'_1 і $\mathrm{G}'_2\backslash j$ та G'_2 не ізоморфні, оскільки містять різну кількість вершин, підграфи $\mathrm{G}'_q, q = 3, \dots, k$ містяться в обох графах.

Покажемо, що на основі f можна побудувати відображення, що проектує $\mathrm{G}'_1\backslash i$ на $\mathrm{G}'_2\backslash j$ та G'_1 на G'_2. Розглянемо $f(\mathrm{G}'_1) = \mathrm{G}'_x$, нехай $x \neq 2$ та $f(\mathrm{G}'_x) = \mathrm{G}'_y$, тоді з транзитивності відношення ізоморфізму випливає, що підграфи G'_1 та G'_y - ізоморфні, а отже зможемо побудувати нове відображення, що проектує G'_1 на G'_y, а G'_x на G'_x (залишає на місці). Повторюючи аналогічну побудову, прийдемо до відображення, що проектує G'_1 на G'_2, оскільки G'_1 завжди буде проектуватися на якийсь новий підграф (попередні будуть проектуватися самі на себе) та G'_2 - єдиний підграф, який міститься лише в одному з графі та може бути ізоморфним з G'_1. Аналогічні міркування можемо застосувати до підграфу $\mathrm{G}'_1\backslash i$, а отже шукане відображення існує.

Отже, підграфи G'_1 та G'_2 і $\mathrm{G}'_1\backslash i$ та $\mathrm{G}'_2\backslash j$ є ізоморфними. Нехай g ізоморфізм з G'_1 у G'_2 і $g(i) = l$. Тоді, якщо l співпадає з вершиною j, вершини i та j є взаємозамінними у G' за означенням (можна утворити наступний автоморфізм: g переводить G'_1 у G'_2 та i у j, g^{-1} – G'_2 у G'_1,

тотожне відображення – G'_q у $G'_q, q = 3, \dots, k$), а отже, за наслідком 3, й у графі G і твердження теореми є доведеним.

Якщо ж $l \neq j$, тоді маємо послідовно-паралельний граф G'_2, в якому є дві відкриті вершини l та j і графи $G'_2 \backslash j$ та $G'_2 \backslash l$ є ізоморфними, оскільки $g(i) = l$ та графи $G'_1 \backslash i$ та $G'_2 \backslash j$ є ізоморфними, а отже отримали початкове твердження для графу меншої розмірності. Тоді можемо всі міркування доведення застосувати до нього знову. Граф є скінченним, тому врешті-решт залишиться граф з однією вершиною, а отже будуть виконані умови з попереднього абзацу. ■

Доведена теорема підтверджує перспективність використання алгоритму генерації неізоморфних підграфів для паралельно-послідовних графів. Подальшого дослідження потребує перевірка твердження 2 з [1] та розробка методу для визначення взаємозамінних відкритих вершин у паралельно-послідовних графах.

Література

1. Караваєв К.Д., Турчина В.А. Аналіз впливу автоморфізму графу на схеми направленого перебору. – Питання прикладної математики і математичного моделювання. Дніпро, 2021 р. – С. 94-104.

2. Valdes J., Tarjan R.E., Lawler E.L. The recognition of series parallel digraphs. – SIAM J. Comput. – 1982. – Pp. 298-313.

КОМБІНОВАНА МОДЕЛЬ ЗАДАЧІ ПЛОСКОГО ПРЯМОКУТНОГО РОЗКРОЮ З РІЗНИМИ ТИПАМИ ЗАГОТОВОК

Ігор Козін, Олег Сардак, Заріфбой Сабіров
Запорізький національний університет Україна
Ургенчський державний університет, Узбекистан

COMBINED MODEL OF THE PROBLEM OF FLAT RECTANGULAR CUTTING WITH DIFFERENT TYPES OF BLANKS

Abstract. Розглядається один із варіантів відомої задачі прямокутного розкрою. Задача полягає у пошуку карт розкрою для певної кількості прямокутників на прямокутних заготовках різних типів і під час низки умов.

Більшість видів задач плоского прямокутного розкрою [1] є *NP*-важкими. Для пошуку оптимальних розв'язків таких задач виправдане застосування метаевристик.

Нехай заданий кінцевий набір із *n* невироджених прямокутників різних розмірів та набір прямокутних заготовок різних типів. Тип заготівлі визначається її розмірами. Передбачається, що кількість заготовок будь-якого типу необмежена.

Критерієм якості розкрою є загальна площа використаних заготовок. Крім того, в задачі можуть розглядатися додаткові критерії такі як мінімізація суми довжин конфігурації розкрою, компактність розкрою і деякі інші. Задача такого типу є *NP*-важкою. Однак широкий спектр застосувань робить виправданим пошуки евристичних та метаевристичних методів знаходження субоптимальних розв'язків подібних задач.

Для пошуку наближеного оптимального розв'язку задачі проведено декомпозицію задачі на дві основні частини:

1. задача плоского прямокутного розкрою з необмеженим безліччю заготовок різних типів;
2. задача оптимізації розкрою в середині кожної заготовки.

Враховуючи високу обчислювальну складність задачі розкрою, була використана гібридна ігрова та еволюційно-фрагментарна модель пошуку оптимуму [2].

Перший гравець пропонує послідовність типів заготовок, що описується цілим вектором $(x_1, x_2, ..., x_K)$. Тут K – максимально припустиме число заготовок, x_i - номер типу *i*-го за порядком заготовки. Другий гравець пропонує перестановку прямокутників, що укладаються - $(s_1, s_2, ..., s_n)$. Прямокутники послідовно розміщуються на заготовці щільно без перетинів за правилом Top-Left (або Left-Top). Завдання гравців - домогтися мінімізації основного та додаткових критеріїв в лексикографічного порядку. Крім того, за кожен прямокутник, який не вдалося розмістити на заготовках, нараховується одиничний штраф. Критерій мінімізації загальної суми штрафів має найвищий пріоритет.

Пошук оптимального рішення для двох гравців здійснюється за допомогою еволюційної моделі. Причому для першого гравця – це еволюційна модель на множині цілочисельних векторів у *K*-мірному цілочисельному паралелепіпеді, а для другого гравця – еволюційно-фрагментарна модель на перестановках з *n* елементів [2].

Література

1. Silvano Martello and Paolo Toth Knapsack problems. — Chichester, UK: John Wiley and Sons, 1990. — P. 221. — ISBN 0471924202
2. Козин И.В., С. И. Полюга Эволюционная модель упаковки многомерных объектов, Вісник Запорізького національного університету. Математичне моделювання і прикладна механіка. – 2010. – № 1. – С. 61-67.

ПРО РОЛЬ ПОПУЛЯРИЗАЦІЇ НАУКИ В СУЧАСНОМУ СУСПІЛЬСТВІ

Козирський В.*, Полевецька О.**, Шендеровський В.***
**Інститут теоретичної фізики НАНУ, **Інститут фізики НАНУ, ***Інститут фізики НАНУ*

«Ipsa scientia potestas est» (Знання є сила)
Френсіс Бекон
Країни, які хочуть мати майбутнє, повинні вкладати кошти у популяризацію наукових досягнень та їхніх авторів
Роальд Гофман, лавреат Нобелевої премії (1981)

Abstract: We present here a short essay on the role of science popularization in today society. The presentation emphasizes the triune essence of popularization and the main ways of achieving its goal. As an illustration, the paper contains some selected examples from the current Ukrainian reality.

Історія розвою й поневірянь цієї глибокої думки (Френсіса Бекона та інших філософів) не є тема подальших міркувань, але є потужне підґрунтя й поважна мотивація так бажання здобувати правдивих знань, як і прагнень поширення вже наявного наукового знання в приступній для широких зацікавлених кіл формі, себто є підвалина вельми важливого, проте часто доволі суперечливого процесу, званого популяризація науки.

Процес той є доволі складний і багатогранний. Найважливіші чинники тут є освіта, просвіта й громадсько-політичне виховання, що є невідкличні взаємопов'язані складники основних функційних обов'язків суспільних і державних інституцій. Часом популяризацію вважають лише засобом просвітницької функції науки. Певна логіка в цім є: як ви, науковці, такі розумні, що сказали один одному «А», то кажіть вже й приступне більшості публіки, що знає читати, таке «Б», яке б вона здатна

була б сприйняти, зрозуміти й погодитися з тим, що науковці не дарма споживають хліб за гроші суспільства. Певно, популяризацію не має підмінювати профанація, що є окрема доволі болюча тема.

Вертаючи до триєдиної місії популяризації, наголосімо, що освіта є перший щабель до засвоєння основ тих знань, що здатні зацікавити особистість предметово й забезпечити грамотне застосування отриманих знань у життєвій практиці, а ще, може й підштовхнути до амбітних прагнень стати до лав тих, що здобувають для людей нові знання. Тут уже стає на передній план просвіта, спільно творена освітянами й науковцями-популяризаторами, що розкривають перед охочими знати можливості розширити свої пізнання світу й зануритися у захопливий світ наукових пошуків і відкриттів. А що лінійне зростання кількости знаного призводить до щонайменше квадратичного зростання нових проблем і перспектив, то процес той вже сам по собі виховує високу суспільну мораль і свідомість.

Популяризація науки є одна з найскладніших проблем для науковця, бо хистка грань між простотою і профанацією є вельми тонка й розпливчаста. Перетворити цікаве для фахівця на цікаве для людини, що не має в тім належного знання , є вельми складна й творча проблема, бо, максимально спрощуючи виклад основних понять і явищ не можна припускатися підміни правдивої природи предмету. Популяризація науки як організована суспільно-виховна діяльність може мати так глобальний напрям доведення інформації до читача, глядача й слухача взагалі, як і націл на окремі групи споживачів за віковою чи фаховою ознакою. Стимуляція зацікавлености здобутками науки сприяє так піднесенню загального культурного рівня суспільства, як і забезпечення розвою науки допливом молодих ентузіястів. Роль талановитих педагогів і науковців-популяризаторів тут неоціненна.

Тепер є доволі поширені погляди, що популяризації в звичайному розумінні підлягають лише ділянки науки, де вже все усталено й добре зрозуміло. Підстави до того є, після знаменитої наукової революції кінця XIX – початку XX століття, зі з'явою релятивістики й квантової механіки тлумачення багатьох спостережуваних явищ втратили звичне механістичне трактування, що призвело до доволі бурхливих і непримиренних дискусій між вельми кваліфікованими провідними вченими. Проте для талановитої людини немає нічого неможливого. Й побудова приступних уяві непідготованого споживача привабливих образів є вже певним випробуваним рецептом тлумачення нових наукових об'єктів і явищ. Це доволі успішно показано на прикладі уявлень про кваркову структуру так званих елементарних частинок і їхніх квантових характеристик типу кольору, смаку, чарівности й т. ін.

Окремим, чи не найважливішим, часто найцікавішим і практично безпрограшним напрямом діяльности науковців-популяризаторів є ознайомлення земляків і громадськости поза межами країни з життям і діяльністю визначних науковців своєї країни, з їхніми життєвими перипетіями, науковими здобутками й громадянською позицією – наголошував видатний фізик, засновник фізичної школи з оптики в недалекому минулому Сергій Вавилов. У такім разі триєдина функція популяризації дістає свого концентрованого, багатократно помноженого й результативного вияву так у загальноосвітньому й просвітницькому, як і у виховному вимірі [1].

Національна академія наук України в останні роки приділяє цій ділянці науки увагу і розуміння. Приміром, уперше оргкомітет Міжнародного конгресу українців, який відбувся 25−28 червня 2018 року в Києві включив до програми на пленарне засідання доповідь «Українські вчені, які змінили світ» (А. Загородній, В. Козирський, В. Шендеровський); актуальним стало висвітлювати роль українських вчених-природознавців у становлення світової космонавтики, розвиток техніки, інформаційних технологій та інших напрямків природничої науки.

Важливість поширення знань про надбання і наукові здобутки працівників сфери науки і культури, а також повернення незаслужено замовчуваних та свідомо вилучених з енциклопедичних джерел імен вчених підтверджує зростання активності комунікаційної діяльності в НАН України. Значно збільшився обсяг, зросла якість представлення інформаційних повідомлень на офіційній сторінці НАН України, докладають зусиль для максимально широкого висвітлення результатів наукової діяльності через різні засоби масової інформації. На початку 2021 року створено Комісію НАН України з питань комунікацій із суспільством і популяризації наукової діяльності. З метою активізації зв'язків між науковою спільнотою і суспільством 26 січня 2022 року Президія НАН України прийняла рішення про заснування Премії НАН України «За популяризацію науки», яку будуть присуджувати щороку засобам масової інформації та їхнім окремим представникам, науковцям і організаторам самостійних проєктів за найкращий матеріал про здобутки вчених, діяльність наукових установ і НАН України загалом, а також за сприяння популяризації науки і піднесення престижу науковця в Україні (запроваджено три номінації).

Принагідно зазначмо, що вперше відбулося вручення премій «За популяризацію науки» (за підсумками конкурсу за 2022 року) на черговій сесії Загальних зборів НАН України 27 квітня 2023 року. У номінації «Найкращий науково-просвітницький проєкт року присуджено премію

за науково-популярний проєкт у соціальних мережах провідному науковому співробітникові відділу теоретичної фізики Інституту фізики НАН України, доктору фізико-математичних наук професору Василеві Шендеровському, який понад три десятиріччя плідно працює в ділянці історії науки, у справі донесення до українців і світової спільноти багатьох унікальних і малознаних фактів і подій з історії розвою світової науки, чільне місце в якій посідають українці й закордонні вчені українського походження. Переоцінити загальнонаціональну важливість цієї потужної шляхетної просвітницької й виховної праці професора неможливо [2].

Слід також наголосити, що наприкінці 2022 року МОН створив платформу «Вільна наука» для популяризації науки. Метою цього проєкту є створення нових можливостей для науковців (щодо популяризації результатів), журналістів (щодо уточнення результатів і перевірки трактувань наукових доробків українських вчених), вітчизняних і міжнародних інвесторів (для ознайомлення з практичними доробками вчених). У межах платформи відкрито реєстрацію для учених і журналістів. Читачі можуть знайомитися з контентом без попередньої реєстрації. Сподіємося, що відкриття платформи «Вільна наука» буде лише першим кроком активної участи держави в шляхетній справі поширення наукових знань.

Усі позитивні зміни в житті людства нерозривно пов’язані з розвитком науки і її застосовань у технологіях і побуті. Нині насправді реальна й невідкладна проблема полягає в тім, щоби повернути людям здоровий глузд в розумінні важливости наукових знань і природню повагу до носіїв тих знань − від вихователькі дитячого садка до провідних учених академій і університетів.

Життя людське є результат історичного оптимізму, годованого науковими знаннями й правильним їхнім використанням. Тяглість його ґрунтується на пильнім збереженні й шанобливій передачі тих знань від покоління до покоління. А це завше пов’язано з вихованням майбутніх дослідників, технологів і просто грамотних свідомих осіб, що можливо лише в разі добре розвиненої системи популяризації наукових знань.

Література:

1. Шендеровський В. Вчені України у світовій науці. Київ: ВД «Простір», 2019. 976 с.

2. Уперше вручено Премію НАН України «За популяризацію науки». Повідомлення НАН України. https://www.nas.gov.ua/UA/Messages/Pages/View.aspx?MessageID=10024

АВТОМОРФІЗМ. СИМЕТРИЧНА СТІЙКІСТЬ ВАГ ВЕРШИН ГРАФУ

Сергій Курапов, Максим Давидовський
Запорізький національний університет,
КЗ «Запорізький обласний інститут післядипломної педагогічної освіти» ЗОР

AUTOMORPHISM. SYMMETRIC STABILITY OF GRAPH VERTEX WEIGHTS

Abstract. The paper investigates the property of symmetric stability of graph vertex weights in the context of automorphism group construction. Construction of graph automorphism group is based on the concept of graph orbit. The set of isometric cycles and edge cuts of the graph G generates vector invariants of the graph. The weights of the vertices and edges of the graph, which are determined by the digital vector invariant of the edge graph IL(G) and the vector integral graph invariant IS(G)&IC(G), induce the symmetry of vertices' weights. The symmetry of graph vertex weights allows us to determine the orbits of the graph and construct the graph automorphism group Aut(G).

Автоморфізм графа є відображенням множини вершин на себе, що зберігає суміжність. Множина автоморфізмів утворює групу графа. Основою для побудови групи автоморфізмів є поняття орбіти. Побудова орбіти тісно пов'язана з кількісною оцінкою вершини або ребра графа, яка називається вагою. Для визначення ваги використовуються інваріанти графа, побудовані на спектрі реберних розрізів графа та спектрі реберних циклів [1]. Вага елементів графа дозволяє виявляти утворюючі цикли і формувати орбіти, що, своєю чергою, дозволяє побудувати групу автоморфізму графа. Розглянуто приклади обчислення певних суграфів графа.

Квадратну матрицю суміжностей реберного графа $A(L(G))$ [2,3] (або матрицю реберних розрізів $W(G)$) можна розглядати як лінійний нільпотентний оператор $W_S(G)$ розміром $m \times m$ (де m – число вершин у графі) [4]. При цьому множення матриць виконується у полі $GF(2)$ і нульове значення матриці реберних розрізів визначається показником ступеня оператора $W_s^p(G)$. Значення показника ступеня p визначає кількість рівнів у спектрі реберних розрізів графа. Кожен рядок (стовпець) є суграфом графа G. Кількість ребер в елементах матриці оператора для одного рівня (або для всіх рівнів) визначає вагу ребра. Для

спектра реберних розрізів сума ваг інцидентних ребер, що належать вершині, визначає її вагу. Таким чином, кожній вершині (ребру) графа можна поставити у відповідність число, що називається вагою вершини (ребра), і використовувати цей параметр для порівняння графів.

Означення 1. Інваріанти графа, що визначаються вагами вершин та ребер, називаються *векторними інваріантами графа*.

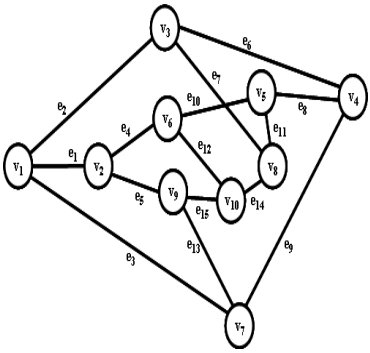


Рис.1 - *Граф G_1*

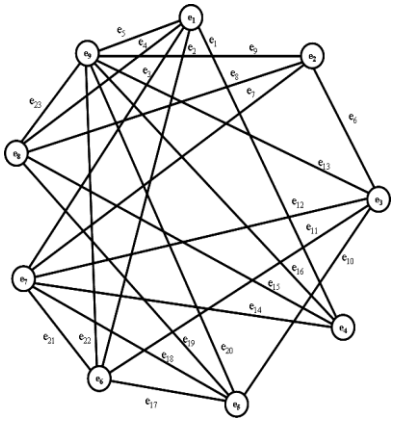


Рис.2 - *Граф G_2*

Кількість рівнів у спектрі реберних розрізів графа G_1 дорівнює 6 (рис. 1). Для кожного рівня складемо кортеж ваг:

$\xi(w(l_0))$ = <4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4>;

$\xi(w(l_1))$ = <10,10,8,10,8,8,10,10,10,10,8,8,10,10,10>;

…

$\xi(w(l_5))$ = <4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4>.

Сумарний кортеж ребер для шести рівнів визначає вектор ваг ребер: $F_e(G_1)$ = <32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32>. Для кожного рівня визначимо кортеж ваг вершини як арифметичну суму ваг інцидентних ребер:

$\varsigma(w(l_0))$ = <12,12,12,12,12,12,12,12,12,12>

$\varsigma(w(l_1))$ = <28,28,28,28,28,28,28,28,28,28>

…

$\varsigma(w(l_5))$ = <12,12,12,12,12,12,12,12,12,12>

Сумарний кортеж вершин для шести рівнів визначає вектор ваг вершин: $F_v(G_1)$ = <96,96,96,96,96,96,96,96,96,96>. За результатами обчислення векторів ваг вершин та ребер будується векторний інваріант

реберних розрізів графа для всього спектра реберних розрізів: $F_e(G_1)$ & $F_v(G_1)$ = (15×32) & (10×96). Або будується векторний інваріант реберних розрізів графа для кожного рівня окремо: $F_e(l_1)$ & $F_v(l_1)$ = (5×8),(10×10) & (10×28).

Спектр реберних розрізів графа може складатися з великої кількості рівнів і може перевищувати кількість вершин та ребер графа. Наприклад, спектр реберних розрізів графа G_2 має 30 рівнів (рис. 2). Отже, графи різняться між собою не тільки кількістю вершин і ребер, але й кількістю рівнів у спектрі реберних розрізів.

Поняття суграфа графа є дуже важливішим у теорії графів. Множина суграфів графа та операція кільцевого додавання суграфів породжують лінійний простір суграфів графа $£_G$. Підпростір розрізів графа $S(G)$ та підпростір циклів графа $C(G)$ належать простору суграфів $£_G$. Тому в описі графа слід враховувати розрізи, а також цикли графа. Отже, практичний опис графа має складатися з матриці суміжностей та матриці ізометричних циклів графа [5].

Означення 2. *Базовим реберним циклом* ребра e_i називається кільцева сума ізометричних циклів та обода графа, що мають у своєму складі ребро e_i.

Елементи базової множини реберних циклів графа можна подати у вигляді квадратної матриці розміром $m \times m$, що складається з нулів та одиниць. Дану матрицю, за аналогією з матрицею суміжностей реберного графа, можна представити як нільпотентний оператор і побудувати спектр реберних циклів графа. Як приклад розглянемо граф G_3 представлений на рис. 4(а) та його спектри реберних розрізів та циклів.

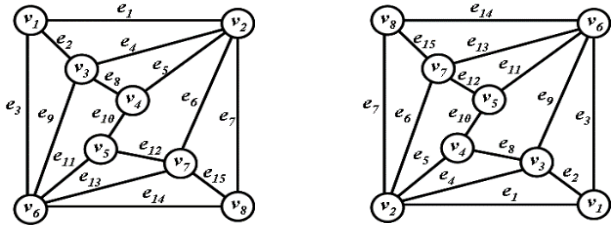


Рис.3 - *Автоморфізм графа* G_3

Визначимо кортеж ваг ребер для спектра реберних циклів:

$\xi(G_3)$ = <5,4,5,8,5,9,5,5,9,6,5,5,8,5,4>.

Кортеж ваг вершин: $\varsigma(G_3)$ $\varsigma(G_3)$ = <14,32,26,16,16,32,26,14>.

Рівні ваги вершин визначаються наступними парами вершин o_1 = {v_1,v_8}, o_2 = {v_2,v_6}, o_3 = {v_3,v_7}, o_4 = {v_4,v_5}.

Як видно з прикладу, рівність ваг для пар вершин $o_1 = \{v_1,v_8\}$, $o_2 = \{v_2,v_6\}$, $o_3 = \{v_3,v_7\}$, $o_4 = \{v_4,v_5\}$ зберігається для всіх відомих векторних інваріантів. Перестановка вершин у графі для множини вершин з рівною вагою породжує ізоморфний граф з іншою нумерацією вершин (рис. 4,б).

Вершинами реберного графа $L(G_3)$ є ребра графа G_3. Множину ізометричних циклів реберного графа $L(G_3)$ можна побудувати як об'єднання наступних трьох відповідних підмножин суграфів графа G_3: (1) підмножини ізометричних циклів відповідних множині центральних розрізів графа $G_3 - S_R^L$; (2) підмножини ізометричних циклів графа $L(G_3)$ відповідних множині ізометричних циклів графа G_3 – C_τ^L та (3) підмножини ізометричних циклів відповідних до підмножини дубль-циклів графа G_3 – C_d^L. Визначимо кортеж ваг ребер та вершин для множини ізометричних циклів реберного графа $L(G_3)$. Кортеж ваг ребер: <13,7,12,14,12,14,12,7,14,6,12,7,14,13,7>. Кортеж ваг вершин: <32,65,42,25,25,65,42,32>. Зауважимо, що рівні ваги вершин визначаються наступними парами вершин $o_1 = \{v_1,v_8\}$, $o_2 = \{v_2,v_6\}$, $o_3 = \{v_3,v_7\}$, $o_4 = \{v_4,v_5\}$.

З порівняння результатів величин кортежів ваг для різних векторних інваріантів графа G_3 випливає, що незважаючи на їхню відмінність існують стійкі підмножини вершин однакової ваги.

Для графа G_3 це наступні підмножини вершин – $o_1 = \{v_1,v_8\}$, $o_2 = \{v_2,v_6\}$, $o_3 = \{v_3,v_7\}$, $o_4 = \{v_4,v_5\}$.

Групу автоморфізму графа G_3 запишемо у вигляді:

$$Aut(G_3) = \begin{cases} p_0 = (1)(2)(3)(4)(5)(6)(7)(8) - \text{тотожний автоморфізм}, \\ p_1 = (18)(26)(37)(45). \end{cases}$$

Означення 3. Здатність породжувати вершини з рівними вагами у векторних інваріантах називається *стійкістю ваги вершин*.

Означення 4. Стійка підмножина вершин, що мають рівні ваги, відповідає *орбітам* графа та позначається символом «*o*».

З розглянутого також випливає, що для розрізнення графів достатньо мати перші два рівні $w_0(G)$ та $w_1(G)$, а для формування спектра реберних циклів достатньо одного базового рівня $\tau_0(G)$, що відповідає цифровому векторному інваріанту графа $IC(L(G))$.

Література

1. Курапов С. В., Давидовский М. В. Вычислительные методы определения инвариантов графа // International Journal of Open Information Technologies, 2021. Т. 9, №. 2. С. 1–8.

2. Свами М. Графы, сети и алгоритмы. Пер. с англ. / М. Свами, К. Тхуласираман – М.: Мир. 1984. – 455 с.

3. Харари Ф. Теория графов. Пер. с англ. Козырева В.П. / под ред. Гаврилова В.Г. / Ф. Харари – М.: Мир. – 1973. – 300 с.

4. Курапов С. В., Давидовський М. В. Алгоритмічні методи кінцевих дискретних структур. Ізоморфізм графів : монографія. Запоріжжя: Запорізький національний університет, 2021.

5. Kavitha T., Liebchen C., Mehlhorn K., Michail D., Rizzi R., Ueckerdt T., Zweig K. A. Cycle bases in graphs — characterization, algorithms, complexity, and applications // Comput. Sci. Rev. 2009. N.3. P. 199—243.

ПЛОЩИННІСТЬ ЛОКАЛЬНО ГОМЕОМОРФНИХ ГРАФІВ

Василь Лещенко
Інститут математики АН України

PLANENESS OF LOCALLY HOMEOMORPHIC GRAPHS

Abstract. It is proved that a graph locally homeomorphic to a given connected graph is a covering of the given graph. Based on the planarity criterion of graph covers, the planarity criterion of locally homeomorphic graphs is formulated.

Доводиться, що граф локально гомеоморфний заданому зв'язному графу, є накриттям заданого графа. На основі критерію площинності накриттів графів [9] формулюється критерій площинності локально гомеоморфних графів.

Необхідні топологічні означення наводяться в роботах [1-6]. Нехай $\tilde{X}$, X − локально лінійно зв'язні топологічні простори і X − зв'язний хаусдорфів простір. Накриттям топологічного простору X називається пара ($\tilde{X}$, f) простору $\tilde{X}$ і неперервного відображення f: $\tilde{X} \to X$, якщо для кожної точки $x \in X$ знайдеться такий лінійно зв'язний відкритий окіл U, що кожна компонента лінійної зв'язності множини $f^{-1}(U)$ відображенням f топологічно відображається на U. Множини $f^{-1}(x)$ для всіх $x \in X$ мають однакову потужність, яка називається числом листів накриття.

В загальному випадку число листів − кардинальне число, яке може бути скінченним або нескінченним. Відповідно, накриття буде n-листим, $n \in N$, або нескінченнолистим.

Далі розглядаються скінченні топологічні графи.

Топологічним графом називається хаусдорфів топологічний простір G, якщо:

1) на просторі G заданий дискретний підпростір G^0. Точки множини G^0 називаються вершинами графа G;

2) підпростір $G^1=G\backslash G^0$ − об'єднання відкритих підпросторів, які попарно не перетинаються і кожен з яких гомеоморфний відкритому інтервалу. Компоненти простору G^1 називаються ребрами графа G;

3) границя кожного ребра − підмножина множини G^0, що складається з однієї або двох точок. Ребро, границя якого складається з однієї точки, називається петлею.

Граф G називається скінченним, якщо він складається із скінченного числа ребер і вершин.

В роботі [8] сформульовані два типи екстремальних задач в теорії графів.

Нехай f – гомоморфізм графа G на граф F; $\succ$ – відношення впорядкування деякої множини графів.

Задача характеризації певного класу P частинних графів графа G за допомогою мінімальних (максимальних) елементів впорядкованої деяким відношенням $\succ$ множини fP образів графів із P при гомоморфізмі f називається (P, f)-мінімальною (максимальною) задачею першого типу відносно $\succ$.

Задача характеризації певного класу P частинних графів графа F за допомогою мінімальних (максимальних) елементів впорядкованої деяким відношенням $\succ$ множини f^{-1}P прообразів графів із P при гомоморфізмі f називається (P, f)-мінімальною (максимальною) задачею другого типу відносно $\succ$.

Відповідно до сформульованих (P, f)-екстремальних задач в роботах [10-13] розглядаються (P, f)-екстремальні графи.

Нехай P – множина деяких частинних графів графа G. Граф H, H∈P, називається (P, f)-мінімальним (максимальним) в G графом, якщо граф f(H) – мінімальний (максимальний) елемент впорядкованої відношенням $\succ$ множини fP, fP=$\{f(F)\ /F\in$P $\}$.

Граф H, H∈P, називається (P, f)-мінімальним (максимальним) в F графом, якщо граф f^{-1}(H) – мінімальний (максимальний) елемент впорядкованої відношенням $\succ$ множини f^{-1}P, f^{-1}P=$\{f^{-1}(F)\ /F\in$P $\}$.

γ_0-мінімальним в G графом, за гомоморфізму f: G $\to$ F, називається (P, f)-мінімальний в G граф при P=$\{F\ /F\subset$G, $f^{-1}f(F)=F$, $\gamma(F)>0\}$, а γ_0-мінімальним в F графом, за гомоморфізму f: G $\to$ F, називається (P, f)-мінімальний в F граф при P=$\{F\ /F\subset$F, $\gamma(F)>0\}$, де $\gamma(F)$ – рід графа F, а $\succ$ – таке відношення впорядкування множини P, що для двох негомеоморфних графів F, F' із P $F\succ F'$ тоді і тільки

тоді, коли граф F містить частинний граф, гомеоморфний графові F'. Для гомеоморфних графів F, F' із Р $F \succ F'$ тоді і тільки тоді, коли граф F – підрозбиття графа F'.

Для n-листих накриттів (G, f) довільного зв'язного графа F встановлені [8-13] множини $\mathscr{H}_n$ образів γ_0-мінімальних в G графів. Ясно, що степінь вершин графів із множини $\mathscr{H}_n$ більший двох, ці графи можуть мати петлі і кратні (паралельні) ребра, $n>1$.

Множини $\mathscr{H}_n$ – вичерпні набори попарно не ізоморфних „заборонених" підграфів графа F, які дають можливість сформулювати критерій площинності n-листих накриттів довільного заданого зв'язного графа: будь-яке n-листе накриття довільного заданого зв'язного графа – площинний граф тоді і тільки тоді, коли заданий граф не містить частинних графів гомеоморфних жодному графу із множини $\mathscr{H}_n$, $n>1$.

Поставимо задачу: знайти необхідні і достатні умови того, що будь-який граф, локально гомеоморфний заданому зв'язному графу – площинний граф.

В термінах „заборонених" підграфів розв'язання цієї задачі, дає критерій площинності накриттів заданого зв'язного графа:

1) якщо локальний гомеоморфізм − гомеоморфізм, то любий граф, гомеоморфний заданому зв'язному графу, буде площинним тоді і тільки тоді, коли заданий граф не містить частинних графів, гомеоморфних одному з графів Куратовського: K_5 чи $K_{3,3}$;

2) якщо локальний гомеоморфізм f: G $\to$ F − 2-кратний гомоморфізм, то (G, f) − 2-листе накриття зв'язного графа F. Згідно [8, 9] будь-який граф, 2-кратно локально гомеоморфний заданому зв'язному графу F, буде площинним графом тоді і тільки тоді, коли заданий граф F не містить частинних графів, гомеоморфних одному з графів H_i, $H_i \in \mathscr{H}_2$, $i \in N_1^{28}$. Графи H_i, $i \in N_1^{28}$, можна одержати із простого ланцюга відповідної довжини ототожненням вказаних в дужках вершин і ребер як в роботі [8].

