

НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ БІОРЕСУРСІВ
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ



НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ Факультет інформаційних технологій Кафедра комп'ютерних наук

Методита та системи штучного інтелекту

Тема №6. Алгоритми еволюційного роєвого інтелекту

Київ - 2026

Зміст

- 1. Стандартний мурашиний алгоритм.*
- 2. Модифікації мурашиного алгоритму.*

Мета лекції

*Вивчення та дослідження евристичних алгоритмів
розв'язання проблем розуму*

Перелік літератури

1. Encyclopedia of artificial intelligence/Eds.: JRDopico, JDde laCalle, AP Sierra.-New York:Information Science Reference,2009. -Vol.1-3. - 1677p.
2. Gen M. Genetic algorithms and engineering design/M. Gen, R. Cheng.-New Jersey:John Wiley&Sons,1997. – 352p.
3. Haupt R. Practica genetic algorithms/R. Haupt, S. Haupt. -New Jersey:John Wiley&Sons,2004. – 261p.
4. Ємельянов В. В. Теорія та практика еволюційного моделювання / В. В. Ємельянов, В. В. Курейчик, Ст М. Курейчик. - М.: Фізмат - літ, 2003. - 432 с.
5. Курейчик В. М. Генетичні алгоритми: монографія/М. Курейчик. - Таганрог: ТРТУ, 1998. - 242 с.
6. Прогресивні технології моделювання, оптимізації та інтелектуальної автоматизації етапів життєвого циклу авіадвигунів: Монографія / О. В. Богуслаєв, Ал. А. Олійник, Ан. О. Олійник, Д. В. Павленко, С. А. Суботін; за ред. Д. В. Павленко, А. Суботіна. - Запоріжжя: ВАТ «МоторСіч», 2009. – 468 с.
7. Рассел С. Штучний інтелект: сучасний підхід /С. Рассел, П. Норвіг. – М.: Вільямс, 2006. – 1408 с.

8. Ротштейн А. П. Інтелектуальні технології ідентифікації: нечіткі множини, генетичні алгоритми, нейронні мережі / А. П. Ротштейн. - Вінниця: Універсум-Вінниця, 1999. – 320 с.

9. Руденко О. Г. Штучні нейронні мережі / О. Г. Руденко, Є. Ст. Бодянский. - Х.: Компанія ЗМІТ, 2006. – 404 с.

10. Рутковська Д. Нейронні мережі, генетичні алгоритми та нечіткі системи / Д. Рутковська, М. Р. Пілінський, Л. Рутковський; пров. з польськ. І. Д. Рудинського. – М.: Гаряча лінія – Телеком, 2004. – 452 с.

11. Скобцов Ю. А. Основи еволюційних обчислень / Ю. А. Скобцов. - Донецьк: ДонНТУ, 2008. – 330 с.

12. Суботін С. О. Неітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей: монографія / С. О. Суботін, А. О. Олійник, О. О. Олійник; під заг. ред. З О. Суботіна. - Запоріжжя: ЗНТУ, 2009. – 375 с.

13. Хайкін С. Нейронні мережі: повний курс / С. Хайкін. - СПб: Видавничий дім "Вільямс", 2005. - 1104 с.

14. Еволюційні методи комп'ютерного моделювання: монографія / А. Ф. Верлань, В. Д. Дмитрієнко, Н. І. Корсунів, Ст А. Шорох. - До: Наукова думка, 1992. – 256 с.

