

МЕТОДИ КОМБІНАТОРНОЇ ОПТИМІЗАЦІЇ, ЯКІ ІНСПІРОВАНІ ЖИВОЮ ТА НЕЖИВОЮ ПРИРОДОЮ

Вінницький національний технічний університет

Анотація

У роботі проведено огляд методів комбінаторної оптимізації із групи "м'яких обчислень", які інспіровані живою та неживою природою. Дані методи є досить оригінальними та ефективними, а їх знання допоможе розв'язати ряд прикладних задач без особливих знань математики.

Ключові слова: комбінаторна оптимізація, задача комівояжера, гамільтоновий цикл, методи живої та неживої природи, імітація відпалу, генетичний та мурашиний алгоритми.

Abstract

In this paper has been conducted a brief overview of the methods of combinatorial optimization from the group of "soft computing", which are inspired by nature. These methods are quite original and effective, and knowledge of them will help the to solve a number of applied tasks without special knowledge of mathematics.

Keywords: combinatorial optimization, travelling salesman problem, the Hamiltonian cycle, nature inspired methods, simulation annealing, genetic and ant algorithms.

Вступ

Розглядається задача, яка була представлена на Всеукраїнській олімпіаді з «Комп'ютерних систем управління та автоматики – КСУА» (ВНТУ, м. Вінниця, 2012). Формулюється вона наступним чином: робот, який виїжджає з контрольної точки S , має побувати в кожній контрольній точці s_i один раз і повернутися до початку у S . Знайти найкоротший маршрут x , якщо відстані між контрольними точками статичні або динамічні. Представити найоригінальніше рішення на ПК [1].

Дану задачу можна звести до задачі комівояжера (travelling salesman problem) на графі $G(U, V)$. Можна виділити такі види даної задачі: симетричну та асиметричну, з частковим упорядкуванням; статичну і динамічну; евклідову та прямокутну тощо. У даній роботі будемо розглядати статичну симетричну задачу з евклідовою метрикою. Маршрут робота задається перестановкою цілих чисел

$$x = \{(v_1, v_2), (v_2, v_3), (v_3, v_4), \dots, (v_N, v_1)\}. \quad (1)$$

Кожному маршруту x ставиться у відповідність функція його довжини

$$f(x) = L(v_1, v_2) + L(v_2, v_3) + L(v_3, v_4) + \dots + L(v_N, v_1), \quad (2)$$

де $L(v_i, v_j)$ – елементи матриці суміжності.

Цільова функція $f(x)$ представляє собою довжину замкнутої ламаної, такої, що проходить через n заданих точок в порядку, який задає перестановка $x_i \in X_n$, де $X_n = \{x_i\}$. Необхідно знайти найменшу або оптимальну довжину цієї ламаної (гамільтоновий цикл найменшої довжини). Але кількість усіх можливих таких циклів на графі для симетричної задачі складає $(n-1)!/2$, а для асиметричної – $(n-1)!$. Абсолютно очевидно, що дана задача може бути вирішена перебором усіх варіантів проходження пунктів і вибором серед них оптимального [2]. Але проблема в тому, що кількість можливих маршрутів дуже швидко зростає зі збільшенням кількості пунктів. Наприклад, для графа порядку 100 кількість варіантів дорівнюватиме 156-розрядному числу, а потужний комп'ютер, який здатний перебирати мільйон варіантів в секунду, працюватиме із задачею впродовж $1,5 \cdot 10^{142}$ років. Крім того, поки невідомо жодного простого критерію або алгебраїчного методу, що

дозволяє відповісти на запитання, чи існує в довільному графі $G(U, V)$ гамільтоновий цикл. Критерії існування дані в роботах Пошана, Неша-Вільямса, Оре, Дірака є занадто загальними і не придатні для довільних графів, які зустрічаються на практиці. Алгебраїчні методи знаходження гамільтонових циклів не можуть бути застосовані до задач з більш ніж кількома десятками вершин, оскільки вони потребують занадто великого часу роботи [3, 4]. Тому необхідно використовувати методи, які дозволяють знайти не найкраще, але оптимальне рішення, причому також необхідно аналізувати структуру графа. Одним із таких методів є розроблена модифікація методу гілок та меж (branch and bound), яка запропонована групою авторів (Дж. Літл, К. Мурті, Д. Суїні, К. Керолл); метод Куна-Манкреса; нейромережі; метаевристичні методи. Із останньої групи виділяються методи, які інспіровані живою та неживою природою (nature inspired algorithms). До них прийнято відносити метод імітації відпалу (simulated annealing), еволюційні методи (evolutionary computation): генетичні алгоритми (genetic algorithms), наслідування мурашиної колонії або мурашині алгоритми (ant colony optimization) тощо. Вони зосереджують пошук на найбільш перспективних частинах області допустимих рішень, дозволяючи значно скоротити час обчислень та знайти певний оптимальний розв'язок [2, 5].