3) якщо локальний гомеоморфізм − 3-кратний гомоморфізм, то (G, f) − 3-листе накриття зв'язного графа F. Згідно [9] будь-який граф, 3-кратно локально гомеоморфний заданому зв'язному графу F, буде площинним графом тоді і тільки тоді, коли заданий граф F не містить частинних графів, гомеоморфних одному з графів H_i, $H_i \in \mathscr{H}_3$, $i \in N_{29}^{33}$: $H_{29}=K_2^2+2v$, $H_{30}=K_2^2+v+r$, $H_{31}=K_2^2+2r$, $H_{32}=K_{1,3}+3v=K_1+3r$, $H_{33}=K_2^3$, де v − петля, r – росток (граф K_2 з однією петлею і одним відкритим кінцем);

4) якщо локальний гомеоморфізм − 4-кратний гомоморфізм, то (G, f) − 4-листе накриття зв'язного графа F. Згідно [9] будь-який граф,

4-кратно локально гомеоморфний заданому зв'язному графу F, буде площинним графом тоді і тільки тоді, коли заданий граф F не містить частинних графів, гомеоморфних одному з графів H_i, $H_i \in \mathcal{H}_4$, $i \in N_{33}^{34}$: граф H_{33} наведений вище, $H_{34} = K_2 + 2v$;

5) якщо локальний гомеоморфізм − n-кратний гомоморфізм, то (G, f) − n-листе накриття зв'язного графа F, $n \geq 5$. Згідно [9] будь-який граф, n-кратно локально гомеоморфний заданому зв'язному графу F, $n \geq 5$, буде площинним графом тоді і тільки тоді, коли заданий граф F не містить частинних графів, гомеоморфних одному з графів H_i, $H_i \in \mathcal{H}_5$, $i \in N_{33}^{35}$: графи H_{33} і H_{34} наведені вище, $H_{35} = K_1 + 2v$, де K_n − повний граф порядка n, G^n − граф, що одержується з графа G заміною кожного ребра n паралельними ребрами, $K_{m,n}$ − повний дводольний граф, n, $m \in N$.

Література

1. Энгелькинг Р. Общая топология. −М.: Мир, 1986. −752с.
2. Бурбаки Н. Общая топология. Основные структуры. −М.: Наука, 1968. −272с.
3. Куратовский К. Топология. Том 2. −М.: Мир, 1969. −624с.
4. Келли Дж. Л. Общая топология. −М.: Наука, 1981. −432с.
5. Архангельский А. В., Пономарев В. И. Основы общей топологии в задачах и упражнениях. −М.: Наука, 1974. −424с.
6. Масси У., Столингс Дж. Алгебраическая топология. −М.: Мир, 1977. −344с.
7. Харари Ф. Теория графов. −М.: Мир, 1971. −300с.
8. Хоменко Н. П., Лещенко В. Г. Два типа экстремальных задач в теории графов // Докл. АН УССР. Сер. А. −1987. −№6, −С. 13-15.
9. Хоменко Н. П., Лещенко В. Г. Плоскостность накрытий графов // АН СССР. Дискретная математика. Том 3 выпуск 2. −1991. −С. 104-113.
10. Хоменко Н. П., Лещенко В. Г. (P, f)-экстремальные графы // Экстремальные графы. – Киев, 1984. – С. 5-15 (Препринт / АН УССР. Ин-т мат.; № 84.44).
11. Лещенко В. Г. ξ-существенные цепи первого типа индекса один // Там же. – С. 28-56.
12. Хоменко Н. П., Лещенко В. Г. Структура (P, f)-экстремальных графов // Структура экстремальных графов. – Киев, 1984. – С. 5-37 (Препринт / АН УССР. Ин-т мат.; № 84.45).
13. Лещенко В. Г. ξ-существенные цепи индекса два // Там же. – С. 46-62.

АЛГОРИТМ ПОШУКУ ЕКСТРЕМАЛЬНИХ ШЛЯХІВ НА ГРАФІ

Олег Павленко, Анатолій Тимошенко, Оксана Тимошенко
Відкритий міжнародний університет розвитку людини «Україна»

Анотація. В роботі розглядається моделювання задачі комівояжера.

Пошук оптимального маршруту перебором всіх варіантів пропорційно (N-1)!

Звичайно, в мережі існує найвигідніший маршрут, але перегляд всіх маршрутів дуже трудомісткий для кількості вузлів більше 20. Пропонується використати адитивний ряд чисел з унікальними сумами елементів. Такий адитивний ряд, як 2^n швидко зростає при збільшенні кількості вузлів. Тут також, велике співвідношення між максимальним і мінімальним елементами ряду. Пропонується використати адитивний ряд з мінімальним значенням найбільшого елементу. Так для n=20 співвідношення між максимальним елементом адитивного ряду і ряду 2^n майже вдвічі менше. Крім того, мінімальний елемент адитивного ряду відрізняється від максимального в такому ж співвідношенні. Унікальна сума чисел дозволяє швидко шукати та порівнювати ряди, які мають такий самий набір вершин, але в іншій послідовності. Це дозволяє вибрати найкращий маршрут та заблокувати всі інші.

Результати моделювання показали перспективність такого алгоритму.

Ключові слова: адитивно-різний ряд чисел, теорія графів, екстремальні шляхи, задача комівояжера.

Вступ

Задачі визначення екстремальних шляхів мережі знайшли широке застосування [1,2]. Пошук найкоротшого шляху мережі реалізується простими алгоритмами, якщо відомі значення величин кожної гілки. Складніше задача визначення найкоротшого шляху в комп'ютерній мережі. Там час передачі інформації від одного вузла до іншого носять ймовірний характер [3]. Задача визначення найдовшого шляху в мережі виникла в системах мереженого планування і управління. У цьому випадку мережа складається з гілок (робіт з певним терміном їх виконання). Найбільший шлях в мережі є мінімальним терміном виконання всього комплексу робіт. Цей шлях називають критичним. Затримка будь-якої роботи цього шляху веде до затримки всього проекту. Проте окремі роботи (гілки) мають резерв часу і затримка їх в межах цього резерву не викликає затримки всього проекту. Задачу про критичний шлях можна звести до задачі визначення найкоротшого шляху. Для цього необхідно всі додатні значення гілок замінити на

від'ємні. Існують інші задачі, які можна звести до пошуку найкоротшого шляху так, наприклад, в роботах [4,5] задача розшифрування суцільного тексту зводиться до задачі визначення найкоротшого шляху. Задачі визначення пропускної здатності потоків, визначення екстремальних шляхів мають прості алгоритми і хороші методи розв'язку. Існують задачі, які прості за постановкою, але мають складні методи розв'язку. До таких задач відноситься задача налаштування багатопрофільного верстату з метою мінімізації часу його налаштування. Як правило, перехід від одної операції до іншої не співпадає зі зворотнім переходом. Тому матриця переходів носить несиметричний характер. Таку задачу представляє граф, в якому вузли є різні операції, а гілки фіксують терміни переходу з однієї операції на іншу. Для n вузлів з повними зв'язками кількість можливих шляхів складають (n-1)!. При n = 20 маємо величину $(n-1)! \approx 121*10^{15}$.

Ряди з унікальними сумами. Під рядами з унікальними сумами ми розуміємо множину чисел у якої кожна під множина має суму елементів яка відрізняється від будь-якою суму іншої підмножини. [1-5]

Сучасні комп'ютерні системи використовують паралелізм шляхом використання великої кількості процесорів. Але в цій задачі необхідна інтенсивна взаємодія між процесорами. Виникає необхідність комунікації процесорів в різні моменти часу. Іншими словами, необхідно створити послідовність n чисел, які мають унікальні суми для будь-якого набору елементів. Така властивість мають ряди типу r^n, де r, n- натуральні числа більше 1. Найменший такий ряд 2^n має унікальні суми будь-якого набору. Виникла задача створення ряду чисел, у якого максимальний елемент менший, ніж ряд 2^n. Вперше такий ряд був запропонований в роботі [6]. В роботі [6] розглянуті властивості такого ряду. На відміну від ряду 2^n цей ряд має значно менший максимальний елемент, але сума ряду більша.

Цей ряд будується для нового набору на базі попереднього додаванням першого елементу, який визначається за певними правилами. Таким чином, попередній ряд є оператором правої різниці. Пізніше з'ясувалося, що для n більше 6 , можна створити кращу послідовність, тобто з меншим значенням максимального елементу. Така послідовність також була створена по тому ж принципу, але вибір нового першого елементу для нової послідовності базується на іншому принципі [7]. Перевірка можливості вдосконалення поліпшеного алгоритму показала, що цей ряд оптимальний з точки зору мінімуму максимального елемента. Для невеликої кількості елементів n менше 9 перевірка зроблена суцільним перебором всіх можливих варіантів. Для n більше 8 використано більш складний алгоритм.

Пошук екстремальних шляхів на графі

Згідно з визначенням адитивна послідовність містить цілі числа, сума будь-яких елементів має унікальне значення, тобто не збігається з сумою інших елементів.

Запропонований у цій роботі спосіб припускає розщеплення вихідного пункту на початковий і кінцевий. Розщеплення вихідного пункту полягає в заміні його на два нових, в один із яких тільки надходять гілки, а з іншого тільки виходять. Таким чином, одержуємо початковий і кінцевий пункти. З початкового пункту посилають сигнал на всі вузли з інформацією про довжину шляху початкового пункту на всі інші вузли [2].

У кожному пункті після отриманої інформації готується набір даних, який містить:

1) Послідовності пройдених маршрутів з додаванням номера даного вузла.
2) Суми адитивних чисел пройденого шляху.
3) Суми довжини гілок пройденого шляху.

Інформація передається на всі вузли, які не містяться в послідовностях відповідних вершинах.

В таблицях 1 і 2 наведено схема роботи.

Дано ряд з 5 вершинами В0п В1 В2 В3 В4 В0к,

де В0п початкова (вихідні сигнали) і В0к кінцева вершина (вхідні сигнали).

Кожному процесору присвоюється адитивно-різне число:

В1 = 3; В2 = 5; В3 = 6; В4 = 7;

З В0п вершини подається сигнал на всі 5 процесорів. З кожного процесора подається інформація з набором отриманих шляхів на всі інші процесори, крім тих вершин, які вже обійшов. При цьому додається довжина пройденого шляху. Також записується адитивно-різне число, як сума чисел із вершин, які включені в маршрут, крім начальної В0п та кінцевої венршини. Приклад:

Маршрут 0—**2—3—1**—4

Сума адитивно-різного ряду 5 + 6 + 3 = 14

Якщо сума адитивно-різних чисел співпадає, тобто приходять послідовності які мають інший порядок проходження маршруту, але мають ті самі точки маршруту (0**231**4, 0**123**4), то залишається серед них маршрут мінімальною довжиною шляху

На кожному етапі додається одна вершина в кожну послідовність і відкидаються зайві варіанти (показано зеленим кольором в таблиці, а також кількість залишених маршрутів і коефіцієнт зменшення варіантів) в таблицях 1 і 2 останні стовпці.

Унікальна сума адитивно-різних чисел дозволяє швидко шукати та порівнювати ряди, які мають такий самий набір вершин але в іншій послідовності.

Так для 5 вершин береться адитивний ряд із 4 чисел: 3, 5, 6, 7.

Таблиця 1

k	Вершина 1	Сума. Адет. ряду	Длина пути	Вершина 2	Сума. Адет. ряду	Длина пути	Вершина 3	Сума. Адет. ряду	Длина пути	Вершина 4	Сума. Адет. ряду	Длина пути	Кількість строк пам’яти	Всього рядів	Кількість відібраних ряді	Кофіцієнт зменшення рядів
1	1		28	2		156	3		42	4		41	1	4		
2	**2** 1	5	293	**1** 2	3	165	**1** 3	3	48	**1** 4	3	65				
	3 1	6	62	**3** 2	6	189	**2** 3	5	303	**2** 4	5	328	3	12		
	4 1	7	78	**4** 2	7	213	**4** 3	7	67	**3** 4	6	68				
3	**3 2** 1	11	326	**1 3** 2	9	195	**2 1** 3	8	313	**2 1** 4	8	330				
	2 3 1	11	323	**1 4** 2	10	237	**1 2** 3	8	312	**1 2** 4	8	337				
	2 4 1	12	365	**3 1** 2	9	199	**1 4** 3	10	91	**1 3** 4	9	74				
	4 2 1	12	350	**4 1** 2	10	215	**2 4** 3	12	354	**3 1** 4	9	99				
	4 3 1	13	87	**4 3** 2	13	214	**4 1** 3	10	98	**3 2** 4	11	361				
	3 4 1	13	105	**3 4** 2	13	240	**4 2** 3	12	360	**2 3** 4	11	329	6	24	12	2
4	**4 3 2** 1	18	351	**1 4 3** 2	16	238	**2 1 4** 3	15	356	**2 3 1** 4	14	360				
	2 4 3 1	18	374	**1 3 4** 2	16	246	**4 2 1** 3	15	370	**1 3 2** 4	14	367				
	2 3 4 1	18	366	**4 3 1** 2	16	224	**4 1 2** 3	15	362	**1 2 3** 4	14	338	3	12	4	3
0	**4 3 2** 1	21	379													
0	**4 3 1** 2	21	380													
0	**2 1 4** 3	21	398													
0	**1 2 3** 4	21	379													

Таблиця 2

k	Вершина 1	Сума. Адет. ряду	Длина пути	Вершина 2	Сума. Адет. ряду	Длина пути	Вершина 3	Сума. Адет. ряду	Длина пути	Вершина 4	Сума. Адет. ряду	Длина пути	кількість строк пам’яти	всього рядів	кількість відібраних ряді	кофіцієнт зменшення рядів
1	1		28	2		156	3		42	4		41	1	4		
2	**2** 1	5	293	**1** 2	3	165	**1** 3	3	48	**1**4	3	65				
	3 1	6	62	**3** 2	6	189	**2** 3	5	303	**2**4	5	328	3	12		
	4 1	7	78	**4** 2	7	213	**4** 3	7	67	**3**4	6	68				
3	**2 3** 1	11	323	**1 3** 2	9	195	**1 2** 3	8	312	**2 1**4	8	330				
	4 2 1	12	350	**4 1** 2	10	215	**1 4** 3	10	91	**1 3**4	9	74				
	4 3 1	13	87	**4 3** 2	13	214	**2 4** 3	12	354	**2 3**4	11	329				
4	**4 3 2** 1	18	351	**4 3 1** 2	16	224	**2 1 4** 3	15	356	**1 2 3**4	14	338				
	4 3 2 1	21	379													
	1 2 3 4	21	379													

В Таблиці 1

k – кількість пройдених етапів. Залежить від кількості вузлів а саме n-1.

Вершини з 1 по 4 – містять в стовпчиках вказані можливі маршрути вузлів.

Суми адитивних чисел пройденого шляху.

Суми довжини гілок пройденого шляху.

Таблиці 2 в останніх стовпчиках

кількість маршрутів на вершині.

всього маршрутів = кількість маршрутів на вершині * кількість вершин.

кількість залишених маршрутів.

коефіцієнт зменшення = всього маршрутів / кількість залишених маршрутів.

В таблиці 2 представлені дані таблиці 1 після відкидання шляхів, які довші за знайдені. Це дозволяє суттєво скоротити число варіантів таблиця 9.

В таблиці 3 представлена матриця суміжності графа. Дано 5 вузлів. Зеленим кольором позначено розв'язок. Знайдено мінімальну довжину шляху 379, та сам шлях 0 1 2 3 4 0 в таблиці 3 позначений зеленим кольором.

Таблиця 3

	0	**1**	**2**	**3**	**4**
0	M	28	156	42	41
1	28	M	137	20	37
2	156	137	M	147	172
3	42	20	147	M	26
4	41	37	172	26	M

В таблиці 4 представлена матриця суміжності графа. Дано 20 вузлів. Це вхідні дані для програми. Зеленим кольором позначено розв'язок. Знайдено мінімальну довжину шляху 1032, та сам шлях вказаний в таблиця 5 для 20 вузлів.

Таблиця 4 - матриця суміжності графа

N	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**
0	M	194	112	98	104	75	171	130	51	60	101	112	61	165	145	143	101	85	52	79
1	194	M	198	219	256	249	73	184	161	138	266	236	209	41	176	57	138	275	221	125
2	112	198	M	208	215	183	138	28	74	127	209	42	172	158	239	172	61	176	163	144
3	98	219	208	M	42	61	227	223	137	101	60	209	37	206	79	162	183	102	54	101
4	104	256	215	42	M	37	256	235	152	128	20	208	51	239	119	199	201	73	52	134
5	75	249	183	61	37	M	239	204	126	114	26	172	42	227	138	193	176	41	28	125
6	171	73	138	227	256	239	M	118	125	128	261	179	206	37	209	86	84	256	211	126
7	130	184	28	223	235	204	118	M	86	134	230	69	189	143	246	165	54	201	183	149
8	51	161	74	137	152	126	125	86	M	54	151	92	104	126	164	118	51	134	101	73
9	60	138	127	101	128	114	128	134	54	M	134	145	78	114	112	84	86	138	85	20
10	101	266	209	60	20	26	261	230	151	134	M	198	57	247	138	209	201	54	51	142
11	112	236	42	209	208	172	179	69	92	145	198	M	172	196	252	204	98	156	158	165
12	61	209	172	37	51	42	206	189	104	78	57	172	M	190	102	153	152	79	20	85
13	165	41	158	206	239	227	37	143	126	114	247	196	190	M	178	52	98	249	199	106
14	145	176	239	79	119	138	209	246	164	112	138	252	102	178	M	126	195	178	122	98
15	143	57	172	162	199	193	86	165	118	84	209	204	153	52	126	M	112	221	165	69
16	101	138	61	183	201	176	84	54	51	86	201	98	152	98	195	112	M	183	151	98
17	85	275	176	102	73	41	256	201	134	138	54	156	79	249	178	221	183	M	60	152
18	52	221	163	54	52	28	211	183	101	85	51	158	20	199	122	165	151	60	M	97
19	79	125	144	101	134	125	126	149	73	20	142	165	85	106	98	69	98	152	97	M

Таблиця 5

0	8	11	2	7	16	6	13	1	15	19	9	14	3	4	10	17	5	18	12	0
	81	92	42	28	54	84	37	41	57	69	20	112	79	42	20	54	41	28	20	61

Таблиця 6. Параметри персонального комп'ютера.

ПК	Процесор	ОЗУ	ОС
ПК1	Inter® Core™ i-3 4130 CPU @ 3,40GHz	8 ГБ	Windows 7.
ПК2	Inter® Core™ 2 Duo CPU E8400 3,40GHz	3 ГБ	Windows XP.

В таблиці 7 представлені результати роботи програми на різних комп'ютерах з такими параметрами:

Вказано в секундах час виконання програми для N вузлів.

Також знайдена мінімальна довжина шляху, та його послідовність проходження ряду.

Таблиця 7

N	Час ПК1 в секунд	Час ПК2 в секунд	довжина шляху	ряд
4	0,0007	0,0002	627	0 2 1 3 0
5	0,0008	0,0003	675	0 2 1 3 4 0
6	0,0009	0,0003	683	0 2 1 3 4 5 0
7	0,0012	0,0005	698	0 2 6 1 3 4 5 0
8	0,0013	0,0008	704	0 2 7 6 1 3 4 5 0
9	0,0021	0,0015	717	0 5 4 3 1 6 7 2 8 0
10	0,0105	0,0038	737	0 5 4 3 9 1 6 7 2 8 0
11	0,0289	0,0101	746	0 5 10 4 3 9 1 6 7 2 8 0
12	0,0758	0,0297	806	0 5 10 4 3 9 1 6 7 2 11 8 0
13	0,1197	0,0951	820	0 5 10 4 3 12 9 1 6 7 2 11 8 0
14	0,2983	0,3306	825	0 5 10 4 3 12 9 1 13 6 7 2 11 8 0
15	1,074	1,2506	918	0 9 8 11 2 7 6 13 1 14 3 4 10 5 12 0
16	4,2615	4,9701	925	0 9 8 11 2 7 6 13 1 15 14 3 4 10 5 12 0
17	17,7792	20,5296	945	0 9 8 11 2 7 16 6 13 1 15 14 3 4 10 5 12 0
18	75,9745	86,8515	1014	0 9 8 11 2 7 16 6 13 1 15 14 3 12 4 10 5 17 0
19	327,1397	370,7264	1020	0 9 8 11 2 7 16 6 13 1 15 14 3 4 10 17 5 18 12 0
20	1412,084		1032	0 8 11 2 7 16 6 13 1 15 19 9 14 3 4 10 17 5 18 12 0

Таблиця 8. Представлено порівняння для алгоритму повного перебору чисел факторіал. (N-1)! варіантів і число варіантів для нашого алгоритму $2^{(N-2)}$. Де N кількість вузлів.

N	(N-1)!	$2^{(N-2)}$
5	24	8
20	121 645 100 408 832 000	262 144

Висновки. Розроблено алгоритм розв'язання задачі комівояжера, з використанням адитивно-різних чисел. Унікальна сума чисел дозволяє швидко шукати та порівнювати поточні маршрути, які мають такий самий набір вершин, але в іншій послідовності, та обирати коротший шлях маршруту.

Алгоритм зменшує кількість варіантів, відкидаючи зайві маршрути.

Швидкість роботи при 20 вершинах становить 23,5 хвилини.

Кількість варіантів $122*10^{15}$ (повний перебір), а після відкидання зайвих маршрутів залишається 262 144 варіантів.

Проект використання адитивних чисел для розв'язання задачі комівояжера перевірено на персональних комп'ютерах.

Результати експериментів в роботі приведені для різних графів з кількістю вузлів від 4 до 20. Час виконання задачі збільшується на кожному кроці приблизна в 4 рази.

Література

1. Oleh Pavlenko, Anatolii Tymoshenko, Oksana Tymoshenko, Andriy Luntovskyy, Yaroslav Pyrih & Igor Melnyk. Searching Extreme Paths Based on Travelling Salesman's Problem for Wireless Emerging Networking, in Springer LNEE 965, 2023, pp 284–304

2. Kostenko L.I., Tymoshenko A.G., Traynin E.Z.: Electronic modeling of operations research tasks [Text]. Academy of Sciences of the Ukrainian SSR, Institute of Cybernetics. — K.: Sciences. opinion, 1973. — 163 p.

3. Lande D., Berezyn B., Pavlenko O.: Development, evaluation and use of a word segmentation algorithm for monitoring systems of national Internet resources / Development, Evaluation and Usage of Word Segmentation Algorithm for National Internet Resources Monitoring Systems. Information technologies and security. Materials of the XVII International Scientific and Practical Conference ITB-2017. - K.: Engineering LLC, 2017. - C.22-31. CEUR Workshop Proceedings (ceur-ws.org). Vol-2067 urn:nbn:de:0074-2067-8. - Selected Papers of the XVII International Scientific and Practical Conference on Information Technologies and Security (ITS 2017) [http://ceur-ws.org/Vol-2067/paper3.pdf]

4. Lande D.V., Berezyn B.A., Pavlenko O.Yu.: Word segmentation algorithm based on the search for the shortest path in a graph. Registration, storage and processing data, 2017. - Vol. 19. - №. 4. - P. 3-15.
5. Aloyan G.S., Timoshenko A.G.: "Method of constructing a series of additively different numbers," Kibernetika, №. 6 (1969) [https://ur.booksc.eu/book/6416122/22173e]
6. Tymoshenko A.G., Luzan K.O., Rohacheva T. V.: Creation of a series of integers with unique sums of elements. Scientific notes of the National Academy of Sciences. - 2002 - Volume 20, Special issue: in two parts. Part 2. - P.516
7. Pavlenko O. Yu.: Determination of the optimal sequence of additively distinct numbers., EUROPEAN SCIENTIFIC DISCUSSIONS. Proceedings of X International Scientific and Practical Conference Rome, Italy, 2021, August 15-17. p. 75-79.

ДОСЛІДЖЕННЯ ЗАДАЧІ ПРО РЮКЗАК В УМОВАХ НЕВИЗНАЧЕНОСТІ

Олена Подковаліхіна, Владислав Смола
Національний університет «Запорізька політехніка»

THE RESEARCH OF THE KNAPSACK PROBLEM UNDER CONDITIONS OF UNCERTAINTY

Abstract. an approach to finding the optimal solution to a problem, whose input parameters are a system of independent random variables, was developed. An application with an interface, that finds optimal solutions and the total cost and writes the results to a file with a given number of experiments, was developed. The influence of statistical uncertainty on uniform and normal distribution laws for the optimal solution of the backpack problem was developed.

Задача пакування рюкзака застосовувалася в логістиці, криптографії, економіці та прикладній математиці, але сфера застосування цієї задачі в теперішній час розширилася: для оптимального керування на підприємствах машинобудівництва, для розподілу інвестицій в умовах статистичної невизначеності, для розподілу ресурсів сервера, для формування команди стартап-проекту, для визначення набору вітрових електроустановок ВЕС та ін. [1-5]. Розширення

застосування в різних сферах цієї задачі показує її незгасаючу актуальність.

Вміння знаходити оптимальні розв'язки, коли змінна підпорядковується певному закону розподілу, грає важливу роль для визначення порівняльної оцінки і тому актуальним є дослідження впливу невизначеності на оптимальний розв'язок задачі про рюкзак.

Загальна постановка задачі про рюкзак

Розглянемо задачу оптимізації упакування рюкзака з вхідними параметрами, що утворюють систему незалежних випадкових величин. В тривимірний простір (рюкзак, мішок, кузов автомобіля тощо) певного об'єму (ваги), який не має перевищувати P умовних одиниць (у.о.), упаковуються предмети n типів. Кожний предмет має свій об'єм (вагу) v_i і вартість (цінність) s_i. Вартість (цінність) або/та об'єм (вага) одного предмету кожного типу може варіюватися за відповідним законом розподілу. Потрібно упакувати предмети в тривимірний простір таким чином, щоб сумарна вартість (цінність) цих предметів була максимальною і не перевищувала об'єм (вагу) простору.

Математична постановка задачі про рюкзак

$$F = \sum_{i=1}^{n} s_i x_i \to max$$

$$\sum_{i=1}^{n} v_i x_i \le P,$$

де n – загальна кількість предметів;

x_i – кількість предметів типу i;

s_i – вартість (цінність) одного упакованого предмету типу i, s_i можуть варіюватися відповідно заданому закону розподілу або залишатися сталими;

v_i – вага (об'єм) одного упакованого предмету типу i, v_i можуть варіюватися відповідно заданому закону розподілу або залишатися сталими;

P – вага (об'єм) рюкзаку;

F – сумарна вартість (цінність) упакованих предметів без перевищення об'єму (ваги) рюкзаку.

Підхід знаходження розв'язків задачі

1. Обрати тип задачі та певний закон розподілу.

Перша задача – вартість може варіюватися відповідно заданому закону розподілу, а об'єм залишається сталим. Друга задача – об'єм може варіюватися відповідно заданому закону розподілу, а вартість залишається сталою. Третя задача – об'єм та вартість можуть варіюватися відповідно заданому закону розподілу.

2. Задати вхідні дані: об’єм (вага) рюкзаку; об’єм (вага) предмету кожного типу; вартість (цінність) предмету кожного типу; кількість експериментів; відхилення вартості або/та об’єму предмету кожного типу.

3. Розрахувати максимальну кількість упакованих предметів, що можуть поміститься в рюкзак для предметів кожного типу.

4. Знайти кількість всіх можливих планів, що можуть поміститься в рюкзак, враховуючи всі умови задачі.

5. Згенерувати випадкові числа вартості (цінності) та/або об’єму (ваги) за обраним законом розподілом, враховуючи відсоток відхилення для кожного типу предметів.

7. Визначити оптимальний план, який дає максимальну сумарну вартість для даних випадкових величин.

8. Зафіксувати знайдену сумарну вартість для певного оптимального плану в виділене місце або в створений масив.

9. Повторити кроки 6-8, поки не буде виконана обрана кількість експериментів.

10. Проаналізувати результати, застосовуючи обрані критерії оптимальності, для знаходження одного оптимального плану відповідної задачі.

Програмна реалізація для знаходження розв’язку задачі

Програмний код було розроблено на мові програмування C++ [6]. Потрібно було розробити застосунок, що знаходить всі оптимальні плани та їх сумарну вартість для заданої кількості експериментів для певної задачі про рюкзак. Для зручності користувача було додано інтерфейс (рис. 1). Для розробки інтерфейсу програми була обрана платформа WinForms, що взаємодіє з мовою програмування C++.

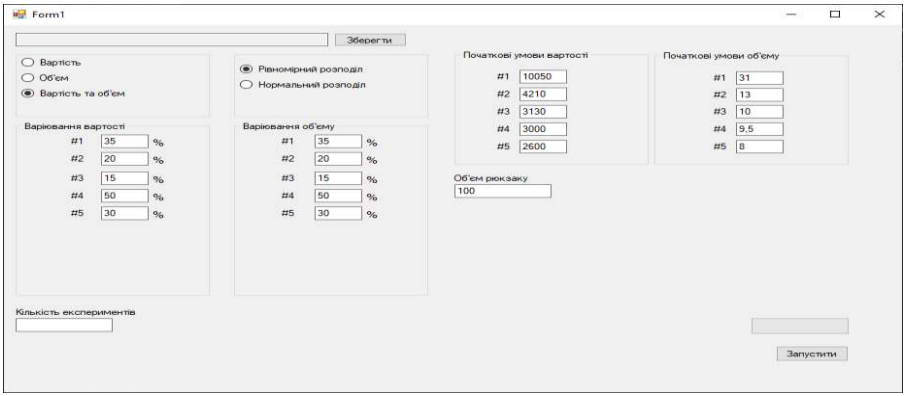


Рис.1 – Інтерфейс застосунку

Критерії оптимальності задачі

В результаті розв'язку задачі можна отримати певну кількість оптимальних планів, яка може досягати декількох сотень, а необхідно обрати лише один оптимальний план. Тому для отриманих розв'язків потрібно застосувати певні критерії оптимальності:

1 Критерій кількості з'явлень відповідного розв'язку у відсотках.

2 Критерій середньоарифметичної сумарної вартості відповідного розв'язку в у.о. (умовні одиниці).

3 Критерій ймовірності, що отримана сумарна вартість для даного розв'язку буде більше ніж k у.о.

Приклад

Розглянемо задачу пакування рюкзака, в якій вартість є випадковою величиною. Є 5 типів предметів, які потрібно упакувати в рюкзак. Об'єм рюкзаку дорівнює 100%. Предмет першого типу займає 31% об'єму рюкзаку, другого – 13%, третього – 10%, четвертого – 9,5% та п'ятого – 8%. Вартість одного предмету першого типу дорівнює 10050 у.о., другого – 4210 у.о., третього – 3130 у.о., четвертого – 3000 у.о. та п'ятого – 2600 у.о. Потрібно обов'язково упакувати не менше ніж по одному предмету першого, другого та третього типу.

Вартість для одного предмету кожного типу може варіюватися згідно певного закону розподілу відповідно наступних відсотків відхилення: один предмет першого типу – 35%, другого – 20%, третього – 15%, четвертого – 50% та п'ятого – 30%.

Необхідно упакувати рюкзак так, щоб сумарна вартість упакованих предметів була максимальною.

Отримаємо розв'язки задачі, в якій вартість є випадковою величиною з рівномірним законом розподілу. Запустимо розроблений застосунок та заповнимо всі блоки вхідними параметрами і проведемо дослідження з кількістю експериментів в 20000 за рівномірним законом розподілу.

В результаті було отримано 8 оптимальних планів. Скористаємося першим критерієм оптимальності. Розраховуємо нижню межу l_{min} (17,85%), що дозволить відсіяти ненадійні оптимальні плани. В таблиці 1 наведені результати після застосування першого критерію.

Таблиця 1

Результати за першим критерієм

Набір	Кількість
	20000
1 1 1 4 1	37,41%

1 2 1 0 4	20,05%
2 2 1 0 0	21,39%

Критерій середньоарифметичної сумарної вартості застосовується до результатів, що отримали після застосування критерія кількості з'явлень. Для початку розраховуємо значення q (34634,8). Всі сумарні вартості, що нижче даного значення відсіюємо. В таблиці 2 наведені результати після застосування другого критерію.

Так як за двома критеріями не залишився один оптимальний план, то застосовуємо третій критерій, а саме критерій ймовірності, що отримана сумарна вартість для даного розв'язку буде більше ніж k у.о. Розраховуємо k та отримаємо значення 35606,7. За цим значенням розраховуємо ймовірність для оптимальних розв'язків. В таблиці 3 наведені результати після застосування третього критерію.

Таблиця 2

Результати за другим критерієм

Набір	Середнє арифметичне
	20000
1 1 1 4 1	35209,7
2 2 1 0 0	36003,6

Таблиця 3

Результати за третім критерієм

Набір	Ймовірність сум. варт. >35606,7
	20000
1 1 1 4 1	45,5
2 2 1 0 0	61,1

Після застосування третього критерія можна остаточно рекомендувати оптимальний план (2-2-1-0-0), а саме: в рюкзак потрібно упакувати два предмети першого типу, два предмети другого типу та один предмет третього типу.

Отримаємо розв'язки задачі, в якій вартість є нормально розподіленою величиною. Запустимо розроблений застосунок, заповнимо всі блоки вхідними параметрами і проведемо дослідження з кількістю експериментів в 20000 за нормальним законом розподілу.

В результаті було отримано 6 оптимальних планів. Скористаємося першим критерієм для відсіювання планів. Розраховуємо нижню межу

l_{min} (17,85%) і відсіємо ненадійні оптимальні плани. В таблиці 4 наведені результати після застосування першого критерію.

Далі застосовуємо другий критерій. Розраховуємо значення q (32631,8) і всі сумарні вартості, що нижче даного значення, відсіємо (табл. 5).

Таблиця 4

Результати за першим критерієм

Набір	Кількість
	20000
1 1 1 4 1	35,7%
1 2 1 0 4	20,8%
1 3 3 0 0	20,1%

Після застосування другого критерія можна остаточно рекомендувати оптимальний план (1-1-1-4-1), а саме: упакувати в рюкзак один предмет першого типу, один предмет другого типу, один предмет третього типу, чотири предмета четвертого типу та один предмет п'ятого типу.