15. Ємельянов В. В., Курейчик Ст Ст, Курейчик В. М. Теорія та практика еволюційного моделювання. - М:Фізматліт, 2003. - 432 с. - [ISBN 5-9221-0337-7](#).
16. Курейчик В. М., Лебедев Б. К., Лебедев О. К. Пошукова адаптація: теорія та практика. - М:Фізматліт. - 272 с. - [ISBN 5-9221-0749-6](#).
17. Гладков Л. А., Курейчик Ст Ст, Курейчик В. М. Генетичні алгоритми: Навчальний посібник. - 2-ге вид. - М:Фізматліт, 2006. - 320 с. - [ISBN 5-9221-0510-8](#).
18. Гладков Л. А., Курейчик В. В., Курейчик Ст М. та ін. Біоінспіровані методи оптимізації: монографія. - М:Фізматліт, 2009. - 384 с. - [ISBN 978-5-9221-1101-0](#).
19. Рутковська Д., Пілінський М., Рутковський Л. Нейронні мережі, генетичні алгоритми та нечіткі системи = Sieci neuronowe, algorytmy genetyczne i systemy rozmyte. - 2-е вид. - М: Гаряча лінія-Телеком, 2008. - 452 с. - [ISBN 5-93517-103-1](#).

Історія роєвих обчислень

Можливість того, що обчислювальна система, наділена простими механізмами мінливості та відбору, могла б функціонувати за аналогією із законами еволюції в природних системах є дуже привабливою. Вона стала причиною появи низки обчислювальних моделей, побудованих за принципами, інспірованих природними системами. Однією з таких є алгоритми роєвого інтелекту.

Поняття роєвого інтелекту (Swarm intelligence) було введено Херардо Беніта Ван Цзіном 1989 року.

Під роєвим інтелектом розуміють систему, що самоорганізується, що складається з безлічі агентів. Агенти підпорядковуються простим правилам поведінки у навколишньому середовищі. Їхня проста взаємодія визначає колективну адаптацію. Таким чином, на поведінці простих агентів формується роєвий інтелект. Прикладами таких систем можуть бути мурашина колонія, бджолиний рій, зграя птахів, риб тощо.

Постановка завдання розбиття графа

Дан граф $G=(X, U)$

- $|X| = n$ – множина вершин (города)
- $|U| = m$ – множина ребер

Дана матриця суміжності $D_{i,j}$

- $i, j \in 1, 2, \dots, n$
- $D_{i,j} = 0$ або 1 (наявність ребер між вершинами i та j)

Дані потужності підграфів M_i

- $|M| = s$ – кількість підграфів

Мета – мінімізувати критерій

- критерій - кількість ребер між підграфами, ЦФ:

$$\sum_{i=0}^{i < s} \sum_{j=i+1}^{j < s} K_{i,j}$$

де $K_{i,j}$ – кількість ребер між підграфами i і j

1. СТАНДАРТНИЙ МУРАШИНИЙ АЛГОРИТМ

Мурашині алгоритми

Цей клас алгоритмів розроблявся у межах наукового напрямки «природні обчислення».

Дослідження у цій галузі розпочалися у середині 90-х років ХХ століття.

Автором ідеї є Марко Доріго.

Основна ідея – моделювання поведінки колонії мурах.

Постановка задачі

Даний граф $G = (X, U)$

$|X| = n$ - безліч вершин (міста)

$|U| = m$ - множина ребер (можливі шляхи між містами).

Дано матрицю чисел $D(i, j)$, де $i, j \in 1, 2, \dots, n$, що являють собою вартість переїзду з вершини x_i в x_j .

Мета: знайти перестановку ϕ з елементів множини X таку, що значення $D(\cdot)$ дорівнює:

$$D(\phi(1), \phi(n)) + \sum_i \{D(\phi(i), \phi(i+1))\} \rightarrow \min$$

Основні положення

Мурашина колонія – багатоагентна система.

Взаємодія – феромон, що відкладається мурахами на шляху, який пройшли.

При виборі напрямку руху мураха виходить з:

- Бажання пройти найкоротший шлях;
- Досвіду інших мурах, інформацію про який отримуємо безпосередньо через рівень феромонів на кожному шляху.

Від'ємний зворотній зв'язок (локальний оптимум)
– випарювання феромонів.

Властивості мурахи

Мураха має «пам'ять», в якій зберігається список місць $J_{i,k}$, які необхідно відвідати мурахою k , що знаходиться в місці i .

Мурахи мають «зір», який визначається за такою формулою:

$$\eta_{ij} = 1/D_{ij}.$$

Мураха вловлює слід феромону, який визначає бажання мурахи пройти по даному ребру. Рівень феромону на момент часу на ребрі D_{ij} буде відповідати $\tau_{ij}(t)$.

