

К.т.н., доцент Зорін Ю.М., магістрант Санжаревський Я.Ю.

**Національний технічний університету України
«Київський політехнічний інститут»**

**“МУРАШИНИЙ” АЛГОРИТМ КАЛЕНДАРНОГО
ПЛАНУВАННЯ В УМОВАХ ОБМЕЖЕННЯ РЕСУРСІВ**

Abstract

*Yuri Zorin, assoc. Prof., PhD; Jan Sanzharevsky, student
Ant algorithm for resource-constrained project scheduling problem*

The paper presents resource-constrained project scheduling problem solution by means of Ant Colony Optimization (ACO). Proposed approach combines Ant algorithm with some heuristics for the improvement of solution quality. An ACO algorithm and heuristics implementation are introduced.

Вступ

Задача календарного планування в умовах обмеження ресурсів (RCPSP) є комбінаторною задачею, яку відносять до класу NP-складних задач [1]. Це комплексна проблема, що узагальнює цілу низку задач, що є її окремими випадками. Даний підхід можна використовувати для вирішення проблеми планування етапів виконання проекту за різних умов (обмеження за ресурсами, часом тощо). Проект можна замінювати на довільний технологічний процес отже ця задача є актуальною для комп'ютерної інженерії.

Є деяка послідовність робіт

$$\bar{J} = J \cup J_0, J_0 = \{0, n+1\}, J = \{1, \dots, n\}, \quad (1)$$

де \bar{J} – загальна кількість робіт, J – роботи, визначені завданням, J_0 - спеціальні роботи (початкова, кінцева).

Спеціальні роботи характеризуються тим, що початкова немає передуючої роботи, кінцева – немає наступної.

Є множина типів ресурсів Q та визначена наступна функція:

$$\xi : Q \rightarrow Z_+, \quad (2)$$

тобто для кожного типу ресурсів задана кількість його екземплярів.

Для кожної роботи визначена функція, що ставить їй у відповідність довжину виконання в часі:

$$\zeta : \bar{J} \rightarrow Z_+. \quad (3)$$

Також задана функція, що встановлює міру потреби деякої роботи в деякому ресурсі в кожен момент часу (час дискретний):

$$\psi : \bar{J} \times Q \rightarrow Z_+. \quad (4)$$

Має місце порядок виконання работ, тобто задане відображення:

$$P : \bar{J} \rightarrow \mathfrak{R}(\bar{J}), \quad (5)$$

що визначає набір работ, що повинні бути виконані перед заданою.

В ході розв'язку задачі будується розклад, що визначається функцією:

$$s : \bar{J} \rightarrow Z_+, \quad (6)$$

яка визначає момент часу початку виконання кожної із работ.

Абсолютно природно для кожної роботи задається час закінчення її виконання:

$$f : \bar{J} \rightarrow Z_+, \quad f(j) = s(j) + \zeta(j). \quad (7)$$

Задача полягає у мінімізації часу виконання усіх работ згідно плану, тобто формально вона формулюється наступним чином:

$$\max\{f(j) \mid j \in \bar{J}\} - \min\{s(j) \mid j \in \bar{J}\} = f(n+1) - s(0) \rightarrow \min. \quad (8)$$

Постановка задачі

Метою роботи є розробка “мурашиного” алгоритму (МА) розв'язання RCPSP. Потрібно адаптувати методи мурашиних колоній до побудови розкладів таким чином, щоб результуючий розв'язок був допустимим:

1. $(\forall j \in \bar{J}) : s(j) \geq \max\{f(p) \mid p \in P(j)\}$.
2. $(\forall t - \text{момент часу}) (\forall q \in Q) \sum_{j \in \bar{J}, t \in [s(j), p(j))} \psi(j, q) \leq \xi(q)$. (9)

Отже, задача полягає у виборі вдалого методу генерації розкладу, набору евристик та адаптації самого МА для отримання якісних рішень, у тому числі мінімізації часу виконання алгоритму.

Адаптування МА та метод побудови розкладу

Як метод генерації розкладу був використаний алгоритм SSGS (Serial

Schedule Generation Scheme) [2]. Цей алгоритм дозволяє побудувати розклад за n кроків. Крім того відомо, що цей алгоритм дозволяє згенерувати оптимальний розв'язок.