Таблиця 5

Результати за другим критерієм

Набір	Середнє арифметичне
	20000
1 1 1 4 1	33073,4

Висновки

Запропоновано підхід знаходження оптимального розв'язку задачі, вхідні параметри якої є системою незалежних випадкових величин. Розроблено застосунок з інтерфейсом, який знаходить оптимальні розв'язки та сумарну вартість і записує результати в файл з заданою кількістю експериментів. Інтерфейс застосунку дозволяє обирати задачу та закон розподілу, задавати початкові умови вартості та об'єму, максимальний об'єм рюкзака та відсоток відхилення вартості або/та об'єму для кожного предмету. Запропоновано критерії оптимальності для аналізу впливу невизначеності на оптимальний план задачі. Досліджено вплив статистичної невизначеності на прикладі рівномірного та нормального законів розподілу на оптимальний розв'язок задачі про рюкзак.

Література

1. Кравченко Є.І. Алгоритм гілок та меж для задачі формування команди та пошуку інвесторів для стартап-проектів [Текст] / Кравченко Є. І. // Журнал науковий огляд – 2018 р. – №6(49) – с. 70 – 76.

2. Ткачук В.М. Квантовий генетичний алгоритм в задачі 0-1 пакування мультиплікативного рюкзака [Текст] / Ткачук В.М. // Інформаційні технології та взаємодії – 2018 р. – с. 220 – 221.

3. Подковаліхіна О.О. Задача розподілу інвестицій в умовах статисти-чної невизначеності [Текст] / Подковаліхіна О.О., Логвіненко В.О., Бахрушин В.Є. // Системні технології. Регіональний міжвузівський збірник наукових праць. – Випуск 2 (121). – Дніпро, 2019. – С. 56 –63.

4. Качор П.М. Застосування задачі пакування рюкзака для оптимального розподілення ресурсів-сервера [Текст] / П. М. Качор ; наук. кер. Н. О. Бондаренко // Мехатронні системи і комп'ютерні технології Т.2: Ресурсозбереження та охорона навколишнього середовища. – Київ : КНУТД, 2019. – с. 67–68.

5. Безсонний В.Л. Розробка алгоритму оптимального управління ризиками небезпечних подій на машинобудівному підприємстві [Текст] / В. Л. Безсонний , к.т.н., доцент, доц. каф., Р. В. Пономаренко , д.т.н., с.н.с., заст. нач. каф., О. В. Третьяков , д.т.н., професор, наук. консультант, К. М. Карпець , к.геогр.н., доцент, провідн. н.с. // Проблеми надзвичайних ситуацій. 2021. № 1(33). – Харьків : ХНЕУ ім. С. Кузнеця, 2021. – с.58–71.

6. Галісєєв Г.В. Системне програмування [Текст] : навч. посіб. / Галісеєв Г.

КОМБІНАТОРНЕ УЗАГАЛЬНЕННЯ: МОДЕЛІ КРИТИЧНИХ ЗНАНЬ

Юрій Прокопчук

Інститут технічної механіки НАН України,

Придніпровська державна академія будівництва та архітектури

COMBINATORIAL GENERALIZATION: CRITICAL KNOWLEDGE MODELS

Abstract. Models of combinatorial generalization of the description of events, situations, scenes based on the paradigm of limiting generalizations are

considered. Critical sketches of images, situations are determined: they cannot be generalized without losing the unambiguity of the conclusion.

Передбачається [1, 2], що довільний стан складної системи або процес можна описати за допомогою безлічі елементарних тестів $\{\tau\}$. Кожен тест задається мережею начерків $Gs(\tau) = \{T \rightarrow_e T'\}_\tau$, де T, T' – домени. Для фіксації того, що в якості множини результатів тесту τ використовується домен T, будемо використовувати нотацію: τ/T. Конкретне значення тесту позначатимемо $\underline{\tau}/T$. Орграф доменів тесту задаються виразом: $G(\tau) = \{\underline{\tau}/T \rightarrow_e \underline{\tau}/T'\}_\tau$. Кожен домен є одночасно задачею розрізнення. Приклади мереж начерків тесту наведено на рис. 1 (вершини – це домени; найнижча вершина - це найточніший домен або базовий домен T_0). Подібне представлення дозволяє розглянути когнітивні моделі категоризації [1, 2, 3], інтелекту та імпліцитного формування глибоких або критичних знань для кожної задачі розрізнення («тонкий зріз» евристик).

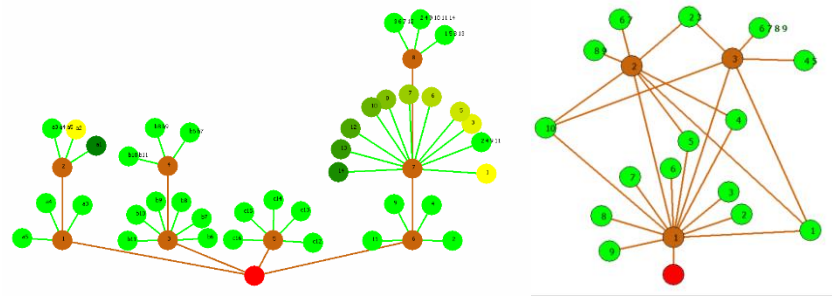


Рис.1 - *Приклади мереж начерків тесту*

Для будь-якого значення $\underline{\tau}/T$ у рамках орграфа доменів (або конфігуратора теста) $G(\tau)$ визначено конус деталізації - $G^{\downarrow}(\underline{\tau}/T)$ та конус узагальнення - $G^{\uparrow}(\underline{\tau}/T)$. Якщо через [•] позначити множину елементів-вершин орграфа, то

$[G^{\downarrow}(\underline{\tau}/T)] = \{\underline{\tau}/T' \mid \underline{\tau}/T' \rightarrow_G \underline{\tau}/T\}$; $\quad [G^{\uparrow}(\underline{\tau}/T)] = \{\underline{\tau}/T' \mid \underline{\tau}/T \rightarrow_G \underline{\tau}/T'\}$.

Аналогічно визначаються конуси $Gs^{\downarrow}(T_\tau)$ та $Gs^{\uparrow}(T_\tau)$.

Конуси узагальнення реалізують принцип структурної когерентності. На основі конусів деталізації реалізується перколяція активності та контрольована галюцинація (“controlled hallucination”). Локальний простір інтерпретації значення $\underline{\tau}/T$ (об'єднання конусів узагальнення та деталізації) позначимо $G^{\uparrow\downarrow}(\underline{\tau}/T)$ (відповідно $Gs^{\uparrow\downarrow}(T_\tau)$).

Приклад локального динамічного смислового простору довільного значення $\underline{\tau}$ показано на рис. 2 (ілюстрація гіперболічної метрики) [1].

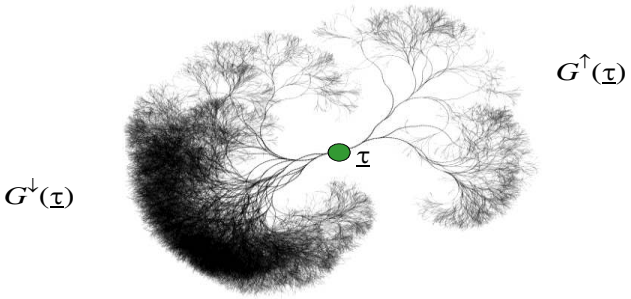


Рис. 2. - *Приклад смислового простору* $G^{\uparrow\downarrow}(\underline{\tau})$

Банком тестів називається онтологічно узгоджена множина орграфів $\{G(\tau)\}$. Різні комбінації доменів для всіх тестів визначають різні рівні узагальненості опису ситуацій дійсності (комбінаторне узагальнення). Отже, банк тестів забезпечує перехід від фізичної реальності до ментального простору та є системою координат довільного начерку образу, сцени, ситуації. Приклади ментальних систем координат наведено на рис. 3 [2].

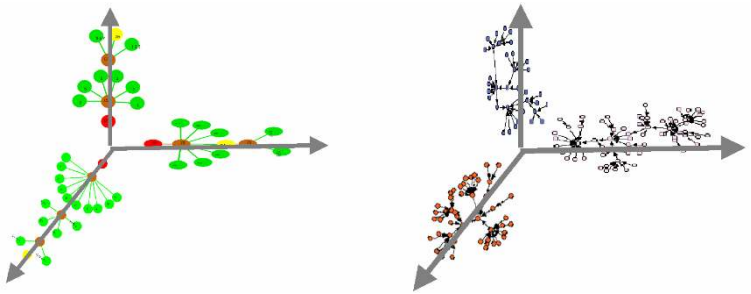


Рис. 3. - *Приклади ментальних систем координат*

Підкреслюю: всі начерки явища, події, процесу, сцени існують одночасно (простір інтерпретацій), однак енергія «уваги» розподілена нерівномірно - вона може переміщуватися між групами начерків (метафора «прожектору» або «свідомості»). Мають місце також процеси

конвергенції-дівергенції енергії на базі мережі мереж начерків. Мережа начерків тесту відображає концепцію «континууму задач розрізнення» (наслідок комбінаторного вибуху) та інкубацію знань [1, 2, 3]. Будь яка задача розрізнення вирішується за допомогою стохастичного процесу під назвою «Креативний перемішуючий шар» [1, 3].

Для розв'язання будь-якої Z-задачі розрізнення формується множина прецедентів з відомими наслідками $\Omega=\{\alpha(\{\underline{\tau}/T\}, \underline{z}/Z)\}$, де $Z = \{1, ..., N\}$ - множина висновків (діагнозів, прогнозів, управлінь); $\{\underline{\tau}/T\}$ - множина значень тестів; $\underline{z}/Z$ - висновки або результати. Фіксація Z-задачі створює випереджувальну систему категоріальних очікувань, яка готує увагу до збирання інформації про ознаки об'єктів (будь який домен – це Z-задача).

Без втрати спільності приймемо, що кожен тест входить до опису прецеденту (ситуації дійсності, сцени) один раз. Крім того, розглядатимемо описи прецедентів з повною інформацією (є значення всіх тестів з $\{G(\tau)\}$). Через $\Omega(\{\tau/T_0\}, Z)$ позначимо апріорні описи прецедентів. Множина всіх начерків бази прецедентів утворює орграф начерків $\Omega(Z)$, у якому $\Omega(\{\tau/T\}, Z)$ — окремий начерк. Можна також розглянути орграф начерків кожного прецеденту α, тоді опис $\alpha(\{\underline{\tau}/T\}, \underline{z}/Z)$ - це окремий малюнок. Усі начерки існують одночасно.

Зафіксуємо рівень узагальненості $\{\tau/T\}$. Начерк бази прецедентів $\Omega(\{\tau/T\},Z)$ називається конфліктним, якщо існують хоча б два прецеденти $\alpha(\{\underline{\tau}/T\}_\alpha, \underline{z}_\alpha)$ та $\beta(\{\underline{\tau}/T\}_\beta, \underline{z}_\beta)$, такі, що $\{\underline{\tau}/T\}_\alpha = \{\underline{\tau}/T\}_\beta$, але $\underline{z}_\alpha \neq \underline{z}_\beta$. Передбачається, що апріорний начерк $\Omega(\{\tau/T_0\}, Z)$ безконфліктний.

Будемо говорити, що домен T' узагальнює домен T у межах орграфа $G(\tau)$, якщо вони не збігаються, і існує шлях $T\rightarrow...\rightarrow T'$. Узагальнення позначимо нотацією: $T < T'$. Нестроге узагальнення (домени можуть збігатися) позначимо нотацією: $T \leq T'$. Зрозуміло, що тільки термінальні вершини орграфа $G(\tau)$ не можна узагальнити. Між будь-якими двома вершинами орграфа доменів або є відношення узагальнення, або ні.

Будемо казати, що опис $\{\tau/T'\}$ узагальнює опис $\{\tau/T\}$, якщо $\forall\tau\ \ T \leq T'$ і $\exists\tau\ \ T < T'$. Узагальнення описів позначатимемо нотацією: $\{\tau/T\}<\{\tau/T'\}$. Нестроге узагальнення (усі домени можуть збігатися) позначимо нотацією $\{\tau/T\}\leq\{\tau/T'\}$.

Якщо начерк бази прецедентів $\Omega(\{\tau/T\}, Z)$ безконфліктний, то безконфліктні також усі начерки $\Omega(\{\tau/T'\}, Z)$, такі, що $\{\tau/T'\}<\{\tau/T\}$. Якщо начерк бази прецедентів $\Omega(\{\tau/T\}, Z)$ конфліктний, то конфліктні також усі начерки $\Omega(\{\tau/T'\}, Z)$, такі, що $\{\tau/T\}<\{\tau/T'\}$.

Начерк бази прецедентів називається критичним - $\Omega(\{\underline{\tau}/T\}^*, Z)$, якщо він безконфліктний, але будь-який узагальнюючий начерк конфліктний.

Решта безконфліктних начерків називаються докритичними. Усі конфліктні начерки називаються закритичними. Закритичні начерки слугують основою для проведення аналогій, метафор і перенесення. Начерки бази прецедентів формують основу концепції "Бази даних, які насичені семантикою". Критичні начерки бази прецедентів відповідають концепції "тонкого зрізу". Саме вони є однією з цілей комбінаторного узагальнення.

Приклад комбінаторного узагальнення подій (концепція Суб'єктивного простору-часу-дій [1]). Подія може мати вигляд $Ev=<\underline{\tau}/T, \{\underline{p}/P\}, \underline{\varepsilon}/E>$, де $\underline{\tau}/T$ - факт; $\{\underline{p}/P\}$ – просторова та супутня інформація (контекст); $\underline{\varepsilon}/E$ – емоції. "Хмара смислів" або мережа начерків події: $Gs(Ev)=<G^{\uparrow\downarrow}(\underline{\tau}/T), \{G^{\uparrow\downarrow}(\underline{p}/P)\}, G^{\uparrow\downarrow}(\underline{\varepsilon}/E)>$. Мережа начерків відображає хвилю активації начерків (рух енергії). Ці хвилі активують, зокрема, евристики індукторного простору (різноманітні каузальні закономірності).

Приклад комбінаторного узагальнення повідомлень, тексту, наративу. Наратив може бути представлений у вигляді $Mes=<\underline{a}_1/A_1, \underline{a}_2/A_2,\ldots, \underline{a}_n/A_n>$, де a_j – тести. "Хмара смислів" або мережа начерків наративу у комуніканта: $Gs(Mes)=<G^{\uparrow\downarrow}(\underline{a}_1/A_1), G^{\uparrow\downarrow}(\underline{a}_2/A_2), \ldots, G^{\uparrow\downarrow}(\underline{a}_n/A_n)>$. Оскільки мережі начерків у кожного комуніканта свої, то і «хмари смислів» також особисті. Потрібно також врахувати особистий досвід [1, 3].

Запропонований підхід використовується для створенні інтелектуальних прикладень в різних галузях – медицині, техніці, транспорті, космічних технологіях [1 - 4].

Література

1. Прокопчук Ю.А. Интуиция: опыт формального исследования. Днепр: ГВУЗ «ПГАСА», 2022. – 724 с.

2. Прокопчук Ю.А. Набросок формальной теории творчества. Днепр : ГВУЗ «ПГАСА», 2017. - 452 с.

3. Прокопчук Ю.О. Когнітивні підходи до інтелекту: асимптотична раціональність та континуум задач розрізнення // Матеріали Міжнародної науково-технічної конференції «Інформаційні Технології в Металургії та Машинобудуванні» (м. Дніпро, 22 березня 2023 року). – Дніпро, Україна: УДУНТ, 2023. – С. 318 - 323. DOI: 10.34185/1991-7848.itmm.2023.01.086

4. Prokopchuk Y., Nosov P., Zinchenko S., Ben A. (2023). Bio-holonic approach to the creation of naturally reliable and safe transportation systems. Materials of the 15th Scientific and Practical Conference «Modern Information

and Innovative Technologies in Transport (MINTT-2023)» (May 24-25, 2023, Kherson). Kherson, Ukraine: XSMA. P. 37 – 41.

МЕТОД ОПРАЦЮВАННЯ ІНФОРМАЦІЇ В СИСТЕМАХ КООРДИНАТ МНОГОВИДІВ

В. В. Різник[1], М.Т. Соломко[2], С.С. Івасів[1], О. М. Дем'янів[1]

[1]Національний університет "Львівська політехніка", кафедра АСУ
[2]Національний університет водного господарства та природокористування, м. Рівне, кафедра обчислювальної техніки

METHOD OF PROCESSING INFORMATION IN COORDINATE SYSTEMS OF MANIFOLDS

Lviv Polytechnic National University
Correspondence: rvv@polynet.lviv.ua

Introduction.
Method of information processing in coordinate systems of manifolds formed on the set of combinational sums of basic vectors, using combinatorial configurations of «ideal ring bundles» type is considered, and on the basis of analysis of publications related to the study and use of spatial coordinate systems an approach to processing information in the basis of the coordinate system of a manifold is developed, where the basis is a subset of this coordinate system set. A one-to-one correspondence between the set of attribute categories of the input data and the set of coordinate sets of the spatial lattice of a manifold is established, the number of ring axes of the coordinate system of which determines the number of attribute categories, and the number of positions on these axes – the number of attributes of the corresponding category. On a concrete example, the peculiarities of the method of information processing in the basis of the torus coordinate system are investigated, which makes it possible to understand the essence of this method, due to the simultaneous encoding of vector data sets by two or more categories of attributes simultaneously. It was found that for processing information on lists of attribute categories it is advantageous to use optimal vector monolithic-group code. It is planned to continue studying the effectiveness of vector combinatorial configurations for processing information using various technologies and formats, which will open up new possibilities for using

methods of optimized coding and information processing in spatial coordinate systems of manifolds, developing optimized vector information technologies, improving data encryption and controlling dynamic objects.

Purpose of the article is to investigate the effectiveness of using combinatorial configurations in the basis of the coordinate system of manifolds for information processing.

Scientific novelty of the work lies in the development of a new approach to the optimal processing of information in the basis of the coordinate system of manifolds.

Practical significance – expansion of possibilities of application of methods of optimized coding and processing of multidimensional data arrays, due to simultaneous encoding of vector data sets into two or more categories of attributes simultaneously, improvement of encryption methods and management of dynamic objects. The possibilities of practical application of the method of optimal processing of information in the basis of the coordinate system of manifolds have not yet been fully disclosed.

Keywords: combinatorial optimization; coordinate system; optimal vector monolithic-group code; data indexing; data encryption.

Conclusions. The efficiency of using combinatorial configurations of the «ideal ring bindings» type for processing information in spatial coordinate systems of manifolds formed on the set of combinational sums of the base vectors of the mentioned configurations is investigated. A one-to-one correspondence between the set of attribute categories of input data and the set of coordinate sets of the spatial lattice of a manifold is established, the number of ring axes of the coordinate system of which determines the number of attribute categories, and the number of positions on these axes – attributes of the corresponding category. A concrete example of processing information on two categories of attributes in the basis of the torus coordinate system is given. The use of combinatorial configurations for data processing in optimal vector information technologies using optimized monolithic-group vector codes outlines a number of advantages aimed at improving methods for encrypting multidimensional signals when controlling dynamic objects, reducing the use of machine time and memory for information processing, due to the coding of information sets of attribute categories concurently. Information technologies based on vector combinatorial configurations are focused on all technical and practical aspects of the latest research and results of international academicians, scientists and practitioners.

Розглянуто метод опрацювання інформації в системах координат многовидів, утворених на множині комбінаційних сум базових векторів

комбінаторних конфігурацій типу «ідеальних кільцевих в'язанок», і на основі аналізу публікацій, пов'язаних з дослідженням та використанням просторових систем координат вироблено підхід до опрацювання інформації в базисі системи координат многовида, де базис – це підмножина множини координат цієї системи. Встановлено взаємно однозначну відповідність між множиною категорій атрибутів вхідних даних і множиною наборів координат просторової решітки многовида, число кільцевих осей системи координат якої визначає кількість категорій атрибутів, а число позицій на цих осях – кількості атрибутів відповідної категорії. На конкретному прикладі досліджено особливості методу опрацювання інформації в базисі системи координат тора, що дає змогу зрозуміти сутність зазначеного методу, завдяки одночасному кодуванню векторних наборів даних за двома і більше категоріями атрибутів одночасно. З'ясовано, що для опрацювання інформації за списками категорій атрибутів вигідно застосовувати оптимальний векторний монолітно-груповий код. Передбачено продовжити дослідження ефективності застосування векторних комбінаторних конфігурацій для опрацювання інформації за різними технологіями та форматами, що дасть змогу відкрити нові можливості застосування методів оптимізованого кодування й опрацювання інформації у просторових системах координат многовидів, розроблення оптимізованих векторних інформаційних технологій, вдосконалення шифрування даних і керування динамічними об'єктами.

Вступ. Сучасні інформаційні технології пов'язані з дослідженням ефективних методів опрацювання масивів даних в різних галузях науки і техніки. Суттєво зростає обсяг даних, які потрібно аналізувати, обробляти і використовувати. В роботі [1] використовується алгоритм швидкого перетворення Фур'є, класи представлення багатовимірних поліномів і двовимірних FIR- фільтрів перетворення частоти. За цими методами ґрунтується на маніпуляціях, таких як вибірка, перетворення Фур'є для стовпчиків і рядків, фільтрація тощо. Обчислювальна складність вищезгаданих методів зростає із числом вимірювань. У статті [2] запропоновано алгоритм декомпозиції та реконструювання багатовимірних сигналів на основі використання полі гармонічних вейвлет-сплайнів. Дослідження мінливості клімату на основі багатовимірного просторово-часового аналізу з розкладанням часових рядів у кожній точці координатної сітки з наступним об'єднанням просторової та часової еволюцій мінливості клімату в природних часових масштабах часово та прос досопрацювання характерна також в прогнозуванні погоди здійснено в [3]. Класифікацію багатовимірних часових рядів з двовимірною вибіркою запропоновано в [4], за якою для

виділення ознак обчислюються власні вектори коваріаційних матриць по рядках і стовпчиках, після чого вживається класифікатор «найближчого сусіда». В роботі [5] розглядається метод сумування сплайнів для відновлення багатовимірних зонних обмежень функцій із дискретних решітчастих вибірок з подальшим використанням полі гармонічних сплайнів. Підсумовуючи огляд, можна бачити, що зараз у науковому світі складається загальна тенденція стосовно опрацювання багатовимірних сигналів, яка ґрунтуються здебільшого на застосуванні алгоритмів швидкого перетворення Фур'є, класів раціонального представлення багатовимірних поліномів, двовимірних FIR-фільтрів перетворення частоти, вейвлет-перетворень та використанні полігармонічних сплайнів. У зв'язку з цим важливого значення набуває розроблення ефективних методів опрацювання інформації, застосовуючи векторні інформаційні технології, здатні подолати проблему швидкісного опрацювання великих масивів даних та надійного їх пересилання каналами зв'язку, використовуючи векторні інформаційні технології.

Мета роботи – дослідити ефективність застосування комбінаторних конфігурацій в базисі системи координат многовидів для опрацювання інформації.

Методика дослідження ґрунтується на використанні теорії багатовимірних комбінаторних конфігурацій типу ідеальних кільцевих в'язанок (ІКВ) для оптимізації опрацювання інформації в системах координат многовидів. При цьому множині кільцевих сум, утворених на *t*–вимірних векторах ІКВ, ставиться у відповідність множина індексованих *t* категорій атрибутів, яка вичерпує множину координат *t*–вимірної решітки многовида. Суть методу полягає у покритті координатною сіткою поверхні многовида визначених розмірів множиною кільцевих вектор-сум за допомогою мінімальної кількості базових векторів для формування в базисі цієї системи координат *t*–вимірного коду, векторні значення вагових розрядів якого збігаються з числовими значеннями базису вищезгаданої системи координат. При цьому кожному конкретному набору даних «категорія-атрибут» взаємно однозначно відповідає кодова комбінація *t*–наборів координат, яка складається з поруч розміщених за кільцевою схемою однойменних двійкових символів, утворюючи оптимальний кільцевий монолітно-груповий *t*–вимірний векторний код в базисі системи координат многовида.

Результати. Фрагменти результатів проведених експериментів представлені в таблицях 1 і 2. У таблиці 1 наведено приклад формування координатної решітки на двовимірній (*t*=2) поверхні тора 3×4 шляхом сумування базових векторів двовимірної ІКВ ((0,1),(1,0),(0,2),(2,2)) з

урахуванням числових значень модулів $m_1=(n-1)=3$, $m_2=n=4$. В результаті розрахунку формується двовимірна ($t=2$) координатна сітка на поверхні тора з розмірами 3×4, де координати кожного вузла сітки зустрічаються рівно один раз ($R=1$).

(0,0) (0,1) (0,2) (0,3)
(1,0) (1,1) (1,2) (1,3)
(2,0) (2,1) (2,2) (2,3)

У таблиці 2 наведено приклад кодування наборів даних за двома ($t=2$) категоріями атрибутів в базисі ((0,1),(1,0),(0,2),(2,2)) системи координат 3×4 тора, де кожній кодовій комбінації двовимірного монолітно-групового коду взаємно однозначно відповідає закодована інформація.

Наукова новизна роботи полягає у розробленні нового підходу до оптимального опрацювання інформації в базисі системи координат многовидів.

Практична значимість – розширення можливостей застосування методів оптимізованого кодування й опрацювання багатовимірних масивів даних, завдяки одночасному кодуванню векторних наборів даних за двома і більше категоріями атрибутів одночасно, удосконалення методів шифрування і керування динамічними об'єктами. Можливості практичного застосування методу оптимального опрацювання інформації в базисі системи координат многовидів ще повністю не розкриті.

Ключові слова: комбінаторна оптимізація; система координат; оптимальний векторний монолітно-груповий код; індексація даних; шифрування даних.

Актуальність вирішення проблеми полягає в розробленні ефективних методів опрацювання інформації.

Об'єкт дослідження – опрацювання інформації в просторовому полі координат многовида.

Метод вирішення завдання грунтується на використанні класичної теорії комбінаторних конфігурацій [6] і теорії багатовимірних ідеальних кільцевих в'язанок (ІКВ) [7].

Розглянемо математичну модель кодування t-вимірних сигналів в базисі t-вимірної системи координат многовида, утвореної на множині комбінаційних сум базових векторів комбінаторних конфігурацій типу ІКВ. Така система координат описується параметрами $n, R, S, m_1, m_2, \ldots, m_t$, де n- число базових t- вимірних векторів ІКВ, R- кількість різних способів утворення однакових кільцевих вектор-сум на множині t-вимірних базових векторів $((k_{11},k_{12},\ldots,k_{1t}), (k_{21},k_{22},\ldots,k_{2t}),\ldots, (k_{i1},k_{i2},\ldots, k_{it}),\ldots,(k_{n1},k_{n2},\ldots,k_{nt}))$, S – порядок обертової симетрії t- вимірної топологічно замкненої просторової решітки координат многовида, що

окриває його поверхню, де m_1, m_2,…, m_t – значення модулів, за якими окреслюються просторові розміри цієї решітки: $m_1 \times m_2 \times \ldots \times m_t$. Модель передбачає покриття множиною модульних сум, утворених n базовими t- векторами ІКВ, множини вузлових координат t - вимірної решітки з розмірами $m_1 \times m_2 \times \ldots \times m_t = n(n-1)/R$, де m_1, m_2,…,m_t – розміри кільцевих осей t - вимірної решітки, яка окриває поверхню (t+1)– вимірного многовида [7]. Множина кільцевих сум утворюється комбінаційним додаванням базових t–векторів за комплексним модулем (m_1,m_2,…,m_t), причому координати кожної вузлової точки t - вимірної решітки координат многовида можна отримати рівно R різними способами послідовного додавання базових векторів.

Інформаційні параметри оптимального n- розрядного t –вимірного векторного коду взаємопов'язані формулами [7]:

$n(n-1) \le S \le n(n-1)^2$, $\quad m_1 \cdot m_2 \cdot \ldots \cdot m_t = n(n-1)/R$

В основу методу закладено комбінаторну оптимізацію t –вимірних вагових розрядів зваженого n- позиційного коду, розрядам якого присвоєно такі значення цілочислових t–векторів, щоб множиною їх комбінаційних сум можна було покрити множину вузлових точок координат t –вимірної решітки многовида.

Наприклад, на кільцевій послідовності з чотирьох (n=4) базових двовимірних (t=2) векторів з ваговими розрядами ((0,1),(1,0),(0,2),(2,2)) можна сформувати $n(n-1)$=12 векторів, які заповнюють поле координат решітки тора 3×4 послідовним додаванням цих векторів з урахуванням числових значень модулів m_1= 3, m_2 =4, $m_1 \cdot m_2 = n(n-1)$=12, S =13 R= 1 (Табл.1).

Таблиця 1. Формування координатної решітки на поверхні тора 3×4

Вектори решітки тора 3×4	Базові вектори ІКВ			
	(0,1)	(1,0)	(0,2)	(2,2)
(0,0)	-	(1,0)	(0,2)	(2,2)
(0,1)	(0,1)	-	-	-
(0,2)	-	-	(0,2)	-
(0,3)	(0,1)	(1,0)	-	(2,2)
(1,0)	-	(1,0)	-	-
(1,1)	(0,1)	(1,0)	-	-
(1,2)	-	(1,0)	(0,2)	-
(1,3)	(0,1)	(1,0)	(0,2)	-
(2,0)	-	-	(0,2)	(2,2)
(2,1)	(0,1)	-	(0,2)	(2,2)
(2,2)	-	-	-	(2,2)
(2,3)	(0,1)	-	-	(2,2)

Приклад кодування даних за двома (t=2) категоріями атрибутів в базисі оптимальної системи координат тора у вигляді двовимірного (t=2) векторного коду з чотирма (n=4) ваговими розрядами ((0,1),(1,0),(0,2),(2,2)) ілюструє Таблиця 2.

У таблиці 2 вхідні дані передбачено опрацьовувати за двома (t=2) категоріями атрибутів у базисі ((0,1),(1,0),(0,2),(2,2)) системи координат тора, перша з яких характеризується трьома (m_1= 3), а друга – чотирма (m_2 =4) атрибутами. Система координат 3×4 зі спільною точкою відліку (0,0) набуває вигляду двовимірного (t=2) векторного коду, який має чотири (n=4) вагові розряди, а векторні значення цих розрядів збігаються з числовими значеннями базису системи координат, причому кожному конкретному набору даних «категорія-атрибут» взаємно однозначно відповідає кодова комбінація двовимірного векторного коду. В цьому прикладі метод дає змогу опрацьовувати масиви даних одночасно за двома категоріями атрибутів, один з яких може набувати трьох (m_1= 3), а другий – чотирьох (m_2 =4) значень.

Таблиця 2. Приклад кодування даних за двома (t=2) категоріями атрибутів в базисі ((0,1),(1,0),(0,2),(2,2)) системи координат 3×4 тора

п/п	Індекси категорій атрибутів		Ваги розрядів двовимірного (t=2) монолітно-групового коду			
	Категорія 1	Категорія 2	(0,1)	(1,3)	(0,2)	(2,3)
1	0	0	0	1	1	1
2	0	1	1	0	0	0
3	0	2	0	0	1	0
4	0	3	1	1	0	1
5	1	0	1	1	0	0
6	1	1	0	1	1	0
7	1	2	1	1	1	0
8	1	3	0	1	0	0
9	2	0	1	0	0	1
10	2	1	0	0	1	1
11	2	2	1	0	1	1
12	2	3	0	0	0	1

В загальному випадку число категорій і атрибутів може змінюватися під час опрацювання інформації залежно від розмірності та розмірів координатної сітки обраного базису. Масиви даних можуть описуватися

t –наборами атрибутів довільного змісту на будь-якому рівні індексації й теоретично нескінченно великим числом категорій атрибутів з використанням оптимізованих *t* –вимірних монолітно-групових кодів.

Висновки. Досліджено ефективність застосування комбінаторних конфігурацій типу «ідеальних кільцевих в'язанок» для опрацювання інформації у просторових системах координат многовидів, утворених на множині комбінаційних сум базових векторів згаданих конфігурацій. Встановлено взаємно однозначну відповідність між множиною категорій атрибутів вхідних даних і множиною наборів координат просторової решітки многовида, число кільцевих осей системи координат якої визначає кількість категорій атрибутів, а число позицій на цих осях – атрибутів відповідної категорії. Наведено конкретний приклад опрацювання інформації за двома категоріями атрибутів в базисі системи координат тора. Застосування дво- і багатовимірних комбінаторних конфігурацій для опрацювання даних з використанням оптимізованих монолітно-групових векторних кодів окреслює ряд переваг, націлених на вдосконалення методів шифрування багатовимірних сигналів під час керування динамічними об'єктами, зменшення використання машинного часу і пам'яті для опрацювання інформації, завдяки одночасному кодуванню інформації за наборами категорій атрибутів. Метод опрацювання інформації в системах координат многовидів орієнтовані на всі технічні та практичні аспекти новітніх досліджень і результатів міжнародних академіків, вчених і практиків.