Рівень феромону

$$\Delta \tau_{ij,k}(t) = \begin{cases} \frac{Q}{L_k(t)}, (i, j) \in T_k(t) \\ 0, (i, j) \notin T_k(t) \end{cases} \quad (2)$$

Де Q - параметр, що має значення порядку довжини оптимального шляху (задається ЛПР),

$L_k(t)$ - довжина маршруту $T_k(t)$. Оновлення феромону визначається наступним виразом:

$$\tau_{ij}(t+1) = (1-p) \cdot \tau_{ij}(t) + \sum_{k=1}^m \Delta \tau_{ij,k}(t) \quad (3)$$

Де m – кількість мурах, p - коефіцієнт випаровування ($0 \leq p \leq 1$).

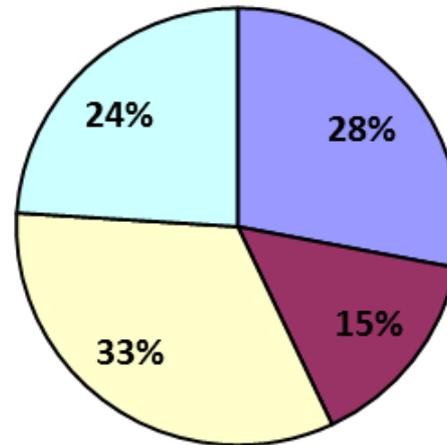
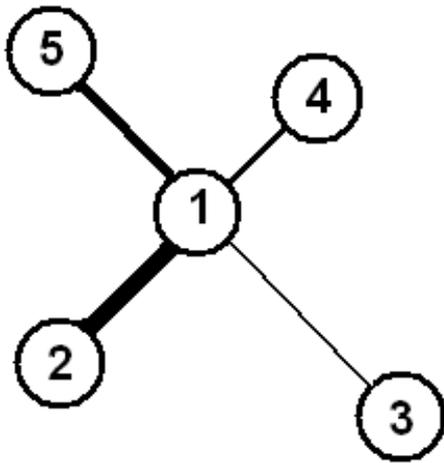
Ймовірність переходу у вершину

Ймовірність переходу мурахи з вершини i у вершину j визначатиметься таким співвідношенням:

$$\begin{cases} P_{ij,k}(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{l \in J_{i,k}} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}(t)]^\beta}, & j \in J_{i,k} \\ P_{ij,k}(t) = 0, & j \notin J_{i,k} \end{cases} \quad (1)$$

Де α, β - параметри, що задають ваги сліду феромону. Вони визначають «жадібність» мурахи. При $\alpha=0$ мураха прагне вибирати найкоротше ребро, при $\beta=0$ мураха прагне вибирати ребро з найбільшою кількістю феромону.