На кожному етапі МА кожна з m мурах отримує деякий розклад. Для вибору наступної роботи мураха використовує порядок, що встановлений за допомогою SSGS. Традиційно, вибір виконується на основі наступних функцій:

$$1) \text{ Евристична інформація - } \eta : \bar{J} \times Z_+ \rightarrow R_+. \quad (10)$$

$$2) \text{ Феромонна інформація - } \tau : \bar{J} \times Z_+ \rightarrow R_+. \quad (11)$$

Обидві з цих функцій визначають наскільки доцільно розмістити дану роботу у визначеному місці розкладу SSGS.

Функція, що визначає ймовірність того, що робота i буде розміщена у місці j розкладу SSGS:

$$p : \bar{J} \times Z_+ \rightarrow R_+, p : (i, j) \mapsto \frac{[\tau(i, j)]^\alpha [\eta(i, j)]^\beta}{\sum_{h \in E} [\tau(h, j)]^\alpha [\eta(h, j)]^\beta}, \quad (12)$$

де E - множина робіт, що допустимо розмістити у j -тому місці, α, β - параметри алгоритму.

Тепер задамо дві допоміжні функції, що визначають два різних методи розрахунку ймовірності (12):

$$\tau_1(i, j) = id(\tau(i, j)) = \tau(i, j), \quad (13)$$

$$\tau_2(i, j) = \sum_{k=1}^j \gamma^{j-k} \tau(i, k), \quad (14)$$

де γ - додатній параметр алгоритму.

Розглянемо наступну криву у просторі відображень типу (11):

$$L : [\tau(i, j)](c) = c \cdot x(j) \cdot \tau_1(i, j) + (1-c) \cdot y(j) \cdot \tau_2(i, j), c \in [0;1]. \quad (15)$$

Функції x, y є ваговими коефіцієнтами у функціональному просторі:

$$x(j) = \sum_{h \in E} \tau_2(h, j). \quad (16)$$

$$y(j) = \sum_{h \in E} \tau_1(h, j). \quad (17)$$

Випаровування виконується тривіальним чином з деяким коефіцієнтом $\rho \in (0;1)$, тобто $\tau(i, j) := (1-\rho) \cdot \tau(i, j)$ [3].

Випишемо результуючу зміну кількості феромонів, що використовувалася в роботі i :

$$\tau(i, j) := (1 - \rho) \cdot \tau(i, j) + \frac{\rho}{2T^*}, \quad (18)$$

де T^* - час виконання усіх робіт для найкращого розкладу, що знайдений на даній ітерації.

Евристична функція має наступний вигляд:

$$\eta(i, j) = \max_{k \in E} \{LS(k) - ES(k)\} - (LS(i) - ES(i)) + 1, \quad (19)$$

де $LS(\cdot), ES(\cdot)$ - є, відповідно, найпізнішим та найранішим моментами часу початку обраної роботи.

Додатково було виконано аналіз впливу різних параметрів на якість алгоритму. Досліджувалися на локальні оптимуми функції, що мають своїми значеннями довжину в часі виконання розкладів, від підмножин параметрів.

Розглядалась стратегія з використанням елітних мурах. Мурахи цього типу виділяють феромони на ребрах, що відповідають глобальному найкращому рішенняю.

Наступна стратегія визначає, яке рішення вважається глобально найкращім. Якщо деяке рішення на протязі визначеної кількості ітерацій залишається найкращим, його замінюють локальним найкращим рішенням.

Результати

Результати тестування даного алгоритму на задачах з [4] показали, що використані евристики дають результати дещо кращі у порівнянні з іншими евристичними (в тому числі з генетичними алгоритмами). Подальші зусилля будуть скеровані на побудову гібридних алгоритмів на основі МА.

Література

1. Brucker P., Drexel R., Mohring H., Neumann K., Pesch E. Resource-constraint project scheduling: Notation, classification, models and methods. – Eur. J. Oper. Res., vol. 112, no. 1. – 1999. – pp. 3-41.
2. Kolisch R., Hartmann S., Heuristic algorithms for solving the resource-constrained project scheduling problem: Classification and computational analysis. – Kluwer. – 1999. – pp. 197-212.
3. Dorigo M., Stutzle T. Ant Colony Optimization. – MIT Press. – 2004. – 305p.
4. Kolisch R., Sprecher A. PSPLIB – A project scheduling problem library. – Eur. J. Oper. Res., vol. 96, no. 1. – 1996. – pp. 205-216.