Література

1. Bose, N.R. (2017). Multidimensional Sampling. *Applied multidimensional Systems Theory*, pp.57-80. DOI: 10.1007/978-3-319-46825-9_3
2. Bacchelli B., Bozzini M., Rabut C. and Varas M.(2005) Decomposition and reconstruction of multidimensional signals using polyharmonic pre-wavelets. *Applied and Computational Harmonic Analysis*.- 18.- 2005.- pp. 282-299.DOI: 10.1016/j.acha.2004.11.007
3. Wu, Z., Feng, J., Qiao, F. and Tan Z.-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. DOI:10.1098/rsta.2015.0197
4. Weng X. and Shen J. (2008). Classification of multivariate time series using two-dimensional singular value decomposition. *Knowledge-Based Systems*, Vol.21, Iss. 7, pp.535-539. DOI: 10.1016/j.knosys.2008.03.014

5. Madych W.R.(1999) Spline type summability for multivariate sampling. *Analysis of Divergence*, pp. 477-512. DOI: 10.1007/978-1-4612-2236-1_27
6. Hall Jr. (1998). Combinatorial Theory / Jr. Hall. – 2 edition: Wiley-Interscience, 1998. – 464 p. DOI: 10.1002 / 9781118032862
7. Різник В.В. (2019) Комбінаторна оптимізація багатовимірних систем. Моделі багатовимірних систем інтелекту. Львів: Видавництво Львівської політехніки, 2019. – 168 с. ISBN 978-966-941-285-0

МЕТОД ПОБУДОВИ ГРАЦІОЗНИХ УНІЦИКЛІЧНИХ ГРАФІВ

Марина Семенюта
Льотна академія НАУ

METHOD FOR CONSTRUCTING GRACEFUL UNICYCLIC GRAPHS

Abstract. We have proposed a way to construct a graceful unicyclic graph from a graceful tree.

Хоча пройшло вже більше 50 років з того моменту як з'явилися перші розмітки графів, відомо не так багато загальних методів побудови будь-якої з розміток. Більшість статей концентрують свої зусилля на створенні сімейств графів, що володіють розміткою певного типу. Детальну інформацію з цього питання можна знайти в электронном журнале «A dynamic survey of graph labeling» під редакцією Д. Галлиана.

Розглядатимемо скінчені неорієнтовані графи, що не містять петель та кратних ребер. Розмітку f графа $G=(V,E)$ розміру q називають граціозною, якщо вона є ін'єктивною функцією з областю визначення V, а її множина значень є підмножиною множини $\{0,1,2,...,q\}$. Крім цього f породжує реберну розмітку, що є бієктивним відображенням множини ребер E в множину $\{0,1,2,...,q\}$. Граф G – граціозний, якщо має граціозну розмітку.

Лема 1. [2] Нехай T – дерево порядку p з граціозною розміткою $f:V(T)\to\{0,1,2,...,p-1)$. Тоді функція

$f^*: V(N) \to \{0,1,2,\ldots,p-1\}$, що визначається формулою $f^*(v) = p - f(p) \quad \forall v \in V(T)$, також задає граціозну розмітку T.

М. Трущинський у 1984 році висинув гіпотезу, що всі уніциклічні графи, крім C_n, $n \equiv 1,2 (\mathrm{mod}4)$ є граціозними [3], яка до теперешнього часу залишається відкритою. В даній роботі розглянемо спосіб побудови граціозних розміток уніциклічних графів. Він базуються на методах генерації графів з графів менших порядків. Сформулюємо та розвяжемо наступну задачу.

Задача 1. Нехай задано граціозне дерево T порядку p ($p \geq 3$). Чи для кожного такого дерева T, існує граф $G = T + e$, що допускає граціозну розмітку.

Будь-який уніциклічний граф G є надграфом дерева T, тобто $G = T + e$, де ребро e з'єднує несуміжні вершини T. Відомо, що багато класів дерев допускають граціозну розмітку. Цей факт та лема 1 використано для розвязання задачі 1.

Алгоритм розв'язування задачі 1

Нехай задано дерево T порядку p ($p \geq 3$) з граціозною розміткою $\varphi: V(T) \to \{0, 1, 2, \ldots, p-1\}$ і $\varphi(v) = 0, \varphi(u) = p - 1$, де $u, v \in V(T)$. Для дерева φ є бієкцією.

Крок 1. Якщо вершина v є висячою в T, то перейти до кроку 2, інакше до кроку 3.

Крок 2. Задати функцію $\varphi': V(T) \to \{0, 1, 2, \ldots, p-1\}$, що визначається формулою: $\varphi'(x) = p - \varphi(x) \ \forall x \in V(T)$. З'єднати v з вершиною w, де $w \in V(T)$ і $\varphi(w) = 1$. Граф $G = T + e$ – уніциклічний, де $e = (vw)$. Задати функцію $f: V(G) \to \{0, 1, 2, \ldots, p\}$:

$$f(x) = \varphi'(x) \quad \forall x \in V(G) - \{v\},$$
$$f(v) = p \text{ для } u \in V(G).$$

Перейти до кроку 6.

Крок 3. Вершина v не є висячою в T. Якщо вершина u – висяча в T, то перейти до кроку 4, інакше до кроку 5.

Крок 4. З'єднати u з вершиною w, де $w \in V(T)$ і $\varphi(w) = 1$. Граф $G = T + e$ – уніциклічний, де $e = (uw)$. Задати функцію $f: V(G) \to \{0, 1, 2, \ldots, p\}$:

$$f(x) = \varphi(x) \quad \forall x \in V(G) - \{u\},$$
$$f(u) = p \text{ для } u \in V(G).$$

Перейти до кроку 6.

Крок 5. Вершина u не є висячою в T. Обрати вершину w, де $w \in V(T), (vw) \in E(T)$ і $\varphi(w) = i$. З'єднати u з вершиною z, де $z \in V(T)$, $(uz) \notin E(T)$ і $\varphi(z) = p - 1 - i$. Граф $G = T + e$, де $e = uz$ – уніциклічний.

Задати функцію $f: V(G) \to \{0, 1, 2, \dots, p\}$:

$$f(x) = \varphi(x) \quad \forall x \in V(G) - \{w\},$$
$$f(w) = p \text{ для } w \in V(G).$$

Перейти до кроку 6.

Крок 6. Граф $G = T + e$ – граціозний.

Обґрунтування алгоритму

Крок 2. Вершина v є висячою в граціозному дереві T, тому суміжна тільки з вершиною u, що має максимальну мітку, тому $vw \notin E(T)$.

Згідно леми 1, розмітка $\varphi'(x) = p - \varphi(x)$ є граціозною для дерева T і породжує реберну розмітку $\varphi'^*(xy) = |\varphi'(x) - \varphi'(y)| = \varphi^*(xy)$ $\forall (xy) \in E(T)$, де $\varphi^*(xy)$ – реберна розмітка, індукована φ.

Вершинна розмітка f є ін'єкцією з $V(G)$ в множину $\{0, 1, 2, \dots, p\}$ і породжує реберну розмітку

$$f^*(xy) = \varphi^*(xy) \quad \forall (xy) \in E(G) - \{(uv), (vw)\},$$
$$f^*(uv) = p, \quad f^*(vw) = p - 1.$$

Звідси маємо: $f^*\big(E(G)\big) = \{0, 1, 2, \dots, p\}$. Отже, f^* – бієктивна функція і f – граціозна розмітка уніциклічного графа $G = T + e$, де $e = (vw)$.

Крок 4. Вершинна розмітка f графа $G = T + e$, де $e = (uw)$ є інєктивною функцією з $V(G)$ в множину $\{0, 1, 2, \dots, p\}$. По аналогії з обґрунтуванням кроку 2, нескладно показати, що породжена нею реберна розмітка – бієктивна функція з $E(G)$ в множину чисел $\{0, 1, 2, \dots, p\}$, тому f – граціозна розмітка G.

Крок 5. Міркування проводяться за аналогією з тими, що наведені для підтвердження правильності результатів кроків 3 і 5.

Таким чином, доведено наступну теорему.

Теорема 1. Для кожного граціозного дерево T порядку p ($p \geq 3$), існує граф $G = T + e$, що допускає граціозну розмітку.

Литература

1. Gallian J. A. A dynamic survey of graph labeling. *The Electronic Journal of Combinatorics*. 2022. DS6: Dec 2. 502 p.

2. E. Robeva, An Extensive Survey of Graceful Trees. *Undergraduate Honors Thesis, Stanford University*, 2011. 39 p.

3. Truszczynski M. Graceful unicyclic graphs. *Demonstatio Mathematica*. – 1984. – V17. – P. 377-387.

ПРО РОЗВ'ЯЗАННЯ ЗАДАЧІ МЕДИЧНОЇ ЛОГІСТИКИ ЯК ДВОЕТАПНОЇ ЗАДАЧІ РОЗМІЩЕННЯ З ОБМЕЖЕННЯМИ НА МАКСИМАЛЬНУ КІЛЬКІСТЬ ОБ'ЄКТІВ

Олексій Сергєєв, Світлана Ус
Національний технічний університет «Дніпровська політехніка»

ON THE SOLUTION OF THE PROBLEM OF MEDICAL LOGISTICS AS A TWO-STAGE PROBLEM OF PLACEMENT WITH RESTRICTIONS ON THE MAXIMUM NUMBER OF OBJECTS

Abstract. We propose an approach to enhance logistics processes within the healthcare industry. Our method employs a genetic algorithm with priority coding to resolve a two-stage location problem. A distinctive feature of our model is its imposing limitation on the maximum number of regional and subregional centers that can be established.

To optimize the algorithm, we have extensively modified the chromosome estimation process. These alterations improve the solution-finding mechanism and population construction process. Using real-world data, we have meticulously tested our modifications on a model task, enhancing our research's credibility and applicability. The outcomes offer valuable insights into the potential of our proposed method.

This paper contributes to discovering efficient and effective solutions to logistical challenges in healthcare. We emphasize the potential of advanced computational techniques, such as genetic algorithms, to tackle complex real-world problems.

Вступ. Оптимізація логістичного процесу набуває все більшого значення в різних галузях, в тому числі і в медицині. Ефективне управління медичною логістикою відіграє вирішальну роль в охороні здоров'я та добробуті людей. Це особливо важливо під час глобальних викликів, коли швидке та ефективне розповсюдження ліків має вирішальне значення. Крім того, налагоджена логістика має важливе значення для надання гуманітарної допомоги, коли своєчасна доставка медичних матеріалів і ресурсів може врятувати життя. Для вдосконалення процесів медичної логістики застосовуються передові технології та алгоритми. Одним із підходів, що набуває популярності, є генетичні алгоритми для розв'язання двоетапних задач розміщення, які дозволяють краще розподіляти та позиціонувати медичні заклади, розподільчі центри та ресурси [1]. Використовуючи реальні дані та застосовуючи їх до цих алгоритмів, медичні організації та логістичні провайдери можуть приймати обґрунтовані рішення, які підвищують

ефективність, оперативність та економічність медичних логістичних операцій.

Метою роботи є дослідження застосування двоетапних задач розміщення в контексті медичної логістики.

Математична модель. Нехай відомі:

– I – кількість регіональних центрів;

– J – кількість субрегіональних центрів;

– K – кількість медичних складів;

– N – максимальна кількість регіональних центрів, що можуть бути розміщені;

– L – максимальна кількість субрегіональних центрів, що можуть бути розміщені.

Метою задачі є мінімізація загальних транспортних витрат на доставку медикаментів з регіональних центрів до складів та вибір оптимальних локацій для субрегіональних центрів. Транспортні витрати включають:

– c_{ij}^1 – вартість транспортування одиниці медичних товарів з регіонального центру i до субрегіонального центру j ($i = \overline{1, I}$, $j = \overline{1, J}$);

– c_{jk}^2 – вартість транспортування одиниці медичних товарів з субрегіонального центру j до складу k ($j = \overline{1, J}$, $k = \overline{1, K}$).

Додатково задані:

– M_j – вартість розміщення субрегіонального центру ($j = \overline{1, J}$);

– P_i – вартість розміщення регіонального центру ($i = \overline{1, I}$);

– r_i – запас медичних засобів призначення у регіональному центрі $i = \overline{1, I}$;

– s_j – запас медичних засобів у субрегіональному центрі $j = \overline{1, J}$;

– d_k – ємність складу ($k = \overline{1, K}$);

Невідомі:

– x_{ij} – кількість одиниць медичних засобів, перевезених з регіонального центру i до субрегіонального центру j;

– y_{jk} – кількість одиниць медичних засобів, перевезених з субрегіонального центру j до складу k;

– v_i – булева змінна, де $v_i = 1$, якщо розміщено регіональний центр *i* або $v_i = 0$ в іншому випадку;

– z_j – булева змінна, де $z_j = 1$, якщо розміщено субрегіональний центр *j* або $z_j = 0$ в іншому випадку;

Маємо задачу: мінімізувати

$$\sum_{i=1}^{I}\sum_{j=1}^{J} c_{ij}^{1} x_{ij} + \sum_{j=1}^{I}\sum_{k=1}^{K} c_{jk}^{2} y_{jk} + \sum_{i=1}^{I} P_i v_i + \sum_{j=1}^{J} M_j z_j , \qquad (1)$$

за обмежень:

$$\sum_{j=1}^{J} y_{jk} \le s_j z_j , \ \forall k = \overline{1, N} , \qquad (2)$$

$$\sum_{j=1}^{J} x_{ij} \le r_i , \ \forall i = \overline{1, I} , \qquad (3)$$

$$\sum_{i=1}^{I}\sum_{j=1}^{J} x_{ij} = \sum_{j=1}^{J}\sum_{k=1}^{K} y_{jk} , \qquad (4)$$

$$\sum_{i=1}^{I} v_i \le N , \qquad (5)$$

$$\sum_{j=1}^{J} z_j \le L . \qquad (6)$$

$$\sum_{j=1}^{J} y_{jk} \ge d_k , \ \forall k = \overline{1, N} . \qquad (7)$$

У цій моделі: (2) – обмеження на ємність субрегіональних центрів, а саме, що кількість медикаментів, перевезених з кожного субрегіонального центру, не більша ніж їх ємність; за допомогою (3) враховано, що кількість медикаментів, перевезених з кожного регіонального центру, не перевищує їх запас; (4) – рівняння балансу між першим та другим етапами медичної логістики. Обмеження (5) та (6) встановлюють ліміт на максимальну кількість регіональних та субрегіональних центрів. Умова (7) гарантує, що кількість медичних засобів, отриманих кожним складом не має перевищувати його ємність.

Підхід до розв'язання.

Для розв'язання задачі (1) – (7) розроблено алгоритм на основі генетичного підходу з пріоритетним кодуванням [2]. Цей алгоритм враховує необхідність розташування регіональних та субрегіональних центрів, забезпечуючи кращу географічно збалансованість. Вводяться

додаткові обмеження на максимальну кількість регіональних та субрегіональних центрів. Це дозволяє контролювати концентрацію ресурсів, гарантуючи їх ефективний розподіл. Хромосоми, що створені на етапі початкової генерації одразу задовольняють обмеженням (2) – (7). Завдяки цьому прискорюється процес розв'язання задачі. Вартість розміщення центрів обох типів інтегровано безпосередньо в цільову функцію, таким чином враховуючи цей аспект витрат в процесі оптимізації. Всі перелічені модифікації мають бути взяті до уваги при оцінюванні на кожній ітерації генетичного алгоритму. Наведемо модифікований алгоритм оцінювання хромосоми.

Крок 1. Застосувати процедуру декодування для визначення плану перевезень.

Крок 2. Якщо кількість регіональних складів більша за N, то відкидаємо хромосому, інакше переходимо до кроку 3.

Крок 3. Якщо об'єм перевезень до субрегіонального центру нульовий, то він не розміщується. Переходимо до кроку 4.

Крок 4. Якщо загальна кількість розміщених субрегіональних центрів більша за L з (5), то відкидаємо хромосому, інакше переходимо до кроку 5.

Крок 5. Якщо загальна кількість розміщених регіональних центрів більша за N з (4), то відкидаємо хромосому, інакше переходимо до кроку 6.

Крок 6. Якщо обмеження (1), (2), (3), (6) не виконуються для зазначеного плану перевезень (обмеження на об'єми перевезень), то відкидаємо хромосому, інакше переходимо до кроку 7.

Крок 7. Обчислюємо значення цільової функції за (7). Алгоритм описаний.

Модельна задача. Для проведення числових експериментів, було розроблено програмну реалізацію алгоритму розв'язання двоетапної задачі розміщення з використанням мови програмування Python та бібліотеки для розробки інтерфейсу користувача Qt5. Застосовано реальні відстані для випадково обраних медичних закладів, локації яких отримані з сервісу OpenStreetMap (OSM). OSM надає всеосяжну і загальнодоступну базу географічної інформації, включаючи дорожню мережу. Використовуючи сервіс OSM, можна отримати точні дані про відстань між медичними закладами, що дозволяє нам включати реалістичний час і відстань у наші алгоритми оптимізації логістики. Використання даних OSM гарантує, що аналіз і процес прийняття рішень ґрунтується на реальних умовах, підвищуючи надійність і практичність наших досліджень.

На рис. 1. зображено результат розв’язання модельної двоетапної задачі розміщення з такими вхідними даними: кількість можливих регіональних центрів $I = 4$; кількість можливих субрегіональних центрів $J = 4$; кількість складів $K = 5$. Вартість розміщення можливих регіональних центрів задано вектором $M = (45, 40, 45, 50)$, де значення подаються в умовних одиницях (у.о.); вартість розміщення можливих субрегіональних центрів задано вектором $N = (50, 55, 50, 45)$ у.о.; вектор запасів на регіональних складах $r = (850, 750, 800, 710)$ у.о.; вектори ємності субрегіонального центру та складів: $s = (790, 550, 680, 560)$ у.о.; $d = (200, 330, 350, 420, 400)$ у.о.. Вартості перевезення для першого та другого розраховані базуючись на реальних відстанях.

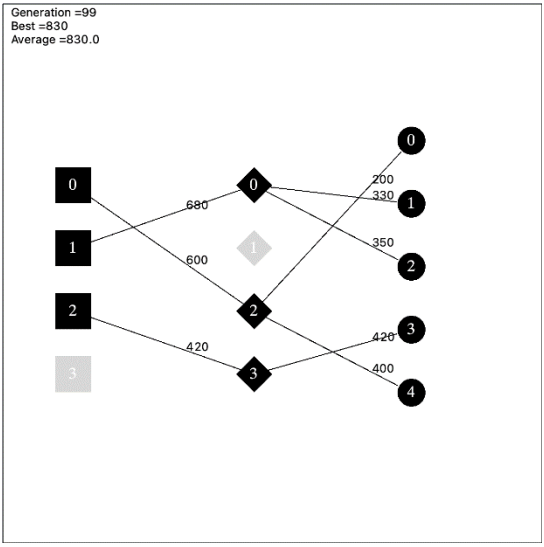


Рис. 1. - Результат розв’язання модельної двоетапної задачі розміщення

Висновки. В роботі розглянуто прикладну проблему оптимізації логістичного процесу у медичній галузі. Запропоновано використання двоетапної задачі розміщення із застосуванням генетичного підходу та пріоритетного кодування. Особливістю моделі є обмеження на максимальну кількість регіональних та субрегіональних центрів, що розміщуються. Авторами змінено процедуру оцінки хромосоми задля

поліпшення процесу побудови популяції. Запропоновані модифікації перевірено на модельній задачі з використанням реальних даних.

Робота є складовою частиною досліджень за держбюджетною науковою темою 0123U100011 «Задачі аналізу, моделювання та оптимізації технологічних процесів у складних системах різної природи», що виконуються в НТУ «Дніпровська політехніка».

Література

1. Lotfi M. A genetic algorithm using priority-based encoding with new operators for fixed charge transportation problems / M. Lotfi, R. Tavakkoli-Moghaddam. // Applied Soft Computing. – 2013. – №13. – pp. 2711–2726.

2. Mitsuo G. A genetic algorithm for two-stage transportation problem using priority-based encoding / G. Mitsuo, A. Fulya, L. Lin. // OR Spectrum. – 2006. – №28. – pp. 337–354.

ВІЗУАЛІЗАЦІЯ СПОРІДНЕНОСТІ ЛІНІЙ СОНЯШНИКУ ЗА МОРФОЛОГІЧНИМИ ОЗНАКАМИ

Сіренко Р.В.
skriler222000@gmail.com
Національний університет «Запорізька політехніка»

The author created the family relationships of sunflower lines visualization system with using graphs. Family connection is determined by the presence of a tolerance ratio, which is set by the researcher

Візуалізація спорідненості ліній соняшнику за морфологогічними ознаками включає в себе використання графічних методів та інструментів для представлення даних про фенотип рослин. Це може допомогти дослідникам виявити взаємозв'язки та патерни, які не завжди очевидні за допомогою традиційних методів аналізу.

Застосування графів дозволяє представити внутрішні зв’язки між об’єктами, які описано великими масивами даних. Для представлення даних на графах необхідно обрати тип відношення та спосіб його представлення на графі [1].

Ціллю дослідження було створення гнучкого інструменту для проведення дослідницької роботи фахівця по виявленню родинних зв’язків ліній соняшнику за статистичними даними польового експерименту, які агреговані в електронних таблицях MS Excel (рис 1).

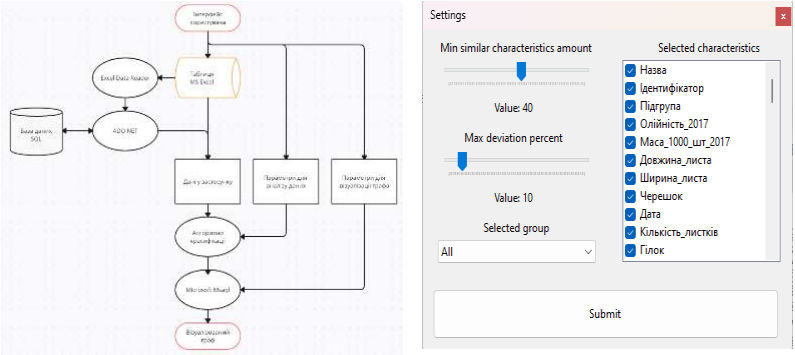
Очевидними вимогами було надання можливості опису відношення толерантності та визначення параметрів цього відношення, а також можливістю візуалізації родинних зв'язків за допомогою графа [2].

	A	B	C	D	E	F	G	H	I
3	Назва	Ідентифікатор	Підгрупа	Олійність 2017	Маса 1000 шт 2017	Довжина листа	Ширина листа	Черешок	Дата
4	ВА1Б	UE0100456	1	46.41	49.88	19.20	19.80	14.60	7-Jul
5	ВА2Б	UE0100519	1	40.71	40.84	18.20	18.00	10.70	13-Jul
6	ЮТ6	UE0100523	2	21.71	72.00	19.40	15.40	16.80	14-Jul
7	InK1589		2	14.60	43.27	16.50	11.30	13.90	16-Jul
8	LD4(без антоц	UE0100566	3	22.61	38.52	17.30	15.80	8.60	17-Jul
9	LD72/2	UE0100568	3			20.33	18.17	11.33	10-Jul
10	LD4/1	UE0100565	3			10.67	11.67	5.50	
11	LD4/2	UE0100566	3	24.50	43.02	18.40	15.70	10.60	7-Jul
12	LD72/1	UE0100567	3	20.79	46.66	16.40	14.30	9.40	7.07; 14.07
13	LD72/3	UE0100569	3	24.24	38.86	19.20	17.00	10.80	10-Jul
14	InK404	UE0100533	3	24.18	46.82	18.80	15.60	11.30	7-Jul
15	ZK224-2	UE0100595	3	28.98	51.03	16.30	13.00	10.00	7-Jul
16	InK235	UE0100919	4	24.20	43.71	14.20	12.60	9.20	3-Jul
17	AP5 56	UE0100882	4	23.72	25.56	13.00	13.40	10.00	3-Jul
18	InK103	UE0100885	4	15.33	30.79	15.70	14.40	11.20	3-Jul
19	ВИР199	UE0100549	5	34.71	60.71	18.50	19.20	8.80	14.07; 12.07
20	ВИР501	UE0101066	5			20.20	21.90	11.00	17-Jul
21	InДН47	UE0100546	5	46.92	38.50	11.20	10.30	8.00	7-Jul
22	ДН47-2	UE0100613	5	26.52	35.53	24.20	26.00	14.10	17-Jul
23	ДН47-8	UE0100616	5	22.33	30.14	15.70	15.50	6.40	14-Jul
24	ДН47-9	UE0100617	5	32.36	43.72	15.70	15.80	8.00	16-Jul
25	ЗЛ22Б	UE0100696	6	36.50	44.29	18.16	18.56	11.82	13-Jul
26	НА89Б	UE0100460	6	41.32	44.38	16.48	16.74	11.92	12.07; 13.07
27	ЗЛ22/319	UE0101290	6	43.41	45.13	17.52	15.88	16.94	10-Jul
28	ЗЛ22/320	UE0101291	6	36.64	45.90	16.22	15.78	10.70	14-Jul
29	ЗЛ22Б/ ВС6	бахрома ВС6	6			18.92	16.74	12.56	17-Jul
30	ЗЛ22/434	UE0101292	6			21.84	23.36	12.32	17-Jul
31	ЗЛ22Б/ 2	UE0100696	6	38.01	30.60	14.92	14.32	11.46	17-Jul

Рис.1 – Дані про лінії соняшнику

Було обрано такі відношення, що мають бути відображено на графі. Відношення «Приналежність групі» відображається кольором, згідно розмітки, що обрана дослідником (рис 1). Відношення толерантності описується «мінімальною кількістю схожих характеристик в межах допустимої похибки для числових значень за певними характеристиками в певній групі ліній соняшнику». Наявність такого відношення між лініями соняшнику представлено на графі ребром або дугою.

На рисунку 2 представлено основні етапи роботи програмного застосунку для дослідження спорідненості ліній соняшнику за фенотипом, такі як завантаження даних з Excel, взаємодія з базою даних, аналіз даних за допомогою методів класифікації, налаштування вхідних параметрів, побудова графа родинних зв'язків, візуалізація результатів.

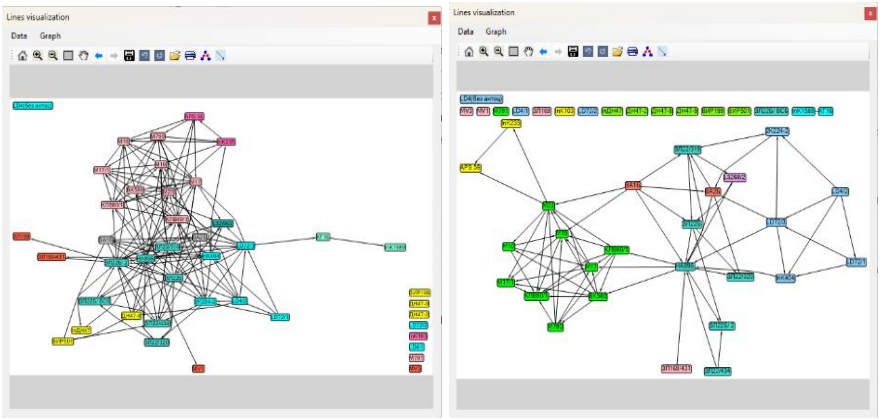


а) б)

Рис. 2 – Програмний застосунок відтворює а) структурну схему та б)надає можливість обирати параметри відношення толерантності у вікні налаштувань

Проведемо експерименти та представимо їхні результати (рис.3). Назва вершини є назвою відповідної лінії соняшнику, а лінії однієї групи мають однаковий колір. Для побудови ребра між вершинами визначається наявністю родинних зв’язків між ними. Для цього порівнюється характеристики кожної лінії з усіма іншими, використовуючи алгоритм повного перебору. Числові дані порівнюються з певною допустимою похибкою, яка за замовчуванням становить 10%. Крім того, встановлюється мінімальна кількість схожих характеристик, яка потрібна для визначення родинного зв'язку. За замовчуванням це значення становить 40 з 75 наявних характеристик.

Перших два експерименти представлені на рисунках 3 а) та 3 б). Дані для експерименту б) наводимо у дужках. Визначимо значення параметрів. Мінімальну кількість схожих характеристик оберемо 40 (50), допустиму похибку виберемо 10 (25)%. Групи всі. Порівнюючи рисунки 3 а) та 3 б) можна побачити, що кількість вільних вершин стала більшою, а більшість ліній соняшнику з кожної підгрупи пов’язані між собою або ж не пов’язані з жодної.



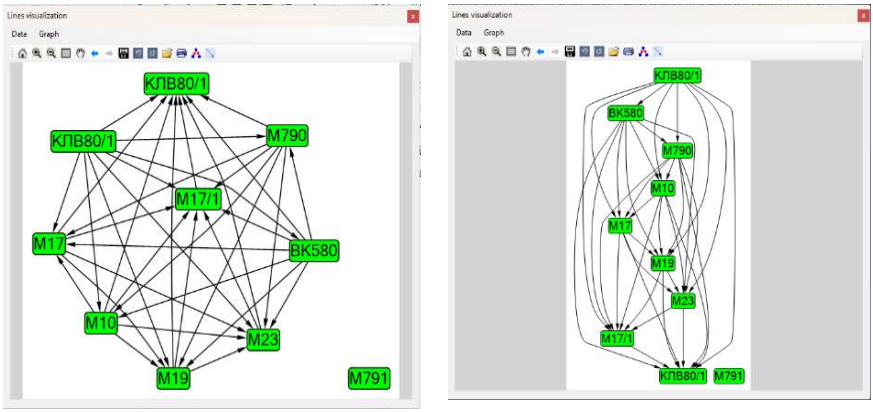
а) б)

Рис.3 – Елемент Graph Viewer з графом, який побудовано за допомогою алгоритму багатовимірного шкалювання а) з параметрами мінімальна кількість схожих характеристик 40, допустима похибка 10%, б) з параметрами мінімальна кількість схожих характеристик 50, допустима похибка 25%

Розглянемо можливості системи по представленню зав'язків різними алгоритмами побудови графів, для покращення сприйняття топологічної структури графа [3,4]. Дотримання одночасно всіх вимог до естетичного представлення графа є неможливим через їхній конкурентний характер. Серед таких правил є: мінімум перетинів ребер; суміжні вершини розташовуються поруч; вершини, з більшою кількістю зав'язків утворюють кластер; розміщення вершин та ребер рівномірно. Різні алгорими представлення графів дозволяють по-різному виконувати ці правила. Головні напрямки розробки алгоритмів візуалізації графів: силові алгоритми; енергетичні алгоритми; алгоритми зменшення розмірності, алгоритми на основі обмежень та особливостей. В роботі [4] автором складено зведену таблицю порівнянь різних алгоритмів за параметрами швидкодії, мінімуму перетинів ребер, симетрії, довжини ребер, рівномірного розподілення вершин, розміру графа.

Система дозволяє досліднику самостійно обрати пріоритетний для нього варіант. Проведемо експеримент з даними сьомої групи та застосуємо різні алгоритми для представлення родинних зав'язків. Побудуємо граф (рис. 4) для сьомої підгрупи параметрами: мінімальна кількість схожих характеристик - 40, допустима похибка - 15%. Характеристики для перевірки включають усі. Для візуалізації графа

будемо використовувати алгоритм багатовимірного шкалювання та алгоритм Сугіями.



а) б)

Рис. 4 – Граф групи 7, а) побудований за допомогою алгоритму багатовимірного шкалювання; б) побудований за допомогою алгоритму Сугіями

Як можна побачити на рисунку 4 а) та 4 б) розташування вершин змінюється, та рівень прозорості топологічної структури дослідник може обрати самостійно.

Програмний застосунок реалізований на мові програмування C# в середовищі Visual Studio зі застосуванням бібліотек ExcelDataReader та Microsoft.Msagl та технологій ADO NET.

Робота виконана в рамках договору про співробітництво між Інститутом олійних культур НААНУ та НУ «Запорізька політехніка», НДР 0121U113264 «Розвиток методів дослідження складних соціально-економічних систем на основі інтелектуальних технологій» (2021-2024рр).

Література

1. Перепелиця В.О., Козін І.В., Терещенко Е.В. Задачі класифікації: підходи, методи, алгоритми : монографія. Запоріжжя: Поліграф, 2008. 188 с.

2. Ляшенко О., Біленький О. Візуалізація даних. Дослідницька аналітика і презентація даних [Текст] / О. Ляшенко, О. Біленький. - Київ: Видавництво "Слово", 2018. - 192 с.

3. Грищенко, І. І. Візуалізація даних у вигляді графа для аналізу та дослідження великих даних. Проблеми програмування [Текст] / І. Грищенко, 2016, № 1-2, с. 158-167.

4. Сіренко, Р.С. Порівняння алгоритмів візуалізації графів [Текст] / Р. Сіренко // Інформаційні технології: теорія і практика: Тези доповідей VI-ї Всеукраїнської науково-практичної інтернет-конференції здобувачів вищої освіти і молодих учених, 2023 р., м. Харків) [Електронний ресурс]. – Харків : ХНУМГ іменіО.М.Бекетова, 2023. С.164-166

МІНІМІЗАЦІЯ ЧАСТКОВО ВИЗНАЧЕНИХ БУЛЕВИХ ФУНКЦІЙ МЕТОДОМ ОБРАЗНИХ ПЕРЕТВОРЕНЬ

Михайло Соломко

Національний університет водного господарства та природокористування

MINIMIZATION OF PARTIALLY DEFINED BOOLEAN FUNCTIONS BY THE METHOD OF FIGURATIVE TRANSFORMATIONS

Abstract. The conducted studies have established that the method of image transformations makes it possible to reduce the complexity of the minimization procedure of partially defined Boolean functions without loss of functionality, compared to sorting through binary definitions. The interpretation of the result is that the combinatorial systems 2-(n, b)-design, 2-(n, x/b)-design represent logical operations. Therefore, the detection of such combinatorial systems in the truth table of logical functions directly and unambiguously establishes the location for equivalent transformations of Boolean expressions. This, in turn, implies an algorithm for simplifying Boolean functions, including partially defined Boolean functions.