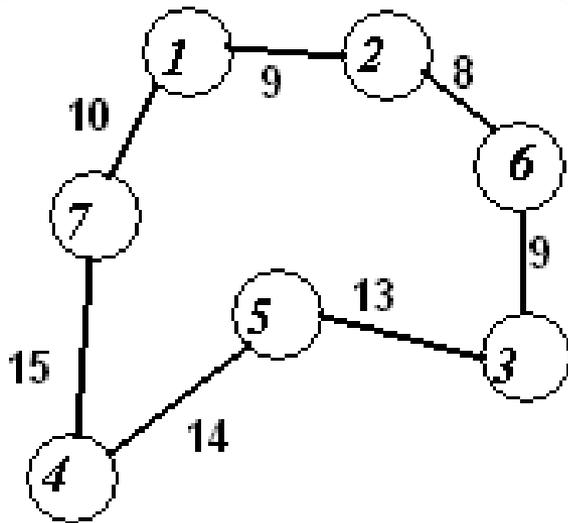
Ймовірність переходу у вершину



- $P(1,2)=28\%$
- $P(1,3)=15\%$
- $P(1,4)=33\%$
- $P(1,5)=24\%$

Товщина ребер показує рівень феромонів

Приклад розрахунку рівня феромонів



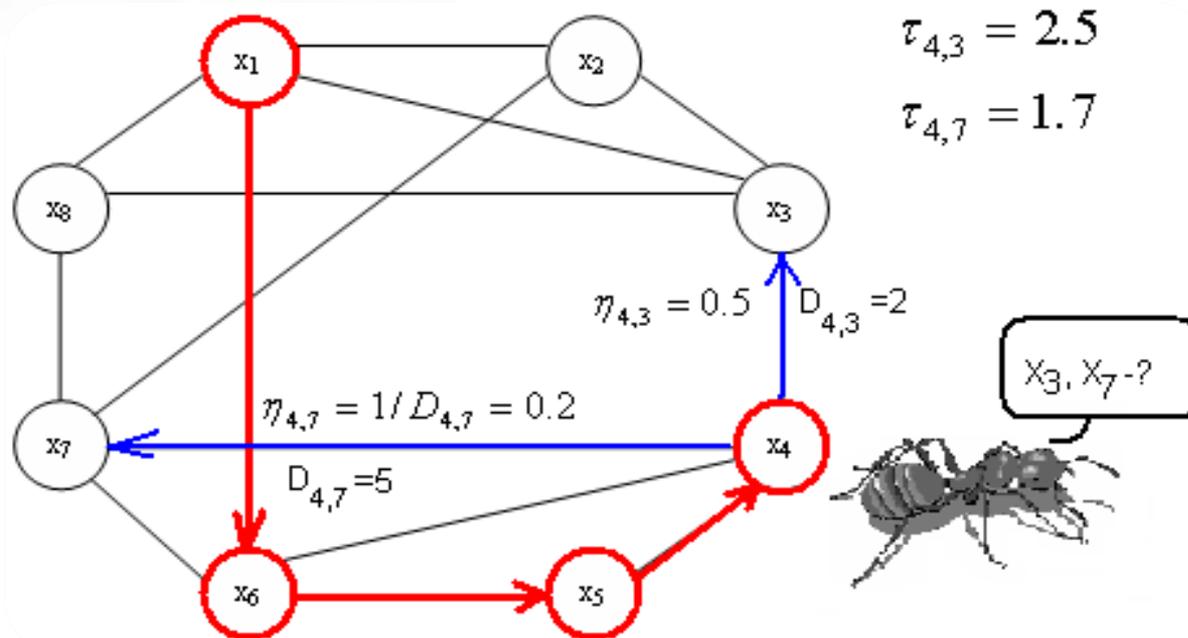
$$T_k = 1-2-6-3-5-4-7-1$$

$$L_k = 9 + 8 + 9 + 13 + 14 + 15 + 10 = 78$$

$$Q = 100$$

$$\Delta \tau_{ij,k}(t) = \frac{Q}{L_k(t)} = \frac{100}{78} = 1.282$$

Приклад поведінки мурахи



Поточна вершина: 4

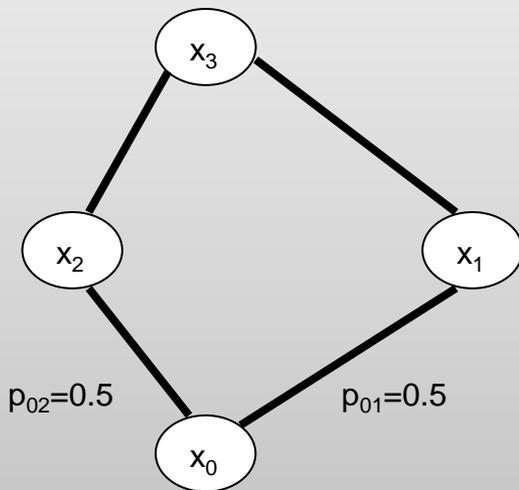
Пройдений маршрут: 1-6-5-4

Не пройдені вершини: 2, 3, 7, 8

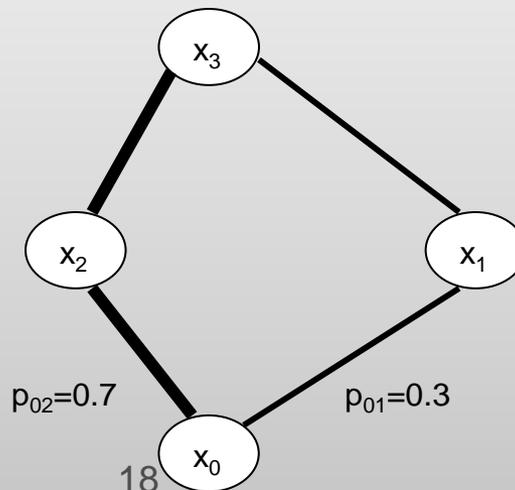
Мураха визначає наступну з суміжних вершину: 3, 7.

приклад

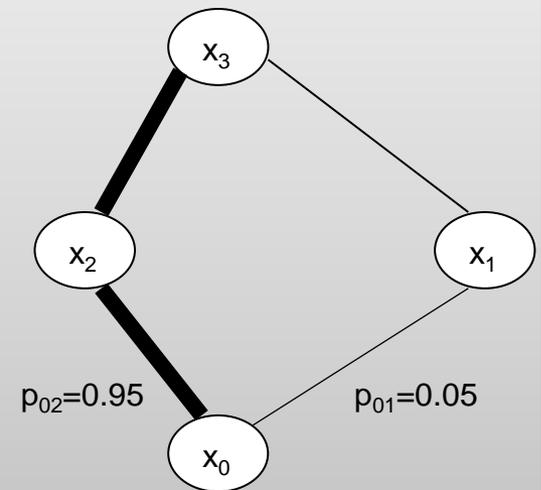
Розглянемо граф із 4 вершин. Знайдемо мінімальний шлях у графі від x_0 до x_3 . Товщина ліній відображає інтенсивність проходження мурах на цій ділянці, p_{01} і p_{02} - ймовірності вибору агентом ребер (x_0, x_1) та (x_0, x_2) відповідно. Спочатку, ймовірність переходу з вершини x_0 в x_1 і x_2 рівні. З часом бажання, а значить і можливість, вибору найкоротшого шляху збільшується.



а) початковий розподіл



б) проміжний розподіл



в) кінцевий розподіл

Тимчасова складність мурашиного алгоритму