Метою даної роботи є огляд оригінальних метаевристичних методів комбінаторної оптимізації, які інспіровані живою та неживою природою, з використанням як тестового полігону задачі комівояжера.

Результати дослідження

Алгоритми, які інспіровані неживою природою, зазвичай представляють певний фізичний процес або певні культурні феномени (меми). Розглянемо один простий та популярний алгоритм.

Алгоритм імітації відпалу (алгоритм Метрополіса) [6-8] – загальний алгоритмічний метод вирішення задачі глобальної оптимізації. Був запропонований незалежно один від одного двома авторами Кіркпатріком [7] та Церні [8] для розв'язання комбінаторних задач дискретної оптимізації. Причому обидва автори використовували у тестах задачу комівояжера. Метод ґрунтується на імітації фізичного процесу, який відбувається при кристалізації речовини, в тому числі при відпалі металів. Передбачається, що атоми вже вишикувалися в кристалічну решітку, але ще допустимі переходи окремих атомів з одного стану в інший. Процес протікає при поступовому зниженні температури. Перехід атома з одного стану в інший відбувається з певною ймовірністю, причому вона зменшується зі зниженням температури T . Стійка кристалічна решітка відповідає мінімуму енергії атомів, тому атом або переходить в стан з меншим рівнем енергії, або залишається на місці. Робота алгоритму ґрунтується на імовірнісному правилі з використанням розподілу Гіббса та "колеса рулетки"

$$p = \begin{cases} 100\%, & \text{якщо } f(x_{new}) - f(x_{current}) < 0; \\ \exp\left(-\frac{\Delta E}{k_b \cdot T_i}\right) \cdot 100\%, & \text{якщо } f(x_{new}) - f(x_{current}) \geq 0, \end{cases} \quad (3)$$

де k_b – константа Больцмана або константа налаштування (часто приймається рівною одиниці).

Даний алгоритм є дуже популярним, оскільки його легко застосувати до різноманітних задач будь-якого рівня складності, в яких потрібно знайти оптимальний результат.

Алгоритми, які інспіровані живою природою, зазвичай визначають поведінку особин у зовнішньому середовищі. Розглянемо два найбільш популярні методи цієї групи, які застосовують до задачі комівояжера на практиці.

Генетичний алгоритм [9, 10] – це проста модель еволюції в природі, яка реалізована у виді комп'ютерної програми. Метаевристика вперше запропонована Дж. Холландом [9]. Вона дозволяє розв'язати певну задачу за поліноміальний час. У її основі лежить використання еволюційних принципів для пошуку оптимального розв'язання. Основний механізм еволюції — це *природний відбір*. Його суть полягає в тому, що більш пристосовані особи мають більше можливостей для виживання і розмноження і, отже, приносять більше нащадків, ніж погано пристосовані особини. Завдяки передаванню генетичної інформації (*генетичному спадкуванню*) нащадки успадковують від батьків основні їхні якості. Таким чином, нащадки сильних індивідуумів також будуть відносно добре пристосованими, а їхня частка в загальній масі особин буде зростати. Після зміни декількох

десятків або сотень поколінь (із N особин) середня пристосованість (*фітнес-функція*) особин даного виду помітно зростає. Імовірнісне правило прийняття рішення з використанням "колеса рулетки" виглядає так

$$p = \frac{f(x_i)}{\sum_{j=1}^N f(x_j)} \cdot 100\% . \quad (4)$$

Рекомбінація і мутація дозволяють змінюватися особинам і пристосовуватися до середовища. Такі алгоритми відносяться до адаптивних пошукових механізмів.

Мурашиний алгоритм або *алгоритм оптимізації наслідуванням мурашиної колонії* [11, 12] – один з ефективних поліноміальних алгоритмів для знаходження наближених розв'язків задачі комівояжера. Ідея запропонована бельгійським дослідником М. Доріго і апробована ним на задачі комівояжера [11]. *Мурашка* представляє собою програмного агента колонії, який забезпечується набором простих правил, що дозволяє йому вибирати шлях на графі. У методі оптимізації мурашиною колонією велика кількість штучних мурашок будують рішення на кожній ітерації випадковим і "жадібним" способом. Кожна мурашка вибирає наступне ребро для включення в своє часткове рішення, ґрунтуючись на евристичному оцінюванні цього ребра і кількості феромону, пов'язаного з цим елементом шляху. По суті для кожного мурахи перехід із пункту i в пункт j залежить від 3-ох складових: *пам'яті мурахи* M (список пунктів, які ще можна відвідати), *видимості* η_{ij} між пунктами (статична інформація), *сліду феромона* $\tau_{i,j}$ (динамічна інформація, досвід мурах). Вибір наступного пункту здійснюється за принципом "колеса рулетки" та імовірнісного рівняння

$$p = \begin{cases} \frac{\eta_{i,j}^\alpha \cdot \tau_{i,j}^\beta}{\sum_{j=1}^{\text{length}(M)} \eta_{i,j}^\alpha \cdot \tau_{i,j}^\beta} \cdot 100\%, & \text{якщо } j \in M; \\ 0\%, & \text{якщо } j \notin M. \end{cases} \quad (5)$$

де α і β – опційні параметри, які задають ваги видимості та феромону на ребрі.