При виготовлені цифрових пристроїв виникають ситуації, коли булева функція не визначена на всіх наборах змінних. Для подолання такого типу невизначеності при створенні аналітичної моделі цифрового пристрою застосовують процедуру довизначення функції на невизначених наборах. Зазначене довизначення є потужним ступенем свободи та ресурсом якісної оптимізації частково визначених булевих функцій.

Існує два способи довизначення функцій:

– $f_1(x_1, x_2, \ldots x_n)$ на всіх невизначених наборах вихідна функція $f(x_1, x_2, \ldots x_n)$ довизначається одиницями;

– $f_0(x_1, x_2, \ldots x_n)$ на всіх невизначених наборах вихідна функція $f(x_1, x_2, \ldots x_n)$ довизначається нулями.

Тоді теорема Квайна, на разі, виводить наступне твердження: мінімальна диз'юнктивна нормальна форма (ДНФ) не повністю визначеної булевої функції визначається як диз'юнкція найбільш коротких імплікант функції. Ці імпліканти в сукупності покривають усі мінтерми досконалої диз'юнктивної нормальної форми (ДДНФ) функції, причому серед обраних простих імплікант відсутні зайві [1].

Булеві функції є окремим випадком не повністю визначених функцій. Не повністю визначені булеві функції можуть бути представлені за допомогою повністю визначених булевих функцій, які отримуються за допомогою відповідних довизначень (перебираючи всі можливі підстановки 0 або 1 замість «–» don't care – байдуже або невизначене значення) [2]. Тоді процедуру мінімізації частково визначеної булевої функції потрібно проводити на кожній визначеній функції та обрати оптимальний результат.

Зі збільшенням кількості невизначених наборів множина повністю визначених булевих функцій суттєво збільшується. Для семи, наприклад, невизначених наборів змінних існує 2^7=128 різних способів бінарного довизначення частково визначених булевих функцій, і, отже, складність процедури мінімізації функції буде зростати.

Таким чином, актуальним аспектом теоретичних наукових досліджень з мінімізації частково визначених булевих функцій методом образних перетворень є:

– зменшення кількості способів бінарного довизначення частково визначених булевих функцій;

– виявлення можливостей вдосконалення процедури мінімізації та розширення апарату синтезу цифрових компонентів на основі частково визначених булевих функцій для застосування їх у цифрових технологіях.

Виявлення локацій рівносильних перетворень за допомогою комбінаторних систем 2–(n, b)-design, 2–(n, x/b)-design

Існує два типи невизначеності для системи на практиці (рис. 1): за входом і за виходом (або вхідна дія не може надійти ззовні, або реакція системи на вхідну дію неважлива).

x_1	x_2	x_3	y
0	0	0	1
0	0	1	–
0	1	0	0
0	1	1	1
1	0	0	–
1	0	1	1
1	1	0	0
1	1	1	0

Невизначеність
Don't Care

Рис. 1. Невизначеності у системі

Частково визначену булеву функцію можна подати множиною повністю визначених булевих функцій, які отримуються за допомогою відповідних довизначень (перебираючи всі можливі підстановки 0 або 1 замість «–») (рис. 2). Тоді процедуру мінімізації частково визначеної булевої функції потрібно проводити на кожній визначеній функції та обрати оптимальний результат.

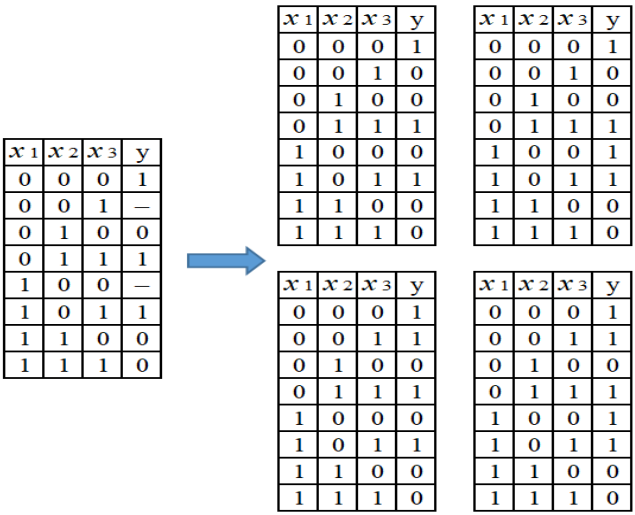


Рис. 2. Частково визначена булева функція $f(x_1, x_2, x_3)$ подана множиною з 2^2=4 повністю визначених булевих функцій $f(x_1, x_2, x_3)$

Зі збільшенням кількості невизначених наборів множина повністю визначених функцій суттєво збільшується. Для шести, наприклад, невизначених наборів змінних існує 2^6=64 різних способів бінарного довизначення частково визначених булевих функцій, і, отже, складність процедури мінімізації функції буде зростати.

Для зменшення кількості способів бінарного довизначення частково визначених булевих функцій потрібно обирати таку бінарну конфігурацію, яка забезпечить умови для проведення операцій супер-склеювання змінних та/або простого склеювання змінних. А також буде включати в себе визначені набори змінних і частину невизначених наборів змінних, зокрема. Для ряду частково визначених булевих функцій оптимальна їх мінімізація буде вимагати використання всіх невизначених наборів змінних.

Комбінаторні системи 2–(*n*, *b*)-design, 2–(*n, x/b*)-design, які входять до складу бінарних структур таблиць істинності забезпечують однозначне виявлення локацій рівносильних перетворень для спрощення булевих функцій. Встановлене місце рівносильних перетворень імплікує алгоритм мінімізації булевих функцій. Таким чином системи 2–(*n*, *b*)-design, 2–(*n, x/b*)-design мають інформаційну ємність, що дозволяє з ефектом замінити бінарне довизначення частково визначених булевих функцій, як то на рис. 2, на метод образних перетворень. Це, у свою чергу, дає змогу без втрати функціональності зменшити складність процедури мінімізації, порівняно з перебором бінарних довизначень частково визначених булевих функцій.

Приклад 1. Частково визначена булева функція *f*(*A*, *B*, *C*, *D*) задана таблицею істинності (табл. 1). Знайти мінімальну ДНФ [3].

Таблиця 1

Таблиця істинності частково визначеної булевої функції *f*(*A*, *B*, *C*, *D*)

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x_1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
x_2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
x_3	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
x_4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
f	1	0	−	−	1	0	−	0	0	0	0	−	1	0	0	0

Розв'язок:

$f_{\text{ТДНФ}}(A,B,C,D) =$

№	A	B	C	D	f
0	0	0	0	0	1
4	0	1	0	0	1
12	1	1	0	0	1
2	0	0	1	0	–
3	0	0	1	1	–
6	0	1	1	0	–
11	1	0	1	1	–

=

A	B	C	D	f
0			0	1
1	1	0	0	1
0	0	1	1	–
1	0	1	1	–

=

A	B	C	D	f
0			0	1
	1	0	0	1
0	0	1	1	–
1	0	1	1	–

. (1)

Перша матриця виразу (1) представляє таблицю істинності частково визначеної ДНФ функції $f(A, B, C, D)$ (табл. 1). До блоків 0, 4, 2, 6 застосовано операцію супер-склеювання змінних [4]. Результат логічних операцій першої матриці записано до другої матриці виразу (1).

Мінімальна ДНФ має вигляд:

$$f(A,B,C,D) = \overline{A}\,\overline{D} + B\overline{C}\,\overline{D}. \qquad (2)$$

Відмічаємо, що невизначені набори змінних 3, 11 під час спрощення функції $f(A, B, C, D)$ (табл. 1) методом образних перетворень не були використані. Це у підсумку зменшило загальну складність спрощення функції. Результат мінімізації (2) співпадає з [3].

На відміну від бінарних довизначень, в розглянутому прикладі, всього одна комбінаторна система 2–(2, 4)-design встановлює локацію потрібних рівносильних перетворень. Це зменшує складність процедури мінімізації, порівняно з перебором довизначених функцій, дає результат спрощення без втрати функціональності заданої булевої функції та забезпечує ефективність заміни бінарних довизначень на метод образних перетворень.

Точний та евристичний метод спрощення частково визначених булевих функцій

За допомогою точного методу спрощення булевих функцій можна оцінювати якість евристичних методів, і навіть визначати напрямок створення евристичного методу [5].

Приклад 2. Методом образних перетворень спростити ДНФ частково визначеної функції f(x1, x2, x3, x4, x5, x6), яка представлена картою Карно [6].

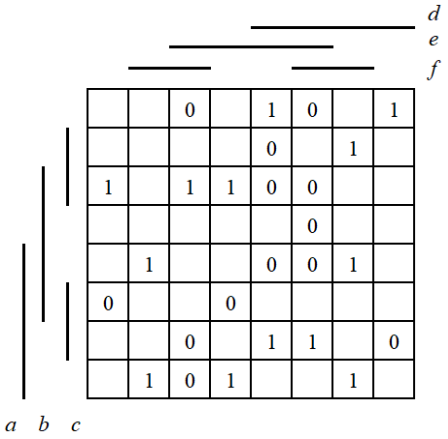


Рис. 3. Частково визначена булева функція шести змінних (карта Карно)

Розв'язок:

Перша матриця виразу (3) представляє таблицю істинності частково визначеної ДНФ функції $f(x_1,x_2,x_3,x_4,x_5,x_6)$ (рис. 3).

До блоків 13, 33, 37, 49, 53, 1, 5, 9, 17, 21, 25, 29, 41, 45, 57, 61, що складають повну комбінаторну систему 2–(4, 16)-design та до блоків 4, 6, 34, 0, 2, 32, 36, 38, що складають повну комбінаторну систему 2–(3, 8)-design, а також до блоків 46, 47, 62, 63, що складають повну комбінаторну систему 2–(2, 4)-design застосуємо операцію супер-склеювання змінних [4]. До блоків 24, 16 та 26, 18, а також до блоків 27, 19 застосуємо операцію простого склеювання змінних. Результат зазначених логічних операцій записуємо до другої матриці виразу (3). У другій матриці виразу (3) невизначені набори змінних не відображаються, оскільки вони не приймають участь у подальшому спрощенні функції.

У підсумку отримана мінімальна функція

$$f_{\text{мднф}} = \overline{x}_2\overline{x}_3\overline{x}_6 + \overline{x}_5x_6 + \overline{x}_1x_2\overline{x}_4 + x_1x_3x_4x_5, \qquad (4)$$

яка містить на два літерали менше, порівняно зі спрощенням за допомогою евристичного методу [6].

Кошт реалізації отриманої мінімальної функції (4) методом образних перетворень складає $k_0 / k_l / k_{in} = 4 / 12 / 6$, де k_0, k_l, k_{in} – кількість кон'юнктермів, літералів та інверторів відповідно.

$F(x_1, x_2, x_3, x_4, x_5, x_6) =$

№	x_1	x_2	x_3	x_4	x_5	x_6	f
4	0	0	0	1	0	0	1
6	0	0	0	1	1	0	1
13	0	0	1	1	0	1	1
24	0	1	1	0	0	0	1
26	0	1	1	0	1	0	1
27	0	1	1	0	1	1	1
33	1	0	0	0	0	1	1
34	1	0	0	0	1	0	1
37	1	0	0	1	0	1	1
46	1	0	1	1	1	0	1
47	1	0	1	1	1	1	1
49	1	1	0	0	0	1	1
53	1	1	0	1	0	1	1
0	0	0	0	0	0	0	–
1	0	0	0	0	0	1	–
2	0	0	0	0	1	0	–
5	0	0	0	1	0	1	–
8	0	0	1	0	0	0	–
9	0	0	1	0	0	1	–
10	0	0	1	0	1	0	–
11	0	0	1	0	1	1	–
12	0	0	1	1	0	0	–
15	0	0	1	1	1	1	–
16	0	1	0	0	0	0	–
17	0	1	0	0	0	1	–
18	0	1	0	0	1	0	–

=

№	x_1	x_2	x_3	x_4	x_5	x_6	f
19	0	1	0	0	1	1	–
20	0	1	0	1	0	0	–
21	0	1	0	1	0	1	–
22	0	1	0	1	1	0	–
25	0	1	1	0	0	1	–
28	0	1	1	1	0	0	–
29	0	1	1	1	0	1	–
32	1	0	0	0	0	0	–
36	1	0	0	1	0	0	–
38	1	0	0	1	1	0	–
39	1	0	0	1	1	1	–
40	1	0	1	0	0	0	–
41	1	0	1	0	0	1	–
42	1	0	1	0	1	0	–
45	1	0	1	1	0	1	–
48	1	1	0	0	0	0	–
50	1	1	0	0	1	0	–
51	1	1	0	0	1	1	–
52	1	1	0	1	0	0	–
57	1	1	1	0	0	1	–
59	1	1	1	0	1	1	–
60	1	1	1	1	0	0	–
61	1	1	1	1	0	1	–
62	1	1	1	1	1	0	–
63	1	1	1	1	1	1	–

=

№	x_1	x_2	x_3	x_4	x_5	x_6	f
–		0	0			0	1
–					0	1	1
–	0	1		0	0		1
–	0	1		0	1	0	1
–	0	1		0		1	1
–	1		1	1	1		1

=

№	x_1	x_2	x_3	x_4	x_5	x_6	f
–		0	0			0	1
–					0	1	1
–	0	1		0	0		1
–	0	1		0	1	0	1
–	0	1		0		1	1
–	1		1	1	1		1
–	0	1		0	1		–

=

№	x_1	x_2	x_3	x_4	x_5	x_6	f
–		0	0			0	1
–					0	1	1
–	0	1		0			1
–	1		1	1	1		1

=

№	x_1	x_2	x_3	x_4	x_5	x_6	f
–		0	0			0	1
–					0	1	1
–	0	1		0			1
–	1		1	1	1		1

. (3)

Мінімальна функція, отримана евристичним методом [6] має вигляд:

$$f_{\text{мдноф}} = \bar{e}f + \bar{b}\bar{c}\bar{f} + \bar{c}e + \bar{a}\bar{c}\bar{d} + acde,$$

з коштом реалізації $k_0 / k_1 / k_{in} = 5 / 14 / 8.$

Література

1. Савельев А. Я. Прикладная теория цифровых автоматов. – М.: Высшая школа, 1987. – 272 с. URL: https://vdoc.pub/documents/-4o35jbu52gg0 (дата обращения: 20. 06. 2023).

2. Прихожий А. А. Частично определённые логические системы и алгоритмы. – Минск: БНТУ, 2013. – 343 с. URL: https://rep.bntu.by/handle/data/37237 (дата обращения: 20. 06. 2023).

3. Мінімізація неповністю визначених логічних функцій. Дискретна математика / Сумський державний університет. URL: https://studfile.net/preview/14499737/page:17/ (дата звернення: 20. 06. 2023).

4. Різник, В.В., Соломко, М.Т. Застосування алгебричної операції супер-склеювання змінних для мінімізації булевих функцій комбінаторни методом // Технологічний аудит та резерви виробництва. – Вип. 6/2 (38), 2017. С. 60 – 76. URL: http://journals.uran.ua/tarp/article/viewFile/118336/112951 (дата звернення: 20. 06. 2023).

5. Поттосин, Ю. В. Минимизация булевых функций в классе ортогональных дизъюнктивных нормальных форм / Ю. В. Поттосин // Информатика. – 2021. − Т. 18, № 2. – С. 33–47. https://doi.org/10.37661/1816-0301-2021-18-2-33-47. URL: https://inf.grid.by/jour/article/view/1131 (дата обращения: 20. 06. 2023).

6. Закревский, А.Д. Торопов, Н.Р. Романов, В.И. ДНФ-реализация частичных булевых функций многих переменных // Информатика. – 2010. − № 1(25). – С. 102–112. URL: https://inf.grid.by/jour/article/view/461/419 (дата обращения: 20. 06. 2023).

ВИПАДКОВИЙ ЛІС: ПОШУК АНОМАЛІЙ У КОМП'ЮТЕРНИХ МЕРЕЖАХ

Денис Сташук, Валерій Дубровін, Юлія Тарасова

Національний університет «Запорізька політехніка»

FRESHCODE

RANDOM FOREST: FINDING ANOMALIES IN COMPUTER NETWORKS

Abstract. Random Forest is one of the most popular and commonly used algorithms by Data Scientists. Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression. One of the most important features of the Random Forest Algorithm is that it can handle the data set containing continuous variables, as in the case of regression, and categorical variables, as in the case of classification.

Постановка проблеми. Виявлення аномалій поведінки мережі – це процес моніторингу корпоративних мереж для виявлення ненормальної поведінки. Після виявлення аномалії функція виявлення аномалії поведінки мережі або ініціює автоматичну відповідь, або сповіщає групи безпеки. Проблеми в комп'ютерних мережах виявляються як викликані ними аномалії трафіку[1]. Загалом, аномалія – це те, що суперечить очікуванням. Наприклад, пошкоджений комутатор може створити неочікуваний трафік в іншій частині мережі або нові коди помилок починають з’являтися, коли служба не працює. Усунення несправностей мережі ґрунтується на аномаліях мережі. Перший метод класифікації аномалій заснований на тому, чим вони відрізняються від звичайного спілкування. Аномалії можуть відрізнятися за типом переданих даних (поведінкові), за кількістю переданих даних (за обсягом) або за обома критеріями. Інший спосіб класифікації аномалій - це залежно від їх причини:

1. Нелюдська помилка - збій обладнання або переривання радіозв'язку через погоду.
2. Людська помилка - збій служби мережі, спричинений неправильною конфігурацією або випадковим від’єднанням мережевого кабелю.
3. Зловмисна діяльність людини – внутрішня атака, коли незадоволений співробітник компанії пошкоджує мережевий принтер, або зовнішня атака, коли зловмисник намагається вимкнути мережу та завдати шкоди репутації.

Одним з методів виявлення аномалій у комп’ютерних мережах є випадковий ліс[2]. Для дослідження роботи алгоритму випадкового лісу розглянемо його роботу у гібридній системі виявлення вторгнень на основі комбінації випадкового лісу та автоенкодера.

Модель системи. Оскільки методи мережевих атак ускладнюються, отримати зразки всіх типів атак стає складно. Під час зустрічі з невідомими атаками під час фази виявлення контрольований класифікатор може погано узагальнювати результат, що може призвести до неправильної класифікації для цих зразків і зниження рівня виявлення. Щоб впоратися з цією проблемою, дослідники спробували створити модель системи виявлення вторгнень, яка тренується тільки на звичайних даних, таких як автоенкодер[3]. Він представляє собою спеціальну неконтрольовану нейронну мережу. Оскільки вона може відновити вхідні дані, помилка реконструкції може служити показником аномалії для її виявлення. Автоенкодер зазвичай має структуру симетрії. Його можна розділити на дві частини: кодер і декодер. Така система намагається якомога більше реконструювати оригінальні вхідні дані. І кодер, і декодер складаються з кількох прихованих шарів. Зокрема, кодер генерує приховане представлення для вхідного зразка. Зазвичай латентні представлення мають нижчу розмірність, ніж оригінальні вхідні дані. Декодер намагається відновити оригінальний вхід зі стиснутої форми. Автоенкодер використовує помилку реконструкції як оцінку аномалії, де один зразок із вищим показником може бути виявлений як атака. Однак через відсутність контролю з боку звичайних зразків і зразків атаки він може не отримати такої ж високої продуктивності, як контрольовані алгоритми, які можуть вивчати межі складних рішень між звичайними зразками та зразками атаки.

Розробка надійної моделі системи виявлення вторгнень з вищим рівнем виявлення та нижчим рівнем помилкових позитивних результатів стає критичною. Зважаючи на те, що різні методи можуть вивчати характеристики даних з різних точок зору, краще використовувати гібридну систему[4], яка поєднує випадковий ліс і автоенкодер. Зокрема, гібридна система включає два етапи на етапі виявлення. Першим кроком є застосування випадкового лісу з імовірнісними методами для виявлення атак. Тоді, розглядаючи, як ще більше зменшити частоту помилкових тривог, на другому кроці використовується модуль автоенкодера.

Алгоритм випадкового лісу. Щоб отримати кращу ефективність прогнозування, ансамблеві методи використовують кілька основних класифікаторів для прийняття рішень. Цей підхід загалом демонструє підвищену продуктивність класифікації порівняно з одним базовим класифікатором. Випадковий ліс є потужним класифікатором ансамблю. Він поєднує пакетування та випадковість функцій для навчання кількох дерев рішень. Дерево рішень є часто використовуваним класифікаційним

підходом. Він намагається вивчити набір правил «якщо-тоді» для класифікації даних. Ілюстрація випадкового лісу показана на рис.1.

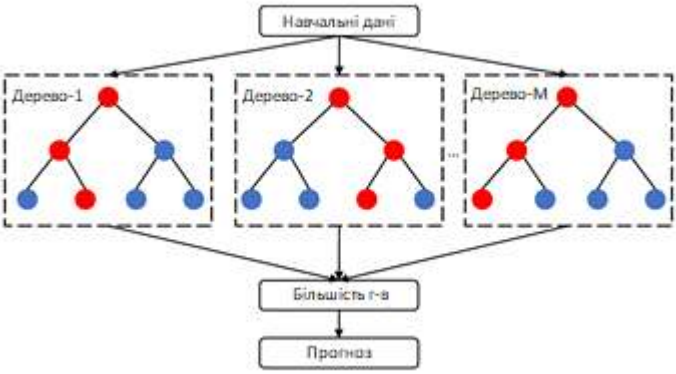


Рис1. - Ілюстрація випадкового лісу

Підхід пакетування є однією з часто використовуваних стратегій ансамблю для побудови кількох класифікаторів. За допомогою початкового завантаження із заміною для вибірки даних велика кількість класифікаторів навчається незалежно. Як показано на рисунку, з урахуванням навчального набору існує *M* різних класифікаторів рішення. Для отримання остаточних результатів передбачення використовується мажоритарне голосування для об'єднання прогнозів від кожного дерева рішень [5].

Щоб побудувати одне дерево рішень, спочатку вибираємо k ознак із j ознак. Значення k встановлюється як $sqrt(j)$. Після цього вибираємо найкращу функцію розділення з вибраних k функцій і ділимо вузол на два дочірні вузли. Домішка Джіні використовується як критерій розщеплення для розділеного вузла. Щоб передбачити вхідну вибірку, випадковий ліс використовує голоси дерев у лісі, зважені за їхніми оцінками ймовірності. Нехай p позначає ймовірність бути передбаченим як атака. Тоді ймовірність бути передбаченим як нормальний дорівнює q і $q = 1- p$. Прогнозовані ймовірності класів вхідної вибірки обчислюються як середні прогнозовані ймовірності класів дерев у лісі. Імовірність класу окремого дерева – це частка зразків одного класу в листовому вузлі [6].

Зазвичай зразки можна віднести до одного класу з найбільшою ймовірністю. Однак у такому підході деякі зразки, що належать до невідомих атак, можуть бути помилково класифіковані як нормальні. У цій частині потрібно визначити поріг T, щоб допомогти процесу

визначення зразків. Якщо ймовірність p, що належить до атаки, перевищує поріг T, вибірки можна класифікувати як атаку. Таким чином, із зразком x_i і відповідною ймовірністю p_i, ми визначаємо вирішувальну функцію $f(\cdot)$. Результати позначені ±1, де +1 - це аномальні зразки:

$$f(x_i) = \begin{cases} -1, & if\ p_i \leq T \\ +1, & if\ p_i > T \end{cases}$$

Переваги алгоритму:

1. Його можна використовувати в задачах класифікації та регресії.

2. Вирішує проблему переобладнання, оскільки вихід базується на більшості голосів або усередненні.

3. Добре працює, навіть якщо дані містять нульові/відсутні значення.

4. Кожне створене дерево рішень є незалежним від іншого; таким чином, він демонструє властивість розпаралелювання.

5. Стабільний, оскільки беруться середні відповіді, надані великою кількістю дерев.

6. Зберігає різноманітність, оскільки не всі атрибути враховуються під час створення кожного дерева рішень, хоча це не так у всіх випадках.

7. Несприйнятливий до розмірності. Оскільки кожне дерево не враховує всі атрибути, простір функцій зменшується.

8. Не потрібно розділяти дані на тренування та тестування, оскільки завжди буде певний відсоток даних, які не видно дереву рішень, створеному за допомогою початкового завантаження.

Недоліки алгоритму:

1. Дуже складний порівняно з деревами рішень, де рішення можна приймати, дотримуючись шляху дерева.

2. Час навчання більше, ніж інші моделі через його складність.

Таким чином, випадковий ліс є прийнятним вибором, якщо хтось хоче створити модель швидко та ефективно, оскільки одна з найкращих переваг випадкового лісу полягає в тому, що він може обробляти відсутні значення. Це один із найкращих методів із високою ефективністю, який широко використовується в різних галузях промисловості завдяки своїй ефективності. Він може обробляти двійкові, безперервні та категоричні дані. Загалом випадковий ліс є швидкою, простою, гнучкою та надійною моделлю з деякими обмеженнями.

Література

1. J. P. Anderson, Computer Security Threat Monitoring and Surveillance, Technical Report, James Anderson Report, Pennsylvania, (1980).

2. L. Breiman, Random Forests, Machine Learning, vol. 45, no. 1, pp. 5–32, (2001).

3. A. Hwoij, M. Al-Kasassbeh, and M. Al-Fayoumi, “Detecting network anomalies using rule-based machine learning within SNMP-MIB dataset,” *Cryptography and Security*, 2020, arXiv:2002.02368.

4. Breiman, L. Random Forests. *Mach. Learn.* **2001**, 45, 5–32.

5. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, 12, 2825–2830.

6. Géron, A. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*; O’Reilly Media, Inc.: Sebastopol, CA, USA, 2022.

ДВОЕТАПНА ТРАНСПОРТНА ЗАДАЧА З УМОВАМИ НА ПОТРЕБИ СПОЖИВАЧІВ ТА ПРОПУСКНІ СПРОМОЖНОСТІ ПРОМІЖНИХ ПУНКТІВ[1]

Стецюк П.І., Хом’як О.М.
stetsyukp@gmail.com
Інститут кібернетики імені В.М. Глушкова НАН України

A TWO-STAGE TRANSPORTATION PROBLEM WITH CONDITIONS ON CONSUMER NEEDS AND CAPACITIES OF INTERMEDIATE POINTS

Abstract. A mathematical model of a two-stage transportation problem with two-sided constraints on consumer needs and upper bounds on the capacities of intermediate points is proposed. Its partial case is the classic two-stage transportation problem, which determines the most economical plan for the transportation of products from suppliers to consumers through intermediate points. The results of computational experiments using the Gurobi solver for discrete analog of optimal set partitioning model problem are presented.

У статті [1] запропоновано математичну модель двоетапної транспортної задачі з двосторонніми обмеженнями на невідомі потреби

[1] Робота підтримана Volkswagen Foundation (грант № 97775)

споживачів. Нижче доповнимо її можливістю врахування верхніх границь на пропускні спроможності пропускних пунктів. Розглянемо застосування нової моделі для розв'язання модельної задачі оптимального розбиття множини та наведемо результати обчислювальних експериментів з використанням солвера Gurobi.

Формулювання задачі. Нехай в m пунктах постачання $A_1,...,A_m$ є $a_1,...,a_m$ одиниць продукції, яку потрібно перевезти до n споживачів $B_1,...,B_n$. Об'єми потреб споживачів будемо вважати невідомими, а їх нижні межі $b_1^{low},...,b_n^{low}$ та верхні межі $b_1^{up},...,b_n^{up}$ будемо вважати заданими. Транспортувати продукцію від постачальників до споживачів можна через l проміжних пунктів $D_1,...,D_l$, де $d_1^{up},...,d_l^{up}$ задають їх максимальні спроможності. Потрібно знайти оптимальний план транспортування продукції, де c_{ik} – витрати на перевезення одиниці продукції від постачальника A_i до проміжного пункту D_k, а c_{kj} – витрати на перевезення одиниці продукції від проміжного пункту D_k до споживача B_j, та відповідні оптимальному плану потреби споживачів.

Нехай $x=\{x_{ik}\}_{i=1,...,m}^{k=1,...,l}$, де x_{ik} – кількість одиниць продукції, яка перевозиться від постачальника A_i до пункту D_k; $y=\{y_{kj}\}_{k=1,...,l}^{j=1,...,n}$, де y_{kj} – кількість продукції від пункту D_k до споживача B_j; $z=\{z_j\}_{j=1,...,n}$, де z_j – кількість продукції, яка постачається споживачу B_j.

Двоетапна транспортна задача з обмеженнями на потреби споживачів та пропускні спроможності проміжних пунктів має такий вигляд: знайти

$$f_{xyz}^{*}=f\left(x^{*},y^{*},z^{*}\right)=\min_{x,y}\left\{f\left(x,y,z\right)=\sum_{i=1}^{m}\sum_{k=1}^{l}c_{ik}x_{ik}+\sum_{k=1}^{l}\sum_{j=1}^{n}c_{kj}y_{kj}\right\} \tag{1}$$

за обмежень

$$\sum_{k=1}^{l}x_{ik}=a_i,\ i=\overline{1,m}, \tag{2}$$

$$\sum_{k=1}^{l}y_{kj}=z_j,\ j=\overline{1,n}, \tag{3}$$

$$\sum_{i=1}^{m} x_{ik} - \sum_{j=1}^{n} y_{kj} = 0,\ k = \overline{1,l}, \tag{4}$$

$$\sum_{i=1}^{m} x_{ik} \le d_k^{up},\ k = \overline{1,l} \tag{5}$$

$$\sum_{j=1}^{n} z_j = \sum_{i=1}^{m} a_i, \tag{6}$$

$$b_j^{low} \le z_j \le b_j^{up}, \quad j = \overline{1,n}, \tag{7}$$

$$x_{ik} \ge 0,\ y_{kj} \ge 0,\ i = \overline{1,m},\ k = \overline{1,l},\ j = \overline{1,n}. \tag{8}$$

Задача (1) – (8) є задачею лінійного програмування (ЛП-задача), де цільова функція $f(x, y, z)$ задає сумарні витрати на транспортування продукції від постачальників до проміжних пунктів та від проміжних пунктів до споживачів. Обмеження (2) означають транспортування $a_1, ..., a_m$ одиниць продукції із пунктів постачання до проміжних пунктів, а обмеження (3) – що споживачам потрібно доставити невідомі об'єми $z_1, ..., z_n$ одиниць продукції з проміжних пунктів. Обмеження (4) задають умови на те, щоб уся продукція, яка приходить від постачальників до кожного проміжного пункту, була обов'язково відправлена споживачам. Обмеження (5) задають верхні межі на пропускні спроможності проміжних пунктів. Їх також можна записати так:

$$\sum_{j=1}^{n} y_{kj} \le d_k^{up},\ k = \overline{1,l}. \tag{9}$$

Обмеження (6) задає умову на те, щоб сумарна продукція постачальників дорівнювала сумарній продукції виробників. Обмеження (7) задає нижні та верхні межі на невідомі потреби споживачів.

Твердження 1. *Система обмежень (2) – (8) є сумісною тоді й лише тоді, коли виконуються умови*

$$\sum_{j=1}^{n} b_j^{low} \le \sum_{i=1}^{m} a_i \le \sum_{j=1}^{n} b_j^{up}, \quad \sum_{i=1}^{m} a_i \le \sum_{k=1}^{l} d_k^{up} \tag{10}$$

Перша частина умов (10) означає, що обмеження (2) – (5), (6) є лінійно залежними та одну з рівностей в обмеженнях (2) або (4) може

бути вилучено, причому довільну.

Частковий випадок задачі (1) – (9). Якщо $d_k^{up} = \sum_{i=1}^{m} a_i,\ k = \overline{1,l},$ $b_j^{low} = b_j^{up} = b_j,\ j = \overline{1,n},$ то задача (1) – (7) переходить в класичну двоетапну транспортну задачу, яка має такий вигляд: знайти

$$f_{xy}^* = f\left(x^*, y^*\right) = \min_{x,y}\left\{ f\left(x,y\right) = \sum_{i=1}^{m}\sum_{k=1}^{l} c_{ik}x_{ik} + \sum_{k=1}^{l}\sum_{j=1}^{n} c_{kj}y_{kj} \right\} \tag{11}$$

за обмежень

$$\sum_{k=1}^{l} x_{ik} = a_i,\ i = \overline{1,m}, \tag{12}$$

$$\sum_{k=1}^{l} y_{kj} = b_j,\ j = \overline{1,n}, \tag{13}$$

$$\sum_{i=1}^{m} x_{ik} - \sum_{j=1}^{n} y_{kj} = 0,\ k = \overline{1,l}, \tag{14}$$

$$x_{ik} \geq 0,\ y_{kj} \geq 0,\ i = \overline{1,m},\ k = \overline{1,l},\ j = \overline{1,n}. \tag{15}$$

Задача (11) – (15) є ЛП-задачею, де цільова функція $f\left(x,y\right)$ задає сумарні витрати на транспортування продукції від постачальників до споживачів. Обмеження (12) гарантують транспортування $a_1,...,a_m$ одиниць продукції із пунктів постачання до проміжних пунктів, а обмеження (13) гарантують, що споживачам буде доставлено необхідні об'єми $b_1,...,b_n$ одиниць продукції з проміжних пунктів. Обмеження (14) задають умови на те, щоб уся продукція, яка приходить від постачальників до кожного проміжного пункту, була обов'язково відправлена споживачам.