ТСА за Дж.МакКонеллу, М.Дорігота оцінюється як:

$$O(t * m * n^2)$$

t – кількість ітерацій (час життя колонії)

m – кількість мурах

n – кість вершин в графі

2. МОДИФІКАЦІЇ МУРАШИНОГО АЛГОРИТМУ

«Елітні» мурахи

Рівень феромонів на ребрах кращого маршруту дорівнює наступній величині:

$$\Delta \tau_e(t) = A^e \cdot \frac{Q}{L^*(t)}$$

$L^*(t)$ - довжина кращого маршруту на момент часу t

A^e - «авторитет» елітних мурах

$A^e = 0$. Повністю ігнорується вплив «елітної» мурашки.

$A^e \in (0, 1)$. У цьому випадку рівень феромонів на найкращому маршруті зменшується.

$A^e = 1$. Вплив «елітних» мурах рівнозначно іншим.

$A^e \in (1, \infty)$. Посилюється вплив «елітних» мурах.



Початкове розташування колонії

«Ковдра» – стандартне розміщення мурах, у кожній вершині знаходиться по одній мурахі. Тоді ТСА: $O(t*n^3)$, оскільки $n=m$;

"Дробовик" - випадковий розподіл мурах на вершини графа;

"Фокусування" - вся колонія знаходиться в одній вершині;

«Блукаюча колонія» – на кожній ітерації, вся колонія переміщається у випадково обрану вершину.

Шаблони

Шаблони – фрагменти рішень.

Шаблон формується як масив V з n елементів, що складається з 0 і неповторних чисел від 1 до n , де n -число вершин.