Для покращення часових характеристик алгоритму в розрахунок іноді вводять *елітних мурах*, які підсилюють ребра найкращого маршруту. Основними перевагами даного методу є мала похибка знайденого рішення, низькі часові витрати при роботі з задачами великої розмірності, простота модифікації, можливість розпаралелювання тощо. Метод вважається одним із найефективніших і широко застосовується на практиці [12].

Висновки

У даній роботі розглянуто оригінальні комп'ютерні методи розв'язання задачі про робота з олімпіади КСУА-2012 [1]. Розв'язання даної задачі представляє відому складність (NP -повна), тому на практиці розв'язати її простим перебором варіантів не представляється можливим. Часто її зводять до задачі цілочисельного лінійного програмування, для вирішення якої використовується метод гілок та меж. Цікавими методами розв'язання даної задачі є методи із групи "м'яких обчислень", які інспіровані природою. Тому досить оригінальним було б рішення з використанням саме таких методів, які є досить ефективними та гнучкими. Ще одним підходом є використання штучних нейронних мереж Хопфілда, Хеммінга, Кохонена тощо.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Всеукраїнська студентська олімпіада зі спеціальності "Комп'ютеризовані системи управління та автоматика" [Електронний ресурс]. – Режим доступу: <http://olimpksua.vntu.edu.ua/tasksUA/task2012.rar>.
2. Іванов Ю.Ю. Вступ до Computer Science. Дискретна математика: цікава та не дуже: лекції, алгоритми та задачі / Ю.Ю. Іванов. – 2018. – 89 с. – Режим доступу: https://iq.vntu.edu.ua/method/read_url.php?tbl_num=2&url=/fdb/1166/Discrete_Math_by_IVANOV.djvu.

3. Алексеев С.В. Обработка информации в сложных организационных системах // С.В. Алексеев, В.О. Мартовицкий / Системы обработки информации. – 2014. – Выпуск 2 (118). – С. 84-88.
4. Кристофидес Н. Теория графов. Алгоритмический подход / Н. Кристофидес. – М.: Мир, 1978. – 360 с.
5. Ерзин А.И. Задачи маршрутизации / А.И. Ерзин, Ю.А. Кочетов. – Новосибирск: РИЦ НГУ, 2014. – 95 с.
6. Азенкотт Р. Процедура «отпуска» // Р. Азенкотт / Труды семинара Н. Бурбаки за 1988 г. Сборник статей: составитель А.И. Кострикин. – М.: Мир, 1990. – С. 235-251.
7. Kirkpatrick S. Optimization by Simulated Annealing // S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi / Science. – 1983. – P. 671-680.
8. Cerny V. Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm // V. Cerny / Journal of Optimization Theory Applications. – 1985. – № 45. – P. 41-51.
9. Holland J.H. Adaptation in Natural and Artificial Systems / J.H. Holland. – MIT Press, 1992. – 225 p.
10. Эволюционные методы моделирования и оптимизации сложных систем / Е.С. Семенкин, М.Н. Жукова, В.Г. Жуков, И.А. Панфилов, В.В. Тынченко. – Красноярск: СФУ, 2007. – 310 с.
11. Dorigo M. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem // M. Dorigo, L.M. Gambardella / IEEE Transactions of Evolutionary Computing. – 1997. – V. 1. – P. 53-66.
12. Shtovba S. Ant Algorithms: Theory and Applications // S. Shtovba / Programming and Computer Software. – 2005. – V. 31. – № 4. – P. 167-178.

Гайовий Артем Олександрович — студент групи I-15б, факультет комп'ютерних систем і автоматики, Вінницький національний технічний університет, м. Вінниця.

Іванов Юрій Юрійович — канд. техн. наук, асистент кафедри автоматики та інформаційно-виміральної техніки, Вінницький національний технічний університет, м. Вінниця, e-mail: Yura881990@i.ua.

Науковий керівник: **Іванов Юрій Юрійович** — канд. техн. наук, асистент кафедри автоматики та інформаційно-виміральної техніки, Вінницький національний технічний університет, м. Вінниця.

Gayovy Artem O. — student, Faculty of Computer Systems and Automation, Vinnytsia National Technical University, Vinnytsia.

Ivanov Yuriy Yu. — Cand. Sc. (Eng), Assistant Professor, Faculty of Computer Systems and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: Yura881990@i.ua.

Supervisor: **Ivanov Yuriy Yu.** — Cand. Sc. (Eng), Assistant Professor, Faculty of Computer Systems and Automation, Vinnytsia National Technical University, Vinnytsia.