Твердження 2 [2]. *Система обмежень (12) – (15) є сумісною тоді й лише тоді, коли виконується умова*

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j. \tag{16}$$

Рівність в умові (16) означає, що обмеження (12) – (14) є лінійно залежними, тому одну довільну рівність з обмежень (12), (13) та (14)

може бути вилучено.

Обчислювальний експеримент. Розглянемо застосування задачі (1) – (8) до модельного прикладу задачі оптимального розбиття-розподілу, описаної в статті [3]. Деякий постачальник однорідного ресурсу, неперервно розподілений зі щільністю $\rho(x)=1$ в області $\Omega=\left\{x=\left(x^{(1)},x^{(2)}\right):\ 0\le x^{(1)}\le 1,\ 0\le x^{(2)}\le 1\right\}$, постачає його в п'ять пунктів (першого етапу) для зберігання. Задано координати цих пунктів та координати пунктів (другого етапу) споживання ресурсу, що зберігався в пунктах першого етапу (рис.1.).

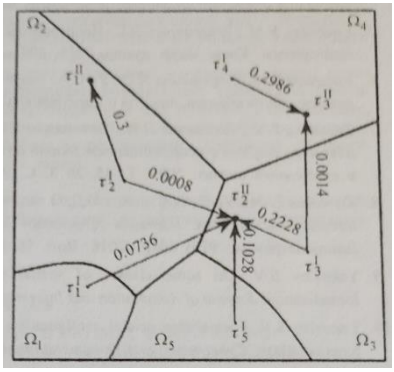


Рис.1 – Модельний приклад задачі оптимального розбиття-розподілу [3].

Потрібно розбити множину Ω постачальників ресурсу на сфери їх обслуговування в п’яти пунктах першого етапу і визначити об'єми перевезень v_{ij}, $i=\overline{1,5}$, $j=\overline{1,3}$, від пунктів першого етапу в пункти другого етапу так, щоб мінімізувати сумарну вартість транспортування ресурсу від постачальників в пункти першого етапу і доставки ресурсу в пункти другого етапу.

Модельний приклад можна описати за допомогою задачі (1) – (8), де постачальниками є m ділянок з дискретного розбиття області Ω, споживачами є три пункти другого етапу, а проміжними пунктами – п’ять пунктів першого етапу. Якщо обсяги ресурсу постачальників вибрати пропорційними площам ділянок, то розв’язок задачі (1) – (8) буде відповідати одному з варіантів оптимального розбиття-розподілу множини Ω з точністю до вибраної дискретизації ділянок.

Два оптимальні розбиття множини Ω для квадратної сітки 500×500 наведено на рис. 2. На рис. 2, а представлено оптимальне розбиття множини Ω для $d_k^{up}=\sum_{i=1}^{m}a_i,\ k=\overline{1,l},$ а на рис. 2, б – для $d_k^{up}=0.25\times\sum_{i=1}^{m}a_i,\ k=\overline{1,l}.$ При цьому об'єми споживання у пунктах другого етапу були рівними 0.22896; 0.428868; 0.342172 (рис. 2, а, $f_{xyz}^{*}=0,477676$) та 0.25; 0.5; 0.25 (рис. 2, б, $f_{xyz}^{*}=0,485096$). Для вибраної сітки задачі (1) – (8) мають 1.250.018 змінних та 250.014 лінійних обмежень. Вони розв'язувалися за допомогою солвера Gurobi за декілька секунд на NEOS-сервері.

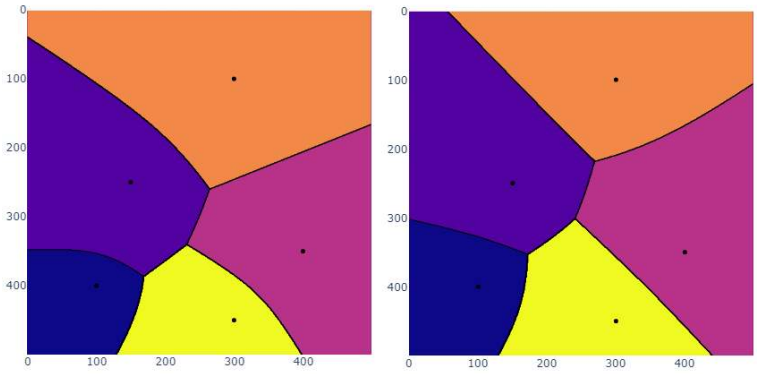


Рис. 2 – Два оптимальні розбиття множини Ω: $0\le b_i\le1,\ i=\overline{1,3}$.

План перевезення v_a відповідає оптимальному розбиттю з рис. 2, а, план перевезення $v_б$ – оптимальному розбиттю з рис. 2, б:

$$v_a=\begin{pmatrix}0.0000 & 0.09066 & 0.0000\\ 0.22896 & 0.0000 & 0.0000\\ 0.0000 & 0.232524 & 0.0000\\ 0.0000 & 0.0000 & 0.342172\\ 0.0000 & 0.105684 & 0.0000\end{pmatrix},\ v_б=\begin{pmatrix}0.0000 & 0.112212 & 0.0000\\ 0.2500 & 0.0000 & 0.0000\\ 0.0000 & 0.2500 & 0.0000\\ 0.0000 & 0.0000 & 0.2500\\ 0.0000 & 0.137788 & 0.0000\end{pmatrix}.$$

Висновок. Запропоновано математичну модель двоетапної транспортної задачі з двосторонніми обмеженнями на потреби споживачів та верхніми границями на пропускні спроможності

проміжних пунктів. Її частковим випадком є класична двоетапна транспортна задача, яка визначає найбільш економічний план транспортування продукції від постачальників до споживачів через проміжні пункти. Наведено результати обчислювальних експериментів з використанням солвера Gurobi для дискретного аналога модельної задачі оптимального розбиття-розподілу множини з заданими центрами підмножин.

Література

1. Стецюк П.І., Хом'як О.М., Ляшко В.І. Двоетапна транспортна задача з невідомими потребами споживачів. Наукові записки НаУКМА. Комп'ютерні науки. 2022. Т. 5.

2. Карагодова О.О., Кігель В.Р., Рожок В.Д. Дослідження операцій: Навч. посіб. К.:Центр учбової літератури, 2007. 256 с.

3. Киселева Е.М., Притоманова О.М., Ус С.А. Решение двухэтапной непрерывно-дискретной задачи оптимального разбиения-распределения с заданным положением центров подмножеств. Кибернетика и системный анализ. 2020. № 1. С. 3-15.

ВЛАСТИВІСТЬ ПОВНОТИ АБО КВАЗІПОВНОТИ БАГАТОКРИТЕРІАЛЬНИХ ЗАДАЧ ПРИ ЗБІЛЬШЕННІ КІЛЬКОСТІ КРИТЕРІЇВ

Терещенко Е.В., Рябенко А.Є.
elina_vt@ukr.net, rjabenkoae@gmail.com
Національний університет «Запорізька політехніка»

The authors conducted research on the influence of the dimension and composition of the criteria vector on the property of completeness / quasi-completeness of multi-criteria discrete optimization problems based on the assessment the cardinalities of sets of feasible solutions, the Pareto set, and a complete set of alternatives

В попередній роботі авторів [1] було доведено достатні умови наявності властивості повноти або квазіповноти двокритеріальних задач дискретної оптимізації задачі Z_{γ_1,γ_2}, де Z_{γ_1,γ_2} є двокритеріальною задачею на двозваженому $n-$вершинному $m-$реберному графі з мінімізуємими критеріями $F_{\gamma_i}(x) \to min,\ i = \overline{1,2}$ векторної цільової функції $F_{\gamma_1,\gamma_2}(x_{\gamma_1,\gamma_2})$. Індекси $\gamma_i,\ i = \overline{1,2}$ є індивідуальними номерами

критеріїв, та їхній порядок $F_{\gamma_i}(x)$, визначає порядок розв'язання задач Z_{γ_i}.

Багатокритеріальна задача Z є повною, якщо для множини допустимих рішень (МДР) $X = \{x\}$ існують такі параметри векторної цільової функції (ВЦФ) $F(x)$, при яких виконується рівність потужності множини допустимих рішень X^0, паретівської множини $\tilde{X}$, повної множини альтернатив X ,

тобто $X^0 = \tilde{X} = X$ [2].

Багатокритеріальна задача Z є квазіповною, якщо для множини допустимих рішень X^0, паретівської множини $\tilde{X}$, повної множини альтернатив X є справедливим співвідношення $X^0 = \tilde{X} < X$ [1].

Теорема 1. Для того щоб двохкритеріальна задача Z_{γ_1,γ_2} з ВЦФ $F_{\gamma_1,\gamma_2}(x_{\gamma_1,\gamma_2}) = (F_{\gamma_1}(x_{\gamma_1}), F_{\gamma_2}(x_{\gamma_2}))$ була повною, достатньо, щоб її МДР X_{γ_1,γ_2} містило такі й тільки такі підграфи $x_{\gamma_1,\gamma_2} = (x_{\gamma_1}, x_{\gamma_2})$, для яких виконується рівність матриць сміжності підграфів x_{γ_1}и x_{γ_2}: $X_{\gamma_1,\gamma_2} = \{x_{\gamma_1,\gamma_2} = (x_{\gamma_1}, x_{\gamma_2}), x_{\gamma_1} = x_{\gamma_2}\}$, тобто $X_{\gamma_1,\gamma_2} = X_{\gamma_1} \cap X_{\gamma_2}$. Потужність МДР, ПМ й ПМА двокритеріальної задачі Z_{γ_1,γ_2} дорівнює $|X_{\gamma_1} \cap X_{\gamma_2}|$.

Теорема 2. Для того, щоб двокритеріальна задача Z_{γ_1,γ_2} с ВЦФ $F_{\gamma_1,\gamma_2}(x_{\gamma_1,\gamma_2}) = (F_{\gamma_1}(x_{\gamma_1}), F_{\gamma_2}(x_{\gamma_2}))$ була квазіповною, достатньо, щоб її МДР X_{γ_1,γ_2} містило такі й тільки такі підграфи x_{γ_1,γ_2}, для яких виконується відношення включення для підграфів x_{γ_1} та x_{γ_2}: $X_{\gamma_1,\gamma_2} = \{x_{\gamma_1,\gamma_2}: x_{\gamma_1,\gamma_2} = (x_{\gamma_1}, x_{\gamma_2}), x_{\gamma_1} \subset x_{\gamma_2}$ або $x_{\gamma_1} \supset x_{\gamma_2}\}$, де x_{γ_1}- допустимий розв'язок задачі Z_{γ_1}, x_{γ_2}- допустимий розв'язок задачі Z_{γ_2}.

Модельними задачами для дослідження властивості повноти слугували задачі з ваговими критеріями такі як «кістяк та гамільтоновий ланцюг» та інші. Для дослідження властивості квазіповноти слугували задачі з ваговими критеріями, в яких не співпадали кількості ребер x_{γ_1}та x_{γ_2}, наприклад «гамільтонів цикл та гамільтонів ланцюг». В [3] було доведено властивість квазіповноти для задачі «кістяк та екстремальне ребро».

Таким чином, відношення між розв'язками x_{γ_1}та x_{γ_2} за відповідними критеріями $F_{\gamma_1}(x_{\gamma_1}), F_{\gamma_2}(x_{\gamma_2})$ є визначальним для властивості повноти та квазіповноти для двокритеріальних задач дискретної оптимізації.

Можна зробити припущення про існування аналогічних закономірностей для задач з вектором критеріїв більшого порядку ніж 2. Розглянемо клас трикритеріальних задач з набором критеріїв, що можуть бути ваговими, або екстремального ребра. Аналогічно [1] введемо символ задач дискретної оптимізації задачі $Z_{\gamma_1,\gamma_2,\gamma_3}$, де трикритеріальна

задача $Z_{\gamma_1,\gamma_2,\gamma_3}$ на тризваженому n −вершинному m −реберному графі з мінімізуємими критеріями $F_{\gamma_i}(x) \to min,\ i = \overline{1,3}$ векторної цільової функції $F_{\gamma_1,\gamma_2,\gamma_3}(x_{\gamma_1,\gamma_2,\gamma_3})$. Індекси $\gamma_i,\ i = \overline{1,3}$ є індивідуальними номерами критеріїв, та їхній порядок $F_{\gamma_i}(x)$, визначає порядок розв'язання задач Z_{γ_i} на n -вершинному графі $G = (V, E),\ |E| = m$, кожному ребру $e_t \in E,\ t = \overline{1,m}$ якого надано трійка значень ваги $\omega_i(e_t)$, $i = \overline{1,3},\ t = \overline{1,m}$.

Перший варіант набору критеріїв буде повторювати ВЦФ $F_{\gamma_1,\gamma_2}(x_{\gamma_1,\gamma_2})$ будь- якої повної задачі з [1], який доповнюється таким критерієм, щоб виконувалася умова рівності матриць сміжності підграфів $x_{\gamma_1},\ x_{\gamma_2}, x_{\gamma_3}$, тобто розв'язком якої буде підграф $x_{\gamma_1,\gamma_2,\gamma_3}$: $X_{\gamma_1,\gamma_2,\gamma_3} = \{x_{\gamma_1,\gamma_2,\gamma_3} = (x_{\gamma_1}, x_{\gamma_2}, x_{\gamma_3}), x_{\gamma_1} = x_{\gamma_2} = x_{\gamma_3}\}$. Для оцінки потужності МДР, ПМ й ПМА трикритериальної задачі $Z_{\gamma_1,\gamma_2,\gamma_3}$ застосуємо алгоритм, запропонований в [1] з застосуванням певного способу визначення ваг ребер. Ребра $e \in E$ перенумеруємо числами $t = t(e) = \overline{1,m},\quad m = |E|$. Для каждого ребра $e_t \in E$ визначимо ваги:

$$\omega_1(t) = 2^t + t,\ \omega_2(t) = 2^{m+1} - \omega_1(t) - \omega_3(t),\ \omega_3(t) = t,$$

що дозволяє стверджувати, що для всіх допустимих розв'язків $x_{\gamma_i},\ i = \overline{1,3}$ равенство $|E_{\gamma_i}| = C_0(n) = C_0 = const$ для трикритериальної задачі $Z_{\gamma_1,\gamma_2,\gamma_3}$, тобто

$$F_{\gamma_1}(x) + F_{\gamma_2}(x) + F_{\gamma_3}(x) = 2^{m+1}C_0 \quad \forall x_{\gamma_i} \in X_{\gamma_i},\ i = \overline{1,3}.$$

Можно показати, що при такій нумерації ваговий критерій набуває однакових значень тільки при співпадінні множини ребер.

На першому етапі йде розв'язання однокритеріальної задачі за критерієм F_{γ_1}, далі на множині отриманих розв'язків X_{γ_1} розв'язується однокритеріальна задача за критерієм F_{γ_2} й отримаємо X_{γ_2}, далі виконаємо етап розв'язання за критерієм F_{γ_3}, що доводить справедливість

Теорема 3. Трикритеріальна задача $Z_{\gamma_1,\gamma_2,\gamma_3}$ на тризваженому графі з ВЦФ $F_{\gamma_1,\gamma_2,\gamma_3}(x_{\gamma_1,\gamma_2,\gamma_3})$, що побудована додаванням третього критерію до ВЦФ $F_{\gamma_1,\gamma_2}(x_{\gamma_1,\gamma_2})$ повної двокритеріальної задачі, не змінює властивості повноти при умові рівності матриць сміжності підграфів x_{γ_1}, $x_{\gamma_2}, x_{\gamma_2}$, розв'язком якої буде підграф $x_{\gamma_1,\gamma_2,\gamma_3}$: $X_{\gamma_1,\gamma_2,\gamma_3} = \{x_{\gamma_1,\gamma_2,\gamma_3} = (x_{\gamma_1}, x_{\gamma_2}, x_{\gamma_3}), x_{\gamma_1} = x_{\gamma_2} = x_{\gamma_3}\}$, та потужності МДР, ПМ й ПМА

дорівнюють $X^0 = \tilde{X} = X = \left|X_{\gamma_1} \cap X_{\gamma_2} \cap X_{\gamma_3}\right|$. Справедливо при довільній нумерації критеріїв.

Другий варіант набору критеріїв буде повторювати ВЦФ $F_{\gamma_1,\gamma_2}\left(x_{\gamma_1,\gamma_2}\right)$ будь- якої повної задачі з [1] та доповнюватися таким критерієм, щоб виконувалася умова включення $x_{\gamma_3} \subset x_{\gamma_1} = x_{\gamma_2}$, тобто розв'язком якої буде підграф $x_{\gamma_1,\gamma_2,\gamma_3}$: $X_{\gamma_1,\gamma_2,\gamma_3} = \{x_{\gamma_1,\gamma_2,\gamma_3} = (x_{\gamma_1}, x_{\gamma_2}, x_{\gamma_3}), x_{\gamma_3} \subset x_{\gamma_1} = x_{\gamma_2}\}$. Застосування запропонованого алгоритму показує, що на третьому етапі певні x_{γ_2} будуть мати однакові значення за третім критерієм, тобто справедливим є твердження, що потужності МДР, ПМ й ПМА відповідають співвідношенню $X^0 = \tilde{X} < X$.

Є важливим, що зміна послідовності критеріїв не вплине на структуру та потужності МДР, ПМ й ПМА. Справедливим є твердження

Теорема 4. Трикритеріальна задача $Z_{\gamma_1,\gamma_2,\gamma_3}$ на тризваженому графі з ВЦФ $F_{\gamma_1,\gamma_2,\gamma_3}\left(x_{\gamma_1,\gamma_2,\gamma_3}\right)$, що побудована додаванням третього критерію до ВЦФ $F_{\gamma_1,\gamma_2}\left(x_{\gamma_1,\gamma_2}\right)$ повної двокритеріальної задачі, змінює властивість повноти на квазіповноту при умові включення $x_{\gamma_3} \subset x_{\gamma_1} = x_{\gamma_2}$, тобто розв'язком буде підграф $x_{\gamma_1,\gamma_2,\gamma_3}$: $X_{\gamma_1,\gamma_2,\gamma_3} = \{x_{\gamma_1,\gamma_2,\gamma_3} = (x_{\gamma_1}, x_{\gamma_2}, x_{\gamma_3}), x_{\gamma_3} \subset x_{\gamma_1} = x_{\gamma_2}\}$. Потужності МДР, ПМ й ПМА відповідають співвідношенню $X^0 = \tilde{X} < X$, при цьому $X^0 = \tilde{X} = X_{\gamma_3}$. Справедливо при довільній нумерації критеріїв.

Побудуємо аналогічні експерименти для квазіповних задач з [1]. Третій варіант набору критеріїв сформуємо з критеріїв, для яких виконується умова включення $x_{\gamma_1} \subset x_{\gamma_2}$ або $x_{\gamma_1} \supset x_{\gamma_2}$ з додаванням критерію, що забезпечує відношення еквівалентності між деякими розв'язками за вже існуючими критеріями, тобто $x_{\gamma_1} = x_{\gamma_3}$ або $x_{\gamma_2} = x_{\gamma_3}$ при $x_{\gamma_1} \subset x_{\gamma_2}$ або $x_{\gamma_1} \supset x_{\gamma_2}$.

Четвертий варіант набору критеріїв сформуємо з критеріїв, для яких виконується умова включення $x_{\gamma_1} \subset x_{\gamma_2}$ або $x_{\gamma_1} \supset x_{\gamma_2}$ з додаванням критерію, що забезпечує аналогічну властивість. Проводячи аналіз, аналогічний попереднім варіантам наборів критеріїв, можемо стверджувати

Теорема 5. Трикритеріальна задача $Z_{\gamma_1,\gamma_2,\gamma_3}$ на тризваженому графі з ВЦФ $F_{\gamma_1,\gamma_2,\gamma_3}\left(x_{\gamma_1,\gamma_2,\gamma_3}\right)$, що побудована додаванням третього критерію до ВЦФ $F_{\gamma_1,\gamma_2}\left(x_{\gamma_1,\gamma_2}\right)$ квазіповної двокритеріальної задачі, не змінює властивість квазіповноти як при умові відношення включення, так й відношення еквівалентності між розв'язками відповідних однокритеріальних задач. Справедливо при довільній нумерації критеріїв.

Справедливим є й узагальнення отриманих результатів на довільну кількість критеріїв та їхню послідовність.

Теорема 6. Багатокритеріальна задача $Z_{\gamma_1,\gamma_2,\dots,\gamma_N}$ на N-зваженому графі з ВЦФ $F_{\gamma_1,\gamma_2,\dots,\gamma_N}(x_{\gamma_1,\gamma_2,\dots,\gamma_N})$, є повною при умові виконання відношення еквівалентності між розв'язками відповідних однокритеріальних задач, тобто МДР, ПМ й ПМА дорівнюють $X^0 = \tilde{X} = X = \left|X_{\gamma_1} \cap X_{\gamma_2} \cap \dots \cap X_{\gamma_N}\right|$, та квазіповною при умові виконання відношення включення між розв'язками відповідних однокритеріальних задач, тобто МДР, ПМ й ПМА задовольняють співвідношення $X^0 = \tilde{X} < X$, при цьому $X^0 = \tilde{X} = \min_i \left|X_{\gamma_i}\right|$. Справедливо при довільній нумерації критеріїв.

Наприклад, багатокритеріальна задача $Z_{\gamma_1,\gamma_2,\dots,\gamma_N}$, в наборі критеріїв якої будуть критерії «кістяк екстремальної ваги» та «екстремальне ребро» на n -вершинному графі $G = (V, E)$, $|E| = m$ з ВЦФ $F_{\gamma_1,\gamma_2,\dots,\gamma_N}(x_{\gamma_1,\gamma_2,\dots,\gamma_N})$, де

$$F_{\gamma_k} = \omega_{\gamma k}(x_{\gamma_1,\gamma_2,\dots,\gamma_N}) = \omega_{\gamma k}(x_{\gamma k}) = \sum_{e\in E_{x\gamma k}} \omega_{\gamma k}(e) \to extr,$$

$$F_{\gamma_j} = \underset{e\in E_{\gamma_j}}{extr}\,\omega_{\gamma_j}(e) \to extr,\ 0 < j \le N, 0 < k \le N,$$

є квазіповною, тобто $X^0 = \tilde{X} < X$. При цьому $X^0 = \tilde{X} < m - (n-2)$. МДР не перевищує кількості кістяків в повному n -вершинному графі $|X| \le n^{n-2}$.

Література

5. Perepelisa, V. On complete and quasi-complete two-criteria optimization problems on graphs / V. Perepelisa, E. Tereschenko // Springer Science+Business Media, 2018. P. 385-390 https://doi.org/10.1007/s10559-018-0040-1

6. Перепелица В.А. Многокритериальные модели и методы для задач оптимизации на графах. — Saarbrücken :LAP LAMBERT Academic Publishing, 2013. – 337с.

7. Перепелица В.А. Квазиполнота класса задач на графах «вес-минимаксное ребро» / В.А.Перепелица, Э.В.Терещенко, А.Е.Рябенко // Питання прикладної математики та математичного моделювання – 2018. – С.139-147. https://doi: 10.15421/321814

ДОСЛІДЖЕННЯ ПІДХОДІВ ГЛИБИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ ТА ОБРОБКИ ПРИРОДНОЇ МОВИ: КОНЦЕПТ ДЛЯ СИСТЕМИ Є-ВОРОГ

Владислав Терницький, Дмитро Широкорад
Національний університет «Запорізька політехніка»

EXPLORING DEEP LEARNING APPROACHES FOR IMAGE RECOGNITION AND NATURAL LANGUAGE PROCESSING: THE CONCEPT FOR E-ENEMY

Abstract. The large-scale invasion of Ukraine by Russia has highlighted the need for innovative approaches to gather and process information in war zones. With the rapid growth of artificial intelligence (AI) models and natural language processing (NLP) techniques, the concept of leveraging deep learning approaches for improving the accuracy and efficiency of intelligence gathering becomes increasingly relevant. This paper explores the concept for e-Enemy, a system developed by the Ukrainian government to gather information on enemy movements through user-submitted applications. The main goal of the concept is to evaluate the accuracy of images and text submitted in the applications to the e-Enemy, the accuracy of text-image correspondence and ensure filtering spam applications. This research aims to contribute to the development of more accurate and efficient methods for extracting valuable intelligence from user-generated content within the context of the e-Enemy system.

Розпізнавання об'єктів відіграє ключову роль у концепті для системи єВорог, оскільки дозволяє виявляти та ідентифікувати ворожу техніку або живу силу на зображеннях, надісланих користувачами. Для вирішення цього завдання було обрано модель You Only Look Once (YOLO) [1] через її ефективність у виявленні та локалізації об'єктів. Вибір даної моделі мотивований її здатністю досягати балансу між точністю та швидкістю обробки. Як наскрізна архітектура згорткової нейронної мережі (CNN) [2], YOLO обробляє все зображення за один прохід і одночасно прогнозує обмежувальні рамки та ймовірності класів для декількох об'єктів. Модель використовує уніфіковану архітектуру, яка розділяє вхідне зображення на сітку і генерує прогнози на основі ознак, виділених у кожній комірці сітки. Вона складається з основної мережі CNN, як правило, варіанту Darknet, за якою слідують кілька згорткових шарів, що прогнозують координати обмежувальних рамок і пов'язані з ними ймовірності класів. Система моделює виявлення як регресійну задачу. Вона розбиває зображення на сітку $S \times S$ і для кожної

комірки прогнозує B клітинок, довірчу ймовірність для цих клітинок, та ймовірності класів C. Ці прогнози кодуються у вигляді $S \times S \times (B * 5 + C)$ тензор [1].

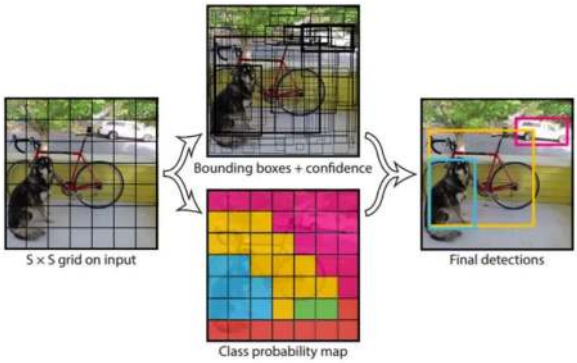


Рис. 1 - Модель YOLO [1]

YOLO використовує єдину мережеву оцінку для прогнозування граничної області, що робить її значно швидшою порівняно з іншими моделями виявлення об'єктів. Крім того, YOLO відома своєю здатністю обробляти невеликі об'єкти завдяки притаманній їй просторовій чутливості та контекстному розумінню.

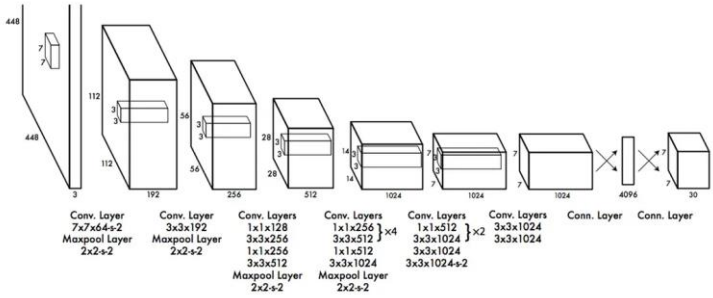


Рис.1 - Архітектура моделі YOLO [1]

Мережа виявлення має 24 згорнуті шари, за якими слідують 2 повністю з'єднані шари. Згорткові шари 1 × 1, що чергуються, зменшують простір ознак від попередніх шарів.

Основні переваги YOLO:

- модель досягає вражаючої продуктивності в режимі реального часу, що дозволяє оперативно аналізувати зображення, надіслані користувачем. Це має вирішальне значення в контексті системи єВорог, де своєчасне і точне виявлення ворожої техніки або живої сили є життєво важливим для прийняття стратегічних рішень і безпеки постраждалого населення;
- уніфікована архітектура YOLO дозволяє ефективно фіксувати контекстні зв'язки між об'єктами на зображенні, що полегшує точну ідентифікацію і локалізацію декількох об'єктів одночасно. Ця можливість є особливо корисною у сценаріях, коли необхідно виявити декілька одиниць ворожої техніки або живої сили на зображенні.

Проте важливо зазначити, що модель YOLO може зіткнутися з проблемами при роботі з сильно закритими або перекритими об'єктами, оскільки вона покладається на комірки сітки для генерування прогнозів. Крім того, через однопрохідну природу YOLO, вона може мати обмеження в точному виявленні об'єктів різного масштабу на зображенні.

У контексті обробки тексту, поданого разом із зображеннями, та перевірки відповідності між текстом і зображенням, було вирішено застосувати модель BERT (Bidirectional Encoder Representations from Transformers) [3] для задач обробки природної мови (NLP). Модель BERT використовує transformer архітектуру, яка складається з декількох шарів з механізмами самоуваги. Ця архітектура дозволяє моделі фіксувати залежності та зв'язки між словами в реченні, тим самим забезпечуючи глибше розуміння тексту. Двонаправлений характер BERT та глибоке розуміння контексту роблять його ефективним для розпізнавання тексту в системі єВорог. Використовуючи BERT, ми зможемо вловити складні значення та нюанси, присутні в текстових описах, пов'язаних з надісланими зображеннями.

У нашому конкретному випадку використання модель BERT буде відповідати за розпізнавання та розуміння текстової інформації, наданої користувачами у поєднанні з надісланими зображеннями. Використовуючи попередньо навчену модель, основною ціллю буде отримувати з тексту змістовне представлення інформації, яке можна використати для оцінки відповідності між текстовими описами та візуальним контентом, зображеним на відповідних фотографіях.

Однією з помітних переваг моделі BERT є її здатність генерувати та розпізнавати зв'язну та контекстуально багату текстову інформацію. Ця здатність необхідна для оцінки того, наскільки точно текст відображає візуальний контент - ворожу техніку або живу силу. Використовуючи здатність BERT фіксувати семантичні зв'язки і контекстну інформацію,

ми можемо проаналізувати відповідність між текстом і зображенням і визначити рівень відповідності або невідповідності між цими двома модальностями. Однак доцільно звернути увагу, що для забезпечення високої точності розпізнавання і розуміння тексту може знадобитися додаткове донавчання моделі використовуючи розмічений набір даних з текстовими описами.

Загальний алгоритм оцінки відповідності між текстом та візуальним контентом буде виглядати наступним чином:

1. Виділення ознак (features). Отримання ознак зображень з вихідних обмежувальних рамок та текстових ознак з текстових описів.

2. Порівняння та відповідність. Обчислення показника подібності або метрики відстані між елементами зображення і тексту, наприклад, косинусової подібності [4] або евклідової відстані [5]. Цей показник вказує на ступінь відповідності між текстовим описом і виявленим візуальним контентом, що дає змогу кількісно оцінити схожість або відмінність між елементами.

3. Застосування порогу до оцінки подібності, щоб визначити, чи достатньою мірою текст і зображення відповідають один одному. Щоб визначити цей поріг, ми можемо використовувати валідаційний набір даних з анотованими парами текст-зображення. Порівнюючи оцінки подібності, отримані за допомогою алгоритму, з базовими анотаціями, ми можемо вибрати відповідне порогове значення, яке відповідає бажаному рівню точності їх відповідності.

Реалізувавши цей алгоритм, ми можемо ефективно оцінити точність відповідності текст-зображення в системі єВорог, збільшуючи загальну продуктивність роботи системи та забезпечуючи дієву фільтрацію спам заявок, а також полегшити роботу аналітиків системи.

Література

1. Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, University of Washington, Allen Institute for AI, Facebook AI Research, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, http://pjreddie.com/yolo/

2. Keiron O'Shea, Ryan Nash (2015). An Introduction to Convolutional Neural Networks. https://arxiv.org/abs/1511.08458

3. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019. https://arxiv.org/abs/1810.04805

4. Faisal Rahutomo, Teruaki Kitasuka, and Masayoshi Aritsugi, Kumamoto University. Semantic Cosine Similarity, 2012.

https://www.researchgate.net/publication/262525676_Semantic_Cosine_Similarity

5. Per-Erik Danielsson, Euclidean distance mapping, Computer Graphics and Image Processing, Volume 14, Issue 3, 1980, Pages 227-248, ISSN 0146-664X, https://doi.org/10.1016/0146-664X(80)90054-4.

ВЛАСТИВОСТІ КОМБІНАТОРНИХ КОНФІГУРАЦІЙ ТА ЇХНЄ ВИКОРИСТАННЯ В КОМБІНАТОРНІЙ ОПТИМІЗАЦІЇ

Надія Тимофієва

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

PROPERTIES OF COMBINATORY CONFIGURATIONS AND THEIR USE IN COMBINATORY OPTIMIZATION

Abstract. The properties of combinatorial configurations are considered, the use of which allows to explain some phenomena that exist in combinatorics and combinatorial optimization. This is the similarity of problems of combinatorics and combinatorial optimization, which explains the universality of algorithms and methods for solving the specified problems. The peculiarity of the structure of combinatorial sets, which are the argument of the objective function, explains the situation of uncertainty in combinatorial optimization problems.

Вступ. В статті описано властивості комбінаторних конфігурацій, використання яких дозволяє пояснити деякі явища, що наявні в комбінаториці та комбінаторній оптимізації. Це – подібність комбінаторних конфігурацій та задач комбінаторної оптимізації, особливість структури комбінаторних множин, які є аргументом цільової функції, завдяки якій в комбінаторній оптимізації виникає ситуація невизначеності. Виявлено задачі, які подібні за аргументом цільової функції, а також встановлена подібність комбінаторних конфігурацій, множини яких генеруються одним і тим же алгоритмом або його модифікацією. Цією властивість пояснюється універсальність методів та алгоритмів в комбінаторній оптимізації.