Алгоритм декодування шаблонів для орієнтованого графа:

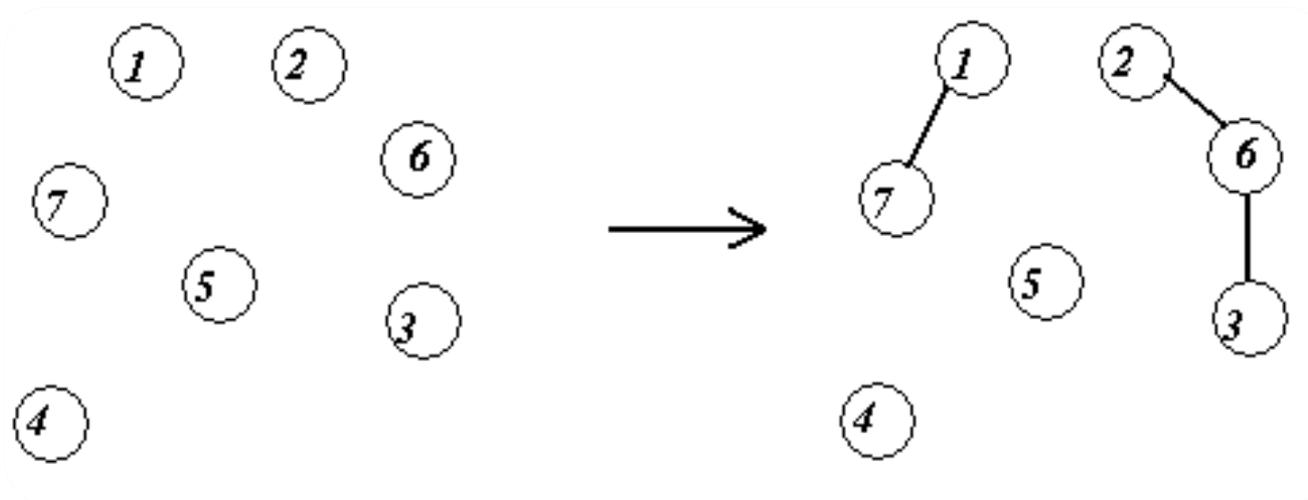
Якщо мураха потрапила у вершину i і $V_i \neq 0$, то він вибирає ребро (i, V_i)

Приклад побудови шаблону

Нехай сформовано наступний шаблон:

$\{7,6,0,0,0,3,0\}$

Тоді у рішенні присутні ребра (1-7), (2-6), (6-3)



Стратегії формування шаблону

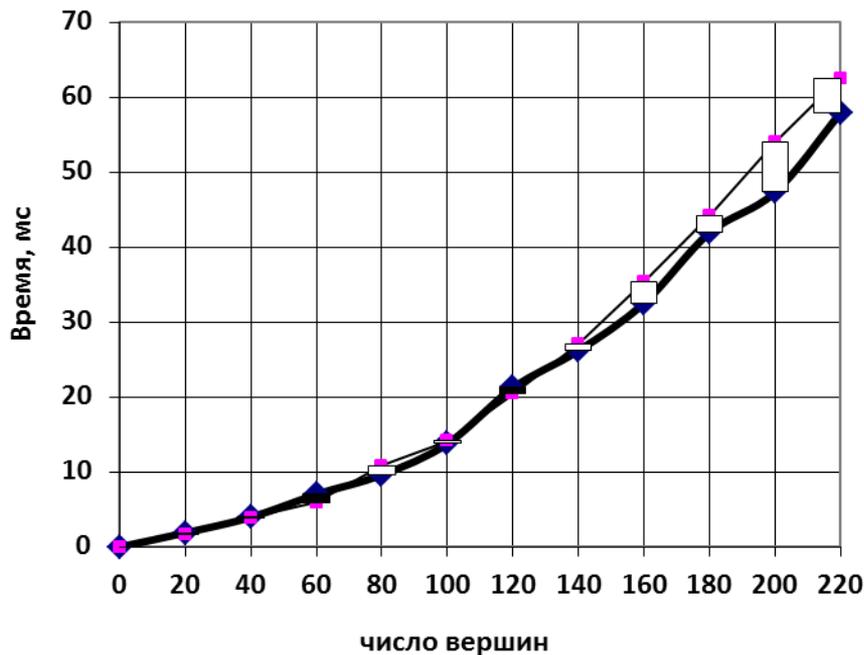
Статична

- Шаблон формується перед роботою мурашиного алгоритму на основі знань про довжину ребер – вибірка K найкоротших ребер