Подібність комбінаторних конфігурацій. Під комбінаторною конфігурацією розуміємо будь-яку сукупність елементів, яка утворюється з усіх або з деяких елементів заданої базової множини $A = \{a_1, ..., a_n\}$. Позначимо її упорядкованою множиною

$w^k = (w_1^k, ..., w_\eta^k)$, де $\eta \in \{1, \dots n\}$ – кількість елементів у w^k (в подальшому η позначатимемо і як η^k), $W = \{w^k\}_1^q$ – множина комбінаторних конфігурацій. Верхній індекс k ($k \in \{1, \dots q\}$) у w^k позначає порядковий номер w^k у W, q – кількість w^k у W.

Рекурентним комбінаторним оператором назвемо сукупність правил, за допомогою яких з елементів базової множини A утворюється комбінаторна конфігурація w^k. Різноманітні типи комбінаторних конфігурацій утворюються за допомогою трьох рекурентних комбінаторних операторів: вибирання, транспозиція, арифметичний [1].

Означення 1. Дві нетотожні комбінаторні конфігурації $w^k = (w_1^k, \dots, w_{\eta^k}^k)$ та $w^i = (w_1^i, \dots, w_{\eta^i}^i)$ назвемо ізоморфними, якщо $\eta^k = \eta^i$.

Означення 2. Підмножину $W_{\eta^k} \subset W$ назвемо підмножиною ізоморфних комбінаторних конфігурацій, якщо її елементи – ізоморфні комбінаторні конфігурації.

Означення 3. Назвемо подібними задачі з комбінаторики або задачі комбінаторної оптимізації різних класів, які розв'язуються за однією і тією ж обчислювальною схемою або модифікацією одного і того ж алгоритму.

Подібність комбінаторних конфігурацій встановлюється за способом їхнього утворення. Сполучення як з повтореннями так і без повторення утворюються єдиною операцією – вибиранням. Перестановки утворюються або транспозицією або вибиранням. Розбиття числа утворюються однією операцією – арифметичною. Розбиття n-елементної множини на підмножини утворюються двома рекурентними комбінаторними операторами: або арифметичним або транспозицією. Розміщення як з повтореннями так і без повторення утворюються двома операціями: або вибиранням або транспозицією. Бінарні послідовності можуть утворюватися двома операціями: арифметичною або операцією вибирання. До того ж, кількість бінарних послідовностей у їхній множині дорівнює 2^n, а кількість сполучень без повторення – відповідно $2^n - 1$. Генерування розбиття n-елементної множини на підмножини проводять алгоритмом розбиття числа і генеруванням перестановок. Упорядкування розміщення як без повторення, так і з повтореннями проводять алгоритмами генерування сполучень і перестановок. Формування бінарних послідовностей з

використанням рекурентного комбінаторного оператора вибирання проводять за тими ж правилами, що і утворення сполучень без повторення.

Отже, за способом утворення подібні такі комбінаторні конфігурації: бінарні послідовності і сполучення без повторення; розбиття n-елементної множини на підмножини і розбиття натурального числа та перестановки; розміщення без повторення (з повтореннями) і сполучення без повторень (з повтореннями) та перестановки.

Подібність задач комбінаторної оптимізації. Наведемо математичну постановку задачі комбінаторної оптимізації.

Нехай задано одну A або кілька множин, наприклад A та B [1]. Вагою назвемо величину, яка встановлює залежність, що існує між елементами $a_s \in A$ і $b_l \in B$ або між елементами однієї і тієї ж множини, $s \in \{1,...,n\}$, $l \in \{1,..., \tilde{n}\}$, n – кількість елементів множини A, $\tilde{n}$ – кількість елементів множини B. Покладемо, що $n = \tilde{n}$. Значення ваг між елементами множин A і B задамо однією або двома симетричними або несиметричними матрицями C і $Q(w^k)$, де $Q(w^k)$ – комбінаторна матриця, $w^k \in W$ – перестановка. Якщо значення ваг описуються однією матрицею C, то для визначення наявності зв'язків між елементами заданих множин для k-го варіанту розв'язку задачі уведемо симетричну або несиметричну (0,1)-матрицю $Q(w^k)$. Елемент $g_{sl}(w^k) = 1$, якщо між $a_s \in A$ та $b_l \in B$ або $a_s \in A$ і $a_l \in A$ для варіанту w^k існує зв'язок, і $g_{sl}(w^k) = 0$ в іншому разі; $g_{sl}(w^k) \in Q(w^k)$.

Подамо елементи h наддіагоналей симетричної комбінаторної матриці $Q(w^k)$ комбінаторною функцією $\beta(f(j), w^k)|_1^m = (\beta_1(f(1), w^k),... \ ..., \beta_m(f(m), w^k))$, а елементи h наддіагоналей симетричної матриці C – функцією натурального аргументу $\varphi(j)|_1^m = (\varphi(1),...,\varphi(m))$, де $m = \dfrac{n(n-1)}{2}$ – кількість елементів h наддіагоналей матриць C і $Q(w^k)$, $h = \overline{1, n-1}$. Якщо матриці $Q(w^k)$ і C – несиметричні, то $\beta(f(j), w^k)|_1^m$ і $\varphi(j)|_1^m$ містять усі їхні елементи, а $m=n^2$ (або $m = n\,\tilde{n}$). Функція цілі для цих задач набуває вигляду

$$F(w^k) = \sum_{j=1}^{m} \beta_j (f(j), w^k) \varphi(j), \quad (1)$$

До задачі комбінаторної оптимізації, аргументом цільової функції в якій є перестановка, зводяться задача комівояжера, задача про призначення, задача розміщення одногабаритних об'єктів на поверхні тощо. Цільова функція для них моделюється виразом (1), за яким проводиться оцінка результату. Завдяки цьому задачі комбінаторної оптимізації, аргументом цільової функції в яких є перестановка і на підмножині ізоморфних комбінаторних конфігурацій (задача кластеризації) розв'язуються універсальними методами, зокрема методом структурно-алфавітного пошуку за однією і тією ж схемою. В задачі кластеризації на деяких ізоморфних підмножинах цільова функція змінюється так як і в задачі комівояжера.

Прикладні задачі складні за своєю природою і основна задача, як правило, розділяється на підзадачі, а цільова функція, за якою оцінюється оптимальний розв'язок, залежить від кількох змінних, якими є комбінаторні конфігурації різних типів. Якщо побудувати математичні постановки задач розпізнавання мовленнєвих сигналів та клінічної діагностики з використанням теорії комбінаторної оптимізації, то можна побачити, що вони розділяються на три підзадачі:

а) структуризація бібліотеки еталонів;

б) пошук у бібліотеці еталонної інформації;

в) задача порівняння еталонної і вхідної інформації.

Для обох класів задач аргументом цільової функції в першій підзадачі є розбиття n-елементної множини на підмножини, в другій підзадачі – розміщення без повторень, а в третій – сполучення без повторень. Тобто, ці задачі подібні за аргументом цільової функції.

Невизначеність в задачах комбінаторної оптимізації, що пов'язана зі структурою множини комбінаторних конфігурацій. Ситуація невизначеності в задачах цього класу виникає внаслідок особливої структури множини комбінаторних конфігурацій, що є аргументом цільової функції.

Закономірність зміни значень цільової функції в задачах комбінаторної оптимізації залежить від упорядкування комбінаторних конфігурацій (аргументу) $w \in W$. На підмножині ізоморфних комбінаторних конфігурацій W_η цільова функція (1) змінюється так, як і на множині перестановок. Можна довести, що на множині перестановок і на підмножині ізоморфних комбінаторних конфігурацій при використанні цільової функції (1) ситуація невизначеності зводиться до

мінімуму. Але на множині W, яка складається з підмножин $W_\eta \subset W$, закономірність зміни значень функції (1) однакова незалежно від вхідних даних, а результат розв'язку задачі – неоднозначний.

Висновок. Отже дослідження та виявлення властивостей комбінаторних конфігурацій дозволяє встановити подібність задач комбінаторики та комбінаторної оптимізації. Відповідно можна пояснити природу універсальності алгоритмів та методів їхнього розв'язання. Особливість структури комбінаторних множин, які є аргументом цільової функції, пояснює ситуацію невизначеності в задачах комбінаторної оптимізації (кластеризації, задача про ранець тощо).

Література

1. Тимофієва Н.К. Теоретико-числові методи розв'язання задач комбінаторної оптимізації. Автореф. дис... докт. техн. наук / Ін-т кібернетики ім. В.М. Глушкова НАН України, Київ. 2007. 32 с.

2. Тимофієва Н.К. Про подібність задач комбінаторної оптимізації та універсальність алгоритмів. *Системні дослідження та інформаційні технології.* 2013. № 4. С. 27–37.

ПЕРЕРИВАННЯ В ЗАДАЧАХ УПОРЯДКУВАННЯ ВЕРШИН ГРАЦІОЗНИХ ДЕРЕВ

Валентина Турчина, Євген Коваленко
Дніпровський національний університет імені Олеся Гончара

INTERRUPTIONS IN THE GRACEFUL TREES' VERTICES PARALLEL SEQUENCING PROBLEMS

Abstract. This research aimed to investigate the parallel sequencing problems for cases when a digraph has a structure of a tree which allows graceful labeling. The main attention was paid to the subclass called stars. The optimal solutions for cases of allowed and denied interruptions were compared. Directions for further research on this problem were stated.

Класичні постановки задач упорядкування вершин орграфів $G = (V,U)$ сформульовані в [1].

Наступні дві основні задачі постійно привертають до себе увагу:

- при заданій ширині $h(S)$ побудувати упорядкування S мінімальної довжини;

- при заданій довжині $l(S)$ побудувати упорядкування S мінімальної ширини.

В даній роботі розглядається перша з них для спеціальних підкласів графів.

Наведемо деякі відомі означення [2].

Означення 1. Індукована мітка ребра — це абсолютна величина різниці міток вершин, інцидентних до цього ребра.

Означення 2. Граціозною називають таку розмітку вершин неорієнтованого графа G із n ребрами, при якій кожній вершині ставиться у взаємно однозначну відповідність мітка із множини $\{0, 1, \ldots n\}$, причому всі індуковані мітки ребер є різними.

Означення 3. Граф називають граціозним, якщо він допускає граціозну розмітку.

Для першої з вказаних задач розглядається випадок, коли заданий орграф має структуру граціозного дерева.

Оскільки поняття граціозності відноситься до неорієнтованих дерев, а в подальшому будуть розглядатися орієнтовані, то можливі різні варіанти заміни ребер на дуги. Окрім того будемо вважати, що ці графи зважені та їх вершини мають одиничну вагу. Одним з найперших класів дерев, для яких було доведено можливість граціозної розмітки, є зірки.

Відомо, що зіркою називається дерево, що має одну вершину степеня, більшого за 1, а степінь решти дорівнює 1.

При детальнішому розгляді орієнтованих зірок можна виділити такі три випадки на основі того, яким саме чином ребра замінюються дугами.

1. Всі дуги виходять із центральної вершини зірки (рис.1 *а*).
2. Всі дуги входять в центральну вершину зірки (рис.1 *б*).
3. Частина дуг входять в центр зірки, решта — виходять (рис.1 *в*).

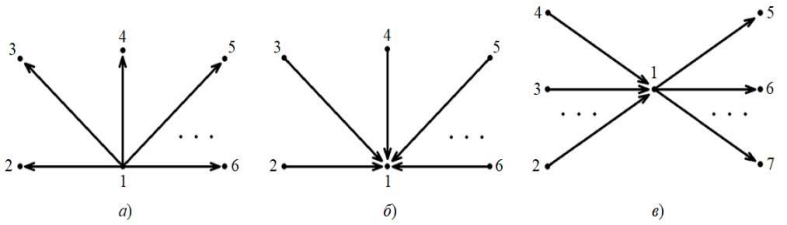


Рис. 1 — Види орієнтованих зірок

При розв’язанні задач паралельного упорядкування для випадку 1 спочатку центральна вершина зірки ставиться на перше місце в упорядкування. Після цього, оскільки решта вершин зірки не пов’язані дугами, вони вноситься в упорядкування так, ніби є ізольованими. Для

випадку 2 аналогічним чином спершу всі вершини, крім центральної, розміщуються в упорядкуванні як ізольовані, після чого заноситься центральна. Випадок 3 є комбінацією попередніх двох. Та частина вершин, із яких дуги прямують до центру зірки вносяться до упорядкування першими, як ізольовані, потім центр зірки, після чого — решта вершин. Задачі упорядкування для графів з ізольованими вершинами детально розглядалися в [3].

Нехай скінченна множина робіт, на порядок виконання яких накладаються умови слідування, моделюється деревом, що є орієнтованою зіркою. З'ясуємо вплив переривань при виконанні робіт на оптимальність розв'язку в відповідних задачах упорядкування для цих випадків та оцінимо цей вплив за формулою, запропонованою в [3]:

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

де l^{*}, $l_{П}^{*}$ — довжина відповідно оптимального упорядкування без переривань та з перериваннями.

При порівнянні виявляється, що оптимальне упорядкування для вхідних та вихідних зірок із кількістю вершин (p+1) відрізняється від випадку графа з p ізольованими вершинами лише тим, що містить ще одну вершину на першій (для вихідних зірок) чи останній (для вхідних) позиції. Це також означає, що довжина оптимального упорядкування буде більшою на величину, що рівна вазі центральної вершини, ніж для відповідного графа з ізольованими вершинами незалежно від ширини упорядкування h.

Тоді для випадку одиничної ваги виграш становить

$$W = \left(1 - \frac{\frac{p}{h} + 1}{\left\lceil \frac{p}{h} \right\rceil + 1}\right) \cdot 100\%.$$

Для $h = p - 1$ отримуємо

$$W = \left(1 - \frac{\frac{1}{p-1} + 2}{3}\right) \cdot 100\%.$$

При $p >> 1$

$$\frac{1}{p-1} \to 0 \Rightarrow W \to 33,(3)\%.$$

Отже, для достатньо великої кількості вершин у вхідній чи вихідній зірці дозвіл переривань даватиме істотний виграш при $h = p - 1$. Тепер узагальнимо міркування, припускаючи, що $2 \le h \le p - 1$ і покажемо залежність $W(h)$ для значення $p = 100$.

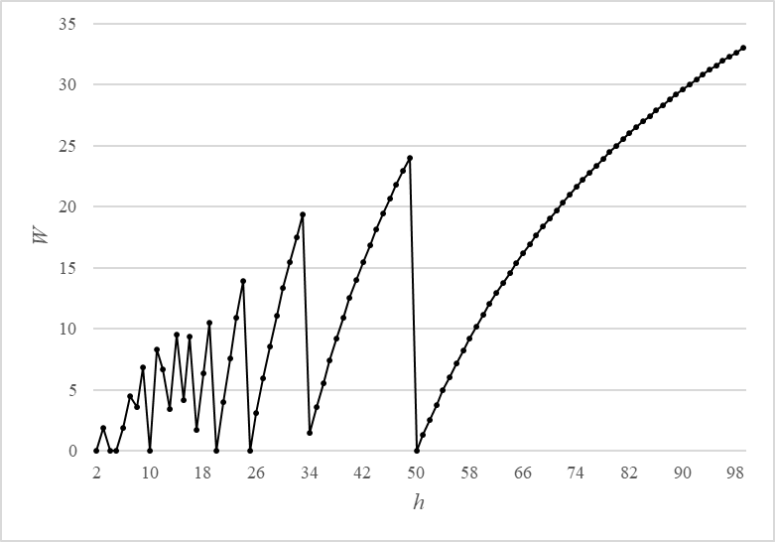


Рис. 2 — Залежність виграшу від ширини упорядкування при $p = 100$

Для комбінованого випадку, коли в зірці є p_1 дуг, що входять до центральної вершини, та p_2 таких, які виходять з неї ($p_1 + p_2 = p$), виграш знаходиться наступним чином:

$$W = \left(1 - \frac{\frac{p_1 + p_2}{h} + 1}{\left\lceil \frac{p_1}{h} \right\rceil + \left\lceil \frac{p_2}{h} \right\rceil + 1}\right) \cdot 100\% \text{ або } W = \left(1 - \frac{p_1 + p_2 + h}{h\left(\left\lceil \frac{p_1}{h} \right\rceil + \left\lceil \frac{p_2}{h} \right\rceil + 1\right)}\right) \cdot 100\%,$$

де $h \ge 2$, а відповідно $p_1, p_2 \ge 3$.

Встановимо залежності виграшу від різних значень параметрів p_1, p_2 та h. Для визначеності припустимо, що $p_1 \ge p_2$. Як видно із рис. 2, максимальне значення виграшу досягається при значеннях h близьких до кількості дуг. Через це вважатимемо, що $h = p_2 - 1$. Виграш при різних комбінаціях значень p_1, p_2 та h для $p = 100$ наведено на рис. 3.

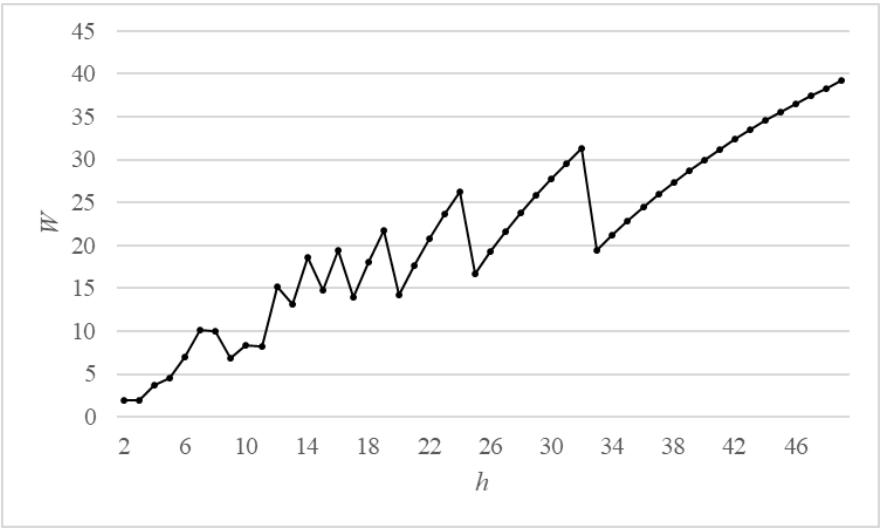


Рис. 3 — Залежність виграшу від параметрів при $p = 100$

Як видно з графіку, максимальне значення виграшу досягається при $p_1 = p_2$. З'ясуємо, до якого значення сходиться виграш. Введемо позначення $p' = p_1 = p_2$, і відповідно маємо $p' = h + 1$. Після підстановки отримуємо:

$$W = \left(1 - \frac{3h+2}{h\left(\left\lceil \frac{h+1}{h} \right\rceil + \left\lceil \frac{h+1}{h} \right\rceil + 1\right)}\right) \cdot 100\%,$$

або при спрощенні

$$W = \left(1 - \frac{3h+2}{5h}\right) \cdot 100\%.$$

Для значень $h >> 2$ (а отже і $p' >> 2$) виграш $W \to 40\%$.

Дозвіл переривань у задачах упорядкування, де орграф є зіркою, може істотно вплинути на оптимальний розв'язок, у тих випадках, коли вершини мають одиничну вагу, а задана ширина близька до кількості висячих вершин (крім комбінованого випадку). Подальшого вивчення потребують задачі, в яких:

- відповідний орграф має структуру іншого підкласу граціозних дерев;

- ваги вершин різні;
- ширина упорядкування задана довільно.

Література

1. Бурдюк В.Я., Турчина В.А. Алгоритмы параллельного упорядочения: учебное пособие. Д.: ДГУ, 1985. 84 с.
2. Петренюк Д.А. Граціозні дерева. Аналіз проблеми та перспективи. Управляющие системы и машины. 2016. С.16–25,33.
3. Коваленко Є.О., Турчина В.А. Аналіз структури графів в задачах паралельного упорядкування з перериваннями. Кропивницький: ПП «Ексклюзив-Систем». 2021. С.86-90.

ВИКОРИСТАННЯ ФРАГМЕНТАРНОЇ МОДЕЛІ ТА ЕВОЛЮЦІЙНОГО АЛГОРИТМУ ДЛЯ РОЗВ'ЯЗКУ ЗАДАЧ РОЗПОДІЛУ РЕСУРСІВ ЧАСУ

Михайло Хурдайло

Запорізький національний університет

USING FRAGMENTAL MODELS AND EVOLUTIONARY ALGORITHM FORSOLVING THE PROBLEMS OF TIME RESOURCE DISTRIBUTION

Abstract. Робота присвячена розробці гібридних алгоритмів на основі фрагментарної моделі для задачі розподілу ресурсів часу.

На сьогодні не відомі алгоритми з поліноміальною складністю для оптимального вирішення цих задач, то актуальним є дослідження наближених методів пошуку оптимального вирішення задач розподілу ресурсів часу та їх прикладного застосування для керування в економічних системах. Багатообіцяючим напрямком в цій царині є розробка алгоритмів, що засновані на методах метаевристики, а саме на використанні еволюційних алгоритмів [1].

Багато задач з теорії розкладів можуть бути зведені до пошуку оптимальної перестановки. Загальне формулювання задачі про оптимальну перестановку таке: на підмножині X множини всіх перестановок з n елементів задано функцію $F : X \to R^1$ Завдання знайти перестановку $x^* \in X$, для якої значення функції F мінімальне (максимальне).

Розглянемо одну з таких задач, яка часто виникає під час складання розкладу занять у навчальних закладах. Має вона таку інтерпретацію: для кожної групи студентів є навчальний план, у якому зазначено, які дисципліни будуть викладатися в семестрі. Кожна дисципліна *i* описується підмножиною номерів тижнів множини всіх тижнів у семестрі, де номер тижня *j* у підмножині *i* означає, що дисципліна *i* викладається на *j*-му тижні від початку читання цієї дисципліни. Передбачається, що кожен предмет кожного тижня викладається однакову кількість годин. Потрібно розмістити всі дисципліни з навчального плану так, щоб розподіл навантаження на студента був запланований рівномірно. Рівномірність навантаження може досягатися оптимізацією різних критеріїв. Один із них, мінімізація максимального тижневого навантаження за семестр, розглядається в цій постановці.

Можлива і більш загальна інтерпретація моделі. Кожен предмет може розглядатися як робота, що вимагає для свого виконання один ресурс. Використовується цей ресурс у кілька етапів, причому обсяг необхідного ресурсу на кожному етапі задовільний і не перевищує 1. Тоді змінними управління є час початку виконання даної роботи, а також машина, на яку ця робота призначена для виконання. Також інтерпретацією моделі може бути, наприклад, комп'ютерна мережа. Ресурсом тут є процесорний час. Тоді предмети інтерпретуються як завдання з виділеними і відомими наперед етапами різного споживання часу процесора.

Показано, що розглянута задача має фрагментарну структуру і може бути зведена до задачі безумовної оптимізації на множині перестановок.

Пропонується дослідити гіпотезу, що для пошуку субоптимальних розв’язків даної задачі можливо використовувати гібридний алгоритм, що поєднує у собі комбінацію однієї з відомих метаевристик еволюційних алгоритмів (генетичний алгоритм або мурашиний алгоритм) разом з фрагментарним алгоритмом [2].

За результатами дослідження, з'ясувалося, що задачі складання оптимального розкладу в загальному випадку є *NP*-складними [3].

Також не можна узагальнюючи стверджувати, що використання еволюційних алгоритмів може надавати розв’язання всіх типів задач зі складання чи оптимізації розкладів, проте є багато успішних прикладів застосування еволюційних алгоритмів для досягнення субоптимального результату [4].

Проте наразі задача складання розкладу не має такого алгоритму вирішення, що за будь-яких умов та обмежень надав би оптимальний результат, а отже й використання субоптимальних результатів, що

отримані за допомогою запропонованого гібридного алгоритму цілком можливо вважати придатними для використання.

Література

1. Chaudhry, I.A. Minimizing flow time for the worker assignment problem in identical parallel machine models using GA. Int J Adv Manuf Technol. 2010. Vol. 48, P.747–760 DOI: https://doi.org/10.1007/s00170-009-2323-1

2. Kozin I.V., Maksyshko N.K., Perepelitsa V.A. Fragmentary Structures in Discrete Optimization Problems. *Cybern Syst Anal.* 2017, Vol. 53, P.931–936. DOI: https://doi.org/10.1007/s10559-017-9995-6

3. Garey M.R., Johnson D.S. Computers and Intractability: A Guide to the Theory of NP-Completeness. 1st edition. Bell Laboratories Murray Hill: New Jersey, 1979. 340 p. URL: https://www.academia.edu/44422461/Computers_and_intractability_a_ guide_to_the_theory_of_np_completeness_garey_amp_johnson (Дата звернення 20.05.2023)

4. Томашевський В. М., Новіков Ю. Л., Камінська П. А. (2010). Складання розкладів занять у дистанційних системах навчання. *Вісник НТУУ «КПІ». Серія: Інформатика, управління та обчислювальна техніка.* 2010. № 52, С.118-130. URL: http://nbuv.gov.ua/UJRN/Vkpi_iuot_2010_52_20 (Дата звернення 20.05.2023).

КВАДРАТНА РІЗНИЦЕВА РОЗМІТКА РЕБЕРНОГО З'ЄДНАННЯ ПЕВНИХ ТИПІВ ГРАФІВ

Зоя Шерман
Міжнародний європейський університет

SQUARE DIFFERENCE LABELING FOR EDGE CONNECTION OF SOME GRAPHS

Abstract. In this paper some methods for constructing Square Difference Labeling of edge connection of m-copies of the cycle C_{2n+1} and cycle C_m are discussed.

В теорії розміток графів є велика кількість нерозв'язаних задач. Інтерес викликають певні задачі, в яких є необхідність дослідити існування квадратної різницевої розмітки графів, отримані ланцюговим з'єднанням та диз'юнктивним об'єднанням. Частково ця задача

розв'язана в роботі Geetha та Kalamani «*Square Difference Labeling for Some Graphs*» у 2019 році [1]. Автори показали, що існує квадратна різницева розмітка графа-зірки S_n з ланцюгом P_k , двох копій цикла C_n з ланцюгом P_k та обмеженого повного графу $B_{n,n}$. Щорічний огляд всіх існуючих розміток представлений в електронному журналі Гальяно «*A dynamic survey of graph labeling*» [2].

У даній робота розглядається квадратна різницева розмітка або *SDL* реберного з'єднання m копій циклу C_{2n+1} і цикла C_m.

Під графом розуміють скінчений неорієнтований граф без петель і кратних ребер. Нехай $G=(V, E)$ є граф з множиною вершин $V(G)$ і множиною ребер $E(G)$. Будемо вважати, що $|V(G)|=p$, $|E(G)|=q$.

Функцію f називають *квадратною різницевою розміткою* графа G з p вершинами, якщо f – бієкція з $V(G)$ на множину $\{0, 1, 2, \ldots, p-1\}$ та індукована нею реберна розмітка $f^*(u,v) = |[f(u)]^2-[f(v)]^2|$ є ін'єкцією з $E(G)$ в множину натуральних чисел. Граф, що допускає квадратну різницеву розмітку, називається квадратним різницевим графом або *SD* графом.

Під реберним з'єднанням m копій циклу C_{2n+1} і циклу C_m будемо розуміти граф, утворений з'єднанням довільної вершини копій циклу C_{2n+1} та кожної вершини циклу C_m ребром як показано на рис. 1.

Метою роботи є дослідження існування квадратної різницевої розмітки реберного з'єднання m копій циклу C_{2n+1} і циклу C_m.

Теорема. Реберне з'єднання m копій циклу C_{2n+1} і циклу C_m допускає квадратну різницеву розмітку для будь-яких натуральних чисел m, n де $m \geq 3$.

Доведення. Нехай $\{v_1^i, v_2^i, v_3^i, \ldots, v_{2n+1}^i\}$ – множина вершин i-ї копії циклу C_{2n+1}, $\{u_1, u_2, \ldots, u_m\}$ – множина вершин циклу C_m, де $m \geq 3$. Задамо вершину розмітку f графа G порядку $|V(G)| = 2m(n + 1)$ наступним чином:

$$f(u_i) = 2m(i-1),$$

$$f\left(v_j^i\right) = 2m(i-1) + j,$$

де $i = 1, 2, \ldots, m, j = 1, 2, \ldots, 2n+1$.

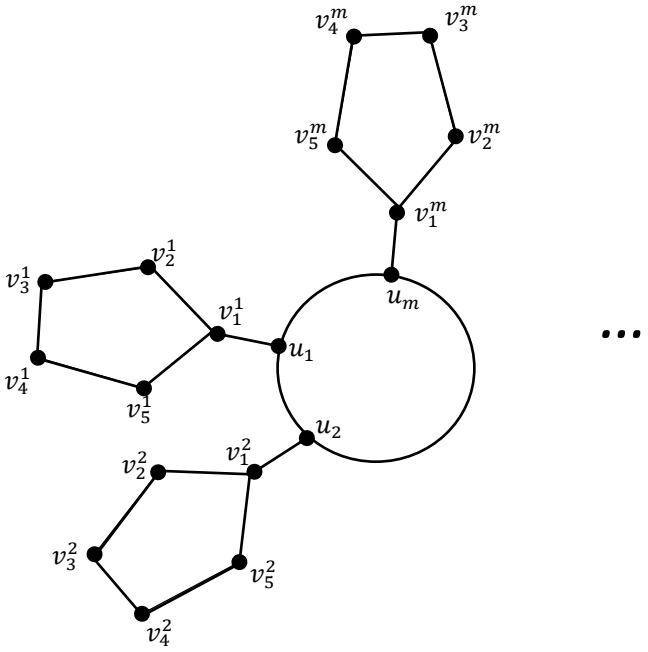


Рис. 1. Реберне з'єднання m копій циклу C_{2n+1} і циклу C_m

Таким чином, f – бієктне відображення множини вершин на множину чисел $\{0, 1, 2, 3, 4, \dots, 2m(n+1)-1\}$.

Для ребер циклу C_{2n+1} отримаємо мітки:

$f^*(v_j^i, v_{j+1}^i) = |[f(v_j^i)]^2 - [f(v_{j+1}^i)]^2| = |2m(i-1)+j + 2m(i-1)+j+1| = 4m(i-1)+2j+1$ (1)

$f^*(v_{2n+1}^i, v_1^i) = |[f(v_{2n+1}^i)]^2 - [f(v_1^i)]^2| = |(2m(i-1)+2n+1)^2 - (2m(i-1)+1)^2| == 8mn(i-1)+4n^2+4n$ (2)

Для ребер циклу C_m та реберного з'єднання m копій циклу C_{2n+1} отримаємо мітки:

$f^*(v_1^i, u_i) = |[f(v_1^i)]^2 - [f(u_i)]^2| = |2m(i-1)+1 + 2m(i-1)| = 4m(i-1)+1,$ (3)

$f^*(u_i, u_{i+1}) = |[f(u_i)]^2 - [f(u_{i+1}]^2| = |2m(i-1)^2 - 2mi^2| = 4mi - 2m,$ (4)

де $1 \le i \le m-1$,

$f^*(u_m, u_1) = |[f(u_m)]^2 - [f(u_1{}^2| = |2m(m-1)^2 - 0| = 2m^3 - 2m^2 + 2m,$ (5)

Мітки ребер, обчислені з використанням формул (1) та (3) утворюють множину непарних різних чисел $S_1 = \{1, 3, 5, 7, 9, \ldots, 4m(n+1) - 3\}$. З формули (2) одержимо одноелементну множину $S_2 = \{2m^3 - 2m^2 + 2m\}$.

Мітки ребер, обчислені з використанням формул (4) та (5) утворюють множину парних різних чисел $S_3 = \{2m, 6m, 10m, \ldots, 4m^2 - 2m\}$.

У результаті зазначених дій формується множина $S = S_1 \cup S_2 \cup S_3 \subset N$, що складається з різних чисел. Отже, f – квадратна різницева розмітка. Теорему доведено.

Для прикладу розглянемо граф, отриманий реберним з'єднанням 5 копій циклу C_5 і циклу C_5, зображений на рис. 2. Побудуємо вершинну розмітку f та отримаємо *SDL* даного графа.

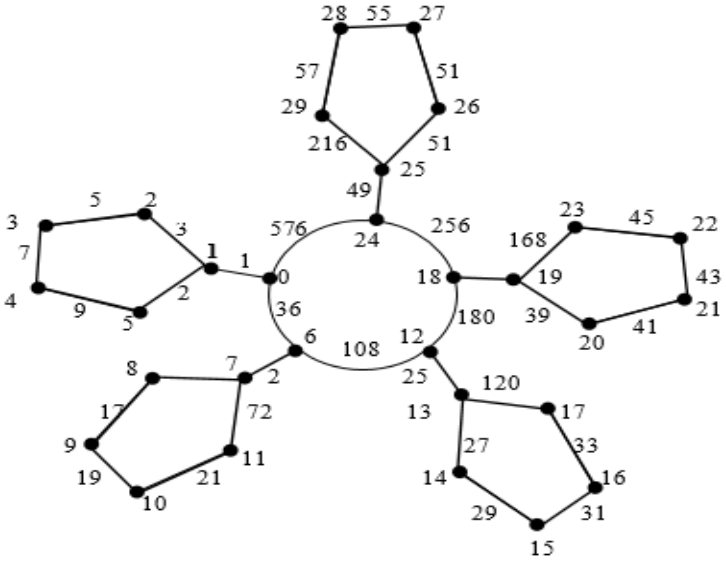


Рис. 2. Реберне з'єднання 5 копій циклу C_5 і циклу C_5

Література

1. T. Geetha and D. Kalamani, «Square Difference Labeling for Some Graphs», *Journal of Computer and Mathematical* Sciences,Vol.10, №4, 695-704, (2019).

URL: https://www.researchgate.net/publication/334815985_Square_Difference_Labeling_for_Some_Graphs

(дата звернення: 20. 04. 2023).

2. J. Gallian, «A dynamic survey of graph labeling» *The Electronic journal of Combinatorics*, pp. 1-623, 2022.

URL: https://www.combinatorics.org/files/Surveys/ds6/ds6v25-2022.pdf (дата звернення: 20. 04. 2023).

РЕЗОЛЮЦІЯ
XXV Міжнародного науково-практичного семінару «Комбінаторні конфігурації та їхні застосування»
14-16 червня 2023 р.

Учасники XXV 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 34 навчальних заклади та організації України та світу. В семінарі брали участь представники вищих навчальних закладів та організацій:

1. Національний університет «Запорізька політехніка»
2. Льотна академія Національного авіаційного університету
3. Інститут кібернетики імені В. М. Глушкова НАН України
4. Київський національний університет імені Тараса Шевченка
5. Національний авіаційний університет
6. Запорізький національний університет
7. Центральноукраїнський національний технічний університет
8. Харківський національний університет міського господарства імені О.Бекетова
9. Національний технічний університет «Дніпровська політехніка»
10. Міжнародний науково-навчальний центр інформаційних технологій та систем НАН та МОН України
11. Національний університет «Львівська політехніка»
12. ДВНЗ «Донбаський державний педагогічний університет»
13. Запорізький обласний інститут післядипломної педагогічної освіти
14. ДВНЗ «Ужгородський національний університет»
15. Інститут фізики НАН України
16. Дніпровський національний університет імені Олеся Гончара
17. Національний університет водного господарства та природокористування
18. Центральноукраїнський державний педагогічний університет імені В. Винниченка
19. Центральноукраїнський національний технічний університет
20. Відкритий міжнародний університет розвитку людини “Україна”
21. Громадська організація «Системні дослідження»
22. Українське фізичне товариство
23. Інститут проблем машинобудування ім. А.М. Підгорного НАН

України
24. Інститут технічної механіки НАН України,
25. Інститут математики НАН України
26. Придніпровська державна академія будівництва та архітектури
27. Freshcode, м.Запоріжжя
28. Institute of Control Systems, Azerbaijan National Academy of Sciences
29. Warwick Business School, UK
30. School of Computer Science, University of St Andrews, UK
31. University of Telecommunications and Posts, Sofia, Bulgaria
32. Інституту інформаційних технологій Лодзьського політехнічного університету, Польща
33. Самаркандський філіал Ташкентського університету інформаційних технологій, Узбекістан
34. Ургенчський державний університет, Узбекістан

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 напрямами:

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 100 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;

14 студентів і аспірантів.

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, Узбекистан, Польща, United Kindom, Азербайджан.

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.

Підкреслили багатовимірність розглянутих задач.

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 практичного застосування результатів наукових досліджень в сфері комбінаторики. Наступного разу зробити віртуальну постер сесію.
2. Поширювати наукові зв’язки з вітчизняними та закордонними провідними закладами для організації і реалізації спільних наукових та освітніх проєктів з розроблення математичних моделей проблем, визначених в дискретних та неперервних просторах, застосування методів ШІ (ML, DL тощо), розроблення курсів для стажування.
3. Підтримати співробітництво або діяльність за принципами відкритої науки в області розробки та необхідність поширення та навчання сучасним практикам роботи з кодом і даними за

принципами відкритої науки, поширювати практичні сучасні навички роботи з кодом та даними.

4. Визначити найкращі доповіді серед молодих вчених семінару, а саме: Станислав Івасів, Олег Дем'янів, Поліна Василькович та Денис Сташук.
5. Акцентувати увагу науковців і викладачів на необхідності розвинення і неухильного використання української наукової термінології у сфері математики та інформатики.
6. 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.

В.о. зав. каф. СА та ОМ Еліна ТЕРЕЩЕНКО

ЗМІСТ

Вітальні слова учасникам семінару 7

ДОПОВІДІ 10

Частина І 10

Shergin V.L., Chala L.E., Udovenko S.G.
Artificial Intelligence dept. NURE, Kharkov, Ukraine,
Informatics and Computer Technics dept. KhNUE, Kharkov, Ukraine
BOUNDARIES OF THE ASSORTATIVITY INDEX FOR SCALE-FREE NETWORKS 10

Shergin V.L., Chala L.E., Udovenko S.G.
Artificial Intelligence dept. NURE, Kharkov, Ukraine,
Informatics and Computer Engineering dept. KNUE, Kharkov, Ukraine
EXPLICIT ESTIMATES OF THE PROBABILITIES OF NASH-EQUILIBRIUM STRATEGIES IN LUPI GAME 14

Kalashnyk G.A., Kalashnyk-Rybalko M.A.
Flight Academy of the National Aviation University
FEATURES OF PROVISION OF FUNCTIONAL STABILITY OF AIRCRAFT ONBOARD EQUIPMENT COMPLEX 18

Shmelova T., Yatsko M., Sierostanov I.
National Aviation UniversityNational Aviation University, aircompany "Bees Airlines"Aircompany "Bees Airlines"
DECISION-MAKING MODELING IN CASE OFEMERGENCY "CAPTAIN INCAPACITATION
AT TAKE OFF 22

Глухов О.Д., Репета В.К.
Національний авіаційний університет
ПРО ЧИСЛО БОНДІВ 3-РЕБЕРНО ЗВ'ЯЗНОГО ГРАФА 27

Караваєв К. Д., Турчина В.А.
Дніпровський національний університет імені Олеся Гончара
ПРО ЗВ'ЯЗОК ЗАДАЧ ІЗОМОРФІЗМУ ГРАФІВ ТА УПОРЯДКУВАННЯ ЇХ ВЕРШИН 30

Козин И.В., Алломов З.К.
Запорожский национальный университет, Украина
Ургенчский государственный университет, Узбекистан
МЕТОД ОТЖИГА НА ФРАГМЕНТАРНОЙ СТРУКТУРЕ ДЛЯ ЗАДАЧИ РАЗМЕЩЕНИЯ ПРОИЗВОДСТВА 34

Кривий С.Л., Нортман Ю.О.
Київський національний університет імені Тараса Шевченка
ПРОТОКОЛ ОБМІНУ ПОВІДОМЛЕННЯМИ НА ОСНОВІ ЛІНІЙНИХ ФУНКЦІЙ І ОПЕРАТОРІВ 36

Мелешко Є.В.
Центральноукраїнський національний технічний університет
МЕТОД ПРОГРАМНОГО ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ З ВИКОРИСТАННЯМ ГРАФОВОЇ БАЗИ ДАНИХ NEO4J 40

Міхав В.В., Мелешко Є.В., Якименко М.С., Бащенко Д.В.
Центральноукраїнський національний технічний університет
ДОСЛІДЖЕННЯ МЕТОДІВ ЗБЕРІГАННЯ ДАНИХ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ НА ОСНОВІ ЗВ'ЯЗНИХ СПИСКІВ 44

Різник В.В.
Національний університет "Львівська політехніка»
МОЖЛИВОСТІ ЗАСТОСУВАННЯ КОМБІНАТОРНИХ КОНФІГУРАЦІЙ ДЛЯ ОПТИМІЗАЦІЇ ІНФОРМАЦІЙНИХ СИСТЕМ (ПАМ'ЯТІ С.ҐОЛОМБА 47

Сардак О.В., Сабиров З.Р.
Запорожский национальный университет, Украина
Ургенчский государственный университет, Узбекистан
АЛГОРИТМ ПРЫГАЮЩИХ ЛЯГУШЕК ДЛЯ ЗАДАЧИ ПРЯМОУГОЛЬНОГО РАСКРОЯ
54

Семенюта М.Ф.
Льотна академія Національного авіаційного університету
ПРО ОДИН ЗІ СПОСОБІВ ГЕНЕРАЦІЇ ГРАЦІОЗНИХ ДЕРЕВ З ДЕРЕВ МЕНШИХ ПОРЯДКІВ
57

Сікірда Ю.В., Шмельова Т.Ф.
Льотна академія Національного авіаційного університету
Національний авіаційний університет
НЕЙРОМЕРЕЖЕВА МОДЕЛЬ ДЛЯ СУМІСНОГО ПРИЙНЯТТЯ РІШЕНЬ АВІАЦІЙНИМ ПЕРСОНАЛОМ ПРИ ВИБОРІ ЗАПАСНОГО АЕРОДРОМУ В ПОЗАШТАТНІЙ СИТУАЦІЇ
60

Смірнова Т.В., Смірнов С.А., Смірнов О.А.
Центральноукраїнський національний технічний університет
ОПТИМАЛЬНА ТЕХНОЛОГІЯ ПОБУДОВИ ХМАРНОЇ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНОЇ СИСТЕМИ АВТОМАТИЗАЦІЇ ВИРОБНИЧИХ ПРОЦЕСІВ
64

Стецюк П.І., Романова Т.Є, Хом'як О.М.
Інститут кібернетики імені В.М. Глушкова НАН України
Інститут проблем машинобудування імені А.М. Підгорного НАН України
ПРО ВИКОРИСТАННЯ СОЛВЕРА BARON ДЛЯ ЗАДАЧІ ЗБАЛАНСОВАНОЇ УПАКОВКИ КРУГІВ
67

Тимофієва Н. К.
Міжнародний науково-навчальний центр інформаційних технологій та систем НАН та МОН України
ДИНАМІКА РОЗГОРТАННЯ ЛОГАРИФМІЧНИХ СПІРАЛЕЙ У ПРИРОДІ
72

Турчина В. А., Коваленко Є. О.
Дніпровський національний університет імені Олеся Гончара
ПАРАЛЕЛЬНІ УПОРЯДКУВАННЯ ДЛЯ ПОВНИХ ДВОДОЛЬНИХ ГРАФІВ
75

Турчина В.А., Малієнко О.О.
Дніпровський національний університет імені Олеся Гончара
СТАБІЛІЗАЦІЯ ДОВЖИНИ УПОРЯДКУВАННЯ ТА АНАЛІЗ ВПЛИВУ ЗМІН ПОЧАТКОВИХ ДАНИХ НА ОПТИМАЛЬНІСТЬ
79

ЧАСТИНА ІІ
83

Olexandr Konovalov
School of Computer Science, University of St Andrews, UK
OPEN SCIENCE AND REPRODUCIBLE RESEARCH
83

Sergey Sgadov
National University «Zaporizhzhia Polytechnic»
GAMMA-TETA ALGORITHM FOR GRAPHS PLANARIZATION
84

Tetiana Shmelova, Yuliya Sikirda, Maxim Yatsko
National Aviation University
Flight Academy of the National Aviation University
GENERAL TECHNIQUE OF DECISION-MAKING BY THE OPERATORS IN FLIGHT EMERGENCY
88

Pavlo Roibul
Zaporizhzhia National University, Ukraine
USING AN EVOLUTIONARY ALGORITHM IN THE PROBLEM OF OPTIMIZATION OF THE UKRAINIAN POWER GRID
94

Анна Бакурова, Станислав Ведмедєв, Еліна Терещенко
Національний університет «Запорізька політехніка»
ФОРМУВАННЯ ЗАПИТІВ КОМПЕТЕНЦІЇ ДО ОНТОЛОГІЇ
95

Євген Бровченко
Відкритий міжнародний університет розвитку людини «Україна», Київ
ЗАХИСТ НЕСТРУКТУРОВАНОЇ ІНФОРМАЦІЇ НА МОБІЛЬНОМУ ПРИСТРОЇ
100

Поліна Василькович, Валентина Турчина
Дніпропетровський національний університет імені Олеся Гончара
ВПЛИВ ТРАНЗИТИВНИХ ДУГ ГРАФА НА ОПТИМАЛЬНІСТЬ В УЗАГАЛЬНЕНІЙ ЗАДАЧІ ПАРАЛЕЛЬНОГО УПОРЯДКУВАННЯ
104

Олександр Глухов
Національний авіаційний університет
ПРО ЗВ'ЯЗНІСТЬ КВАЗІВИПАДКОВИХ ГРАФІВ
109

Інна Горват, Євген Кондрук, Владислав Нерода
Ужгородський національний університет
ПРАКТИЧНІ КЕЙСИ ЗАСТОСУВАННЯ КЛАСТЕРНОГО АНАЛІЗУ
119

Леонід Гуляницький, Ірина Рясна
Інститут кібернетики імені В. М. Глушкова НАН України
НЕЧІТКА ПОСТАНОВКА ЗАДАЧІ СЕГМЕНТУВАННЯ РИНКУ ПРАЦІ
117

Костянтин Караваєв, Валентина Турчина
Дніпровський національний університет імені Олеся Гончара
ПРО НЕОБХІДНІ ТА ДОСТАТНІ УМОВИ НАЯВНОСТІ АВТОМОРФІЗМУ У ПАРАЛЕЛЬНО-ПОСЛІДОВНИХ ГРАФАХ
122

Ігор Козін, Олег Сардак, Заріфбой Сабіров
Запорізький національний університет, Україна
Ургенчський державний університет, Узбекістан
КОМБІНОВАНА МОДЕЛЬ ЗАДАЧІ ПЛОСКОГО ПРЯМОКУТНОГО РОЗКРОЮ З РІЗНИМИ ТИПАМИ ЗАГОТОВОК
129

Козирський В., Полевецька О., Шендеровський В.
Інститут теоретичної фізики НАНУ,
Інститут фізики НАНУ
ПРО РОЛЬ ПОПУЛЯРИЗАЦІЇ НАУКИ В СУЧАСНОМУ СУСПІЛЬСТВІ
131

Сергій Курапов, Максим Давидовський
Запорізький національний університет,
КЗ «Запорізький обласний інститут післядипломної педагогічної освіти» ЗОР
АВТОМОРФІЗМ. СИМЕТРИЧНА СТІЙКІСТЬ ВАГ ВЕРШИН ГРАФУ
135

Василь Лещенко
Інститут математики НАН України
ПЛОЩИННІСТЬ ЛОКАЛЬНО ГОМЕОМОРФНИХ ГРАФІВ
139

Олег Павленко, Анатолій Тимошенко, Оксана Тимошенко
Відкритий міжнародний університет розвитку людини «Україна»
АЛГОРИТМ ПОШУКУ ЕКСТРЕМАЛЬНИХ ШЛЯХІВ НА ГРАФІ
143

Олена Подковаліхіна, Владислав Смола
Національний університет «Запорізька політехніка»
ДОСЛІДЖЕННЯ ЗАДАЧІ ПРО РЮКЗАК В УМОВАХ НЕВИЗНАЧЕНОСТІ
151

Юрій Прокопчук
Інститут технічної механіки НАН України,
Придніпровська державна академія будівництва та архітектури
КОМБІНАТОРНЕ УЗАГАЛЬНЕННЯ: МОДЕЛІ КРИТИЧНИХ ЗНАНЬ
157

Володимир Різник, Михайло Соломко, Олег Дем'янів, Станіслав Івасів
Національний університет "Львівська політехніка", м. Львів, Україна
МЕТОД ОПРАЦЮВАННЯ ІНФОРМАЦІЇ В СИСТЕМАХ КООРДИНАТ МНОГОВИДІВ
162

Марина Семенюта
Льотна академія НАУ
МЕТОД ПОБУДОВИ ГРАЦІОЗНИХ УНІЦИКЛІЧНИХ ГРАФІВ
170

Олексій Сергєєв, Світлана Ус
Національний технічний університет «Дніпровська політехніка»
ПРО РОЗВ'ЯЗАННЯ ЗАДАЧІ МЕДИЧНОЇ ЛОГІСТИКИ ЯК ДВОЕТАПНОЇ ЗАДАЧІ РОЗМІЩЕННЯ З ОБМЕЖЕННЯМИ НА МАКСИМАЛЬНУ КІЛЬКІСТЬ ОБ'ЄКТІВ
173

Руслан Сіренко
Національний університет «Запорізька політехніка»
ВІЗУАЛІЗАЦІЯ СПОРІДНЕНОСТІ ЛІНІЙ СОНЯШНИКУ ЗА МОРФОЛОГІЧНИМИ ОЗНАКАМИ
178

Михайло Соломко
Національний університет водного господарства та природокористування
МІНІМІЗАЦІЯ ЧАСТКОВО ВИЗНАЧЕНИХ БУЛЕВИХ ФУНКЦІЙ МЕТОДОМ ОБРАЗНИХ ПЕРЕТВОРЕНЬ
183

Денис Сташук, Валерій Дубровін, Юлія Тарасова
Національний університет «Запорізька політехніка»
FRESHCODE
ВИПАДКОВИЙ ЛІС: ПОШУК АНОМАЛІЙ У КОМП'ЮТЕРНИХ МЕРЕЖАХ
190

Петро Стецюк, Ольга Хом'як
Інститут кібернетики імені В.М. Глушкова НАН України
ДВОЕТАПНА ТРАНСПОРТНА ЗАДАЧА З УМОВАМИ НА ПОТРЕБИ СПОЖИВАЧІВ ТА ПРОПУСКНІ СПРОМОЖНОСТІ ПРОМІЖНИХ ПУНКТІВ
195

Еліна Терещенко, Антон Рябенко
Національний університет «Запорізька політехніка»
ВЛАСТИВІСТЬ ПОВНОТИ АБО КВАЗІПОВНОТИ БАГАТОКРИТЕРІАЛЬНИХ ЗАДАЧ ПРИ ЗБІЛЬШЕННІ КІЛЬКОСТІ КРИТЕРІЇВ
201

Владислав Терницький, Дмитро Широкорад
Національний університет «Запорізька політехніка»
ДОСЛІДЖЕННЯ ПІДХОДІВ ГЛИБИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ ТА ОБРОБКИ ПРИРОДНОЇ МОВИ: КОНЦЕПТ ДЛЯ СИСТЕМИ Є-ВОРОГ
206

Надія Тимофієва
Міжнародний науково-навчальний центр інформаційних технологій та систем НАН та МОН України
ВЛАСТИВОСТІ КОМБІНАТОРНИХ КОНФІГУРАЦІЙ ТА ЇХНЄ ВИКОРИСТАННЯ В КОМБІНАТОРНІЙ ОПТИМІЗАЦІЇ
210

Валентина Турчина, Євген Коваленко
Дніпровський національний університет імені Олеся Гончара
ПЕРЕРИВАННЯ В ЗАДАЧАХ УПОРЯДКУВАННЯ ВЕРШИН ГРАЦІОЗНИХ ДЕРЕВ
214

Михайло Хурдайло
Запорізький національний університет
ВИКОРИСТАННЯ ФРАГМЕНТАРНОЇ МОДЕЛІ ТА ЕВОЛЮЦІЙНОГО АЛГОРИТМУ ДЛЯ РОЗВ'ЯЗКУ ЗАДАЧ РОЗПОДІЛУ РЕСУРСІВ ЧАСУ
219

Зоя Шерман
Міжнародний європейський університет
SDL ЛАНЦЮГОВОГО З'ЄДНАННЯ ПЕВНИХ ТИПІВ ГРАФІВ
221

РЕЗОЛЮЦІЯ XXV Міжнародного науково-практичного семінару «Комбінаторні конфігурації та їхні застосування» 14-16 червня 2023 р.
226

CONTENT

Congratulatory words to the participants of the seminar 7

REPORTS 10

PART I 10

Shergin V.L., Chala L.E., Udovenko S.G.
Artificial Intelligence dept. NURE, Kharkov, Ukraine,
Informatics and Computer Technics dept. KhNUE, Kharkov, Ukraine
BOUNDARIES OF THE ASSORTATIVITY INDEX FOR SCALE-FREE NETWORKS . 10

Shergin V.L., Chala L.E., Udovenko S.G.
Artificial Intelligence dept. NURE, Kharkov, Ukraine,
Informatics and Computer Engineering dept. KNUE, Kharkov, Ukraine
EXPLICIT ESTIMATES OF THE PROBABILITIES OF NASH-EQUILIBRIUM STRATEGIES IN LUPI GAME 14

Kalashnyk G.A., Kalashnyk-Rybalko M.A.
Flight Academy of the National Aviation University
FEATURES OF PROVISION OF FUNCTIONAL STABILITY OF AIRCRAFT ONBOARD EQUIPMENT COMPLEX ...…… ……..…… 18

Shmelova T., Yatsko M., Sierostanov I.
National Aviation University
Aircompany "Bees Airlines"
DECISION-MAKING MODELING IN CASE OFEMERGENCY "CAPTAIN INCAPACITATION
AT TAKE OFF" …… …….. 22

Глухов О.Д., Репета В.К.
National Aviation University
ON THE NUMBER OF BONDS OF A 3-EDGE CONNECTED GRAPH 27

Караваєв К. Д., Турчина В.А.
Дніпровський національний університет імені Олеся Гончара
ON THE RELATION BETWEEN GRAPH ISOMORPHISM AND PARALLEL SEQUENCING PROBLEMS
30

Kozin I. V., Алломов З.К.
Zaporizhzhia National University, Ukraine
Ургенчский государственный университет, Узбекистан
ANNEALING METHOD ON A FRAGMENTAL STRUCTURE FOR THE PROBLEM OF PRODUCTION LOCATION
34

Кривий С.Л., Нортман Ю.О.
Київський національний університет імені Тараса Шевченка
PROTOCOL FOR EXCHANGE INFORMATION ON THE BASE OF LINEAR FUNCTIONS AND OPERATORS
36

Meleshko Ye.
Центральноукраїнський національний технічний університет
A METHOD OF COMPUTER SIMULATION MODELING OF A RECOMMENDATION SYSTEM USING THE GRAPH DATABASE NEO4J ……..
40

Міхав В.В., Meleshko Ye., Якименко М.С., Бащенко Д.В.
Центральноукраїнський національний технічний університет
RESEARCH METHODS OF DATA STORING OF A RECOMMENDATION SYSTEM BASED ON LINKED LISTS
44

Riznyk, V.V.
Lviv Polytechnic National University
POSSIBILITIES OF APPLICATION OF COMBINATORIAL CONFIGURATIONS FOR OPTIMIZATION OF INFORMATION SYSTEMS (MEMORY S. GOLOMB)
47

Сардак О.В., Сабиров З.Р.
Zaporizhzhia National University, Ukraine
Ургенчский государственный университет, Узбекистан
JUMPING FROG ALGORITHM FOR THE PROBLEM OF RECTANGULAR CUTTING
54

Семенюта М.Ф.
Flight Academy of the National Aviation University
ABOUT ONE OF THE WAYS OF GENERATION OF GRACIOUS TREES FROM TREES OF SMALLER ORDERS
57

Sikirda Yu., Shmelova T.
Flight Academy of the National Aviation University
National Aviation University
NETWORK MODEL FOR COLLABORATIVE DECISION MAKING BY THE AVIATION PERSONNEL DURING CHOOSING AN ALTERNATE AERODROME IN EMERGENCY
60

Смірнова Т.В., Смірнов С.А., Смірнов О.А.
Центральноукраїнський національний технічний університет
OPTIMAL TECHNOLOGY OF CONSTRUCTION OF CLOUD INFORMATION AND COMMUNICATION SYSTEM OF AUTOMATION OF PRODUCTION PROCESSES
64

Stetsyuk P., Romanova T., Хом'як О.М.
Інститут кібернетики імені В.М. Глушкова НАН України
Інститут проблем машинобудування імені А.М. Підгорного НАН України
ON USING BARON SOLVER FOR BALANCED CIRCULAR PACKING PROBLEM
67

Тимофієва Н. К.
Міжнародний науково-навчальний центр інформаційних технологій та систем НАН та МОН України
DYNAMICS OF LOGARITHMIC SPIRAL DEVELOPMENT IN NATURE
72

Турчина В. А., Коваленко Є. О.
Дніпровський національний університет імені Олеся Гончара
PARALLEL SEQUENCINGS FOR COMPLETE BIPARTITE GRAPHS
75

Турчина В.А., Малієнко О.О.
Дніпровський національний університет імені Олеся Гончара
STABILIZATION OF ORDER LENGTH AND ANALYSIS OF THE INFLUENCE OF CHANGES IN INITIAL DATA ON THE OPTIMALITY
79

Part II

83

Olexandr Konovalov
School of Computer Science, University of St Andrews, UK
OPEN SCIENCE AND REPRODUCIBLE RESEARCH

83

Sergey Sgadov
National University «Zaporizhzhia Polytechnic»
GAMMA-TETA ALGORITHM FOR GRAPHS PLANARIZATION

84

Tetiana Shmelova, Yuliya Sikirda, Maxim Yatsko
National Aviation University
Flight Academy of the National Aviation University
GENERAL TECHNIQUE OF DECISION-MAKING BY THE OPERATORS IN FLIGHT EMERGENCY

88

Pavlo Roibul
Zaporizhzhia National University, Ukraine
USING AN EVOLUTIONARY ALGORITHM IN THE PROBLEM OF OPTIMIZATION OF THE UKRAINIAN POWER GRID

94

Anna Bakurova, Stanyslav Vedmedev, Elina Tereschenko
National University «Zaporizhzhia Polytechnic»
FORMATION OF COMPETENCE REQUESTS TO ONTOLOGY

95

Євген Бровченко
Відкритий міжнародний університет розвитку людини «Україна», Київ
PROTECTION OF UNSTRUCTURED INFORMATION ON MOBILE

100

Поліна Василькович, Валентина Турчина
Дніпропетровський національний університет імені Олеся Гончара
INFLUENCE OF TRANSITIVE GRAPH ARCS ON OPTIMALITY IN THE GENERALIZED PARALLEL ARRANGEMENT PROBLEM

104

Олександр Глухов
National Aviation University
ON THE CONNECTIVITY OF QUASI-RANDOM GRAPHS

109

Інна Горват, Євген Кондрук, Владислав Нерода
Ужгородський національний університет
PRACTICAL CASES OF CLUSTER ANALYSIS
119

Hulianytskyi L.F., Riasna I.I.
Інститут кібернетики імені В. М. Глушкова НАН України
FUZZY STATEMENT OF LABOR MARKET SEGMANTATION PROBLEM
117

Костянтин Караваєв, Валентина Турчина
Дніпровський національний університет імені Олеся Гончара
ON NECESSARY AND SUFFICIENT CONDITIONS FOR THE EXISTENCE OF AUTOMORPHISM IN SERIES–PARALLEL GRAPHS
122

Ігор Козін, Олег Сардак, Заріфбой Сабіров
Zaporizhzhia National University, Ukraine
Ургенчський державний університет, Узбекістан
COMBINED MODEL OF THE PROBLEM OF FLAT RECTANGULAR CUTTING WITH DIFFERENT TYPES OF BLANKS
129

Козирський В., Полевецька О., Шендеровський В.
Інститут теоретичної фізики НАНУ,
Інститут фізики НАНУ
ON THE ROLE OF SCIENCE POPULARIZATION IN MODERN SOCIETY
131

Сергій Курапов, Максим Давидовський
Zaporizhzhia National University, Ukraine,
КЗ «Запорізький обласний інститут післядипломної педагогічної освіти» ЗОР
AUTOMORPHISM. SYMMETRIC STABILITY OF GRAPH VERTEX WEIGHTS
135

Василь Лещенко
Інститут математики НАН України
PLANENESS OF LOCALLY HOMEOMORPHIC GRAPHS
139

Pavlenko O. Yu., Tymoshenko A.G.,, Tymoshenko O.,
Відкритий міжнародний університет розвитку людини «Україна»
ALGORITHM FOR SEARCHING EXTREME PATHS ON A GRAPH
143

Олена Подковаліхіна, Владислав Смола
National University «Zaporizhzhia Polytechnic»
THE RESEARCH OF THE KNAPSACK PROBLEM UNDER CONDITIONS OF UNCERTAINTY
151

Prokopchuk Y.
Інститут технічної механіки НАН України,
Придніпровська державна академія будівництва та архітектури
COMBINATORIAL GENERALIZATION: CRITICAL KNOWLEDGE MODELS
157

V.V.Riznyk, M.T.Solomko, S.S.Ivasiv, O.M.Demianiv
Lviv Polytechnic National University
METHOD OF PROCESSING INFORMATION IN COORDINATE SYSTEMS OF MANIFOLDS
162

Марина Семенюта
Flight Academy of the National Aviation University
METHOD FOR CONSTRUCTING GRACEFUL UNICYCLIC GRAPHS
170

Олексій Сергєєв, Світлана Ус
Національний технічний університет «Дніпровська політехніка»
ON THE SOLUTION OF THE PROBLEM OF MEDICAL LOGISTICS AS A TWO-STAGE PROBLEM OF PLACEMENT WITH RESTRICTIONS ON THE MAXIMUM NUMBER OF OBJECTS
173

Руслан Сіренко
National University «Zaporizhzhia Polytechnic»
VISUALIZATION OF BREED LINES OF SUNFLOWER BY MORPHOLOGICAL FEATURES
178

M.T.Solomko
Національний університет водного господарства та природокористування
MINIMIZATION OF PARTIALLY DEFINED BOOLEAN FUNCTIONS BY THE METHOD OF FIGURATIVE TRANSFORMATIONS
183

Денис Сташук, Валерій Дубровін, Юлія Тарасова
National University «Zaporizhzhia Polytechnic»
FRESHCODE
RANDOM FOREST: FINDING ANOMALIES IN COMPUTER NETWORKS
190

Петро Стецюк, Ольга Хом'як
Інститут кібернетики імені В.М. Глушкова НАН України
A TWO-STAGE TRANSPORTATION PROBLEM WITH CONDITIONS ON CONSUMER NEEDS AND CAPACITIES OF INTERMEDIATE POINTS
195

E. Tereschenko, Антон Рябенко
National University «Zaporizhzhia Polytechnic»
THE PROPERTY OF COMPLETENESS OR QUASI COMPLETENESS OF MULTI-CRITERIA PROBLEMS WITH INCREASING THE NUMBER OF CRITERIA
201

Владислав Терницький, Дмитро Широкорад
National University «Zaporizhzhia Polytechnic»
EXPLORING DEEP LEARNING APPROACHES FOR IMAGE RECOGNITION AND NATURAL LANGUAGE PROCESSING: THE CONCEPT FOR E-ENEMY
206

Надія Тимофієва
Міжнародний науково-навчальний центр інформаційних технологій та систем НАН та МОН України
PROPERTIES OF COMBINATORY CONFIGURATIONS AND THEIR USE IN COMBINATORY OPTIMIZATION
210

Валентина Турчина, Євген Коваленко
Дніпровський національний університет імені Олеся Гончара
INTERRUPTIONS IN THE GRACEFUL TREES' VERTICES PARALLEL SEQUENCING PROBLEMS
214

Михайло Хурдайло
Zaporizhzhia National University, Ukraine
USING FRAGMENTAL MODELS AND EVOLUTIONARY ALGORITHM FORSOLVING THE PROBLEMS OF TIME RESOURCE DISTRIBUTION
219

Зоя Шерман
Міжнародний Європейський університет
SQUARE DIFFERENCE LABELING FOR EDGE CONNECTION OF SOME GRAPHS 221

РЕЗОЛЮЦІЯ XXV Міжнародного науково-практичного семінару «Комбінаторні конфігурації та їхні застосування» 14-16 червня 2023 р. 226

Додаткова інформація про можливості публікації результатів досліджень

Посилання на видання IGI Global (USA) (публікації в журналах та видання книг):

ЖУРНАЛИ

- These Journals Indexed In: Web of Science Emerging Sources Citation Index (ESCI), SCOPUS, Compendex (Elsevier Engineering Index), INSPEC, PsycINFO, Ei Compendex, etc.

https://www.igi-global.com/journal/international-journal-information-technologies-systems/1098

2 рази/рік

https://www.igi-global.com/journal/international-journal-decision-support-system/1120

- *SCOPUS:*

https://www.igi-global.com/journal/international-journal-information-system-modeling/1157
https://www.igi-global.com/journal/international-journal-decision-support-system/1120
https://www.igi-global.com/journal/international-journal-applied-logistics/1151
etc.

- in Scopus (more 70 Journals)
  https://www.igi-global.com/journals/indices/scopus/39

https://www.igi-global.com/journal/international-journal-information-technologies-systems/1098

https://www.igi-global.com/article/collaborative-decision-making-cdm-in-emergency-caused-by-captain-incapacitation/320477

КНИГИ

Типи книг:

- Авторські (більше 6 розділів)
- Інші (більше 15 розділів)

https://www.igi-global.com/publish/submit-a-proposal/

Наукове електронне видання
Можна використовувати в локальному
та мережному режимах

Матеріали XXV Міжнародного науково-практичного семінару імені А.Я. Петренюка КОМБІНАТОРНІ КОНФІГУРАЦІЇ ТА ЇХНІ ЗАСТОСУВАННЯ

14-16 червня 2023 року

Один електронний оптичний диск (DVD-ROM); супровідна документація.
Тираж 100 прим. Зам. № 461

Видавець і виготовлювач
Національний університет «Запорізька політехніка»
Україна, 69063, м. Запоріжжя, вул. Жуковського, 64
Тел.: (061) 769–82–96, 220–12–14

Свідоцтво суб'єкта видавничої справи ДК № 6952 від 22.10.2019.