Динамічна

- Шаблон формується при функціонуванні мурашиного алгоритму. При цьому якщо вірогідність переходу з вершини i в j перевищуватиме задане обмеження ($P_{lim} < 1$), то ребро (i, j) вноситься до шаблону.

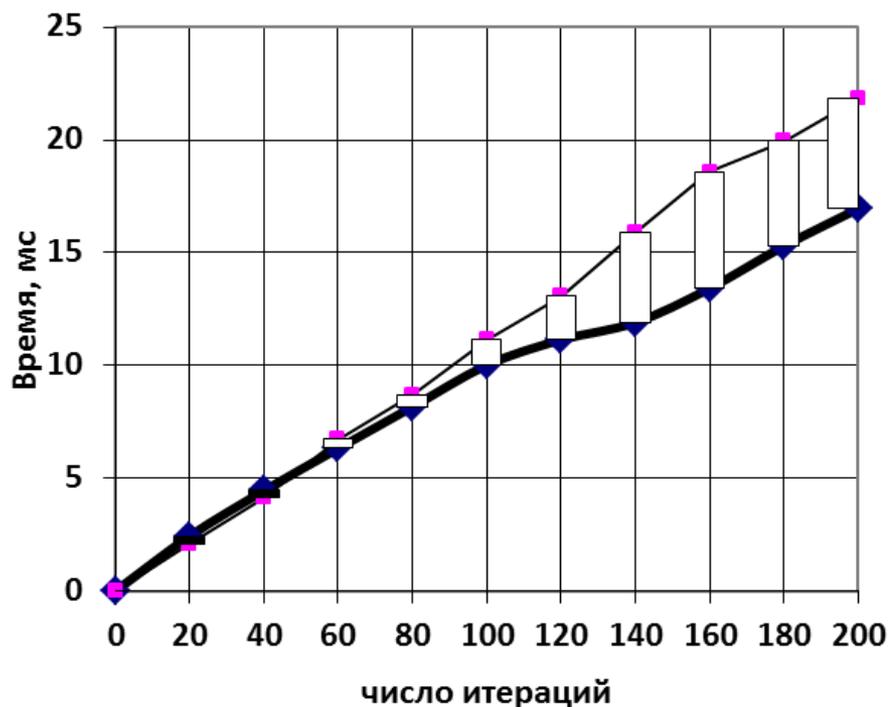
Ефект використання шаблонів



Товстою лінією позначено графік ТСА з використанням шаблону ($P_s = 0.999$), тонкою – без нього, прямокутники – різницю у роботі за часом. Експериментальні дослідження показали, що мурашиний алгоритм із використанням шаблону працює менше часу, ніж без нього.

Залежність часу роботи алгоритму від вершин

Ефект використання шаблонів



Товстою лінією позначено графік ВСА з використанням шаблону ($P_s = 0.999$), тонкою – без нього, прямокутники – різницю у роботі за часом. Експериментальні дослідження показали, що мурашиний алгоритм із використанням шаблону працює менше часу, ніж без нього.

Залежність часу роботи алгоритму від числа ітерацій

Локальний пошук

Ця модифікація полягає в використанні часткового перебору;

Для перебору використовується метод «гілок та кордонів»;

Перебір застосовується до випадкової частини рішення, тобто до підмаршруту;

Кількість вершин у підмаршруті незначна і становить близько 7-15.

«Випрямляч» - інтелектуальний блок, який приймає на вході безліч вершин і ребер між ними, і видає їх найкоротший обхід;

Можна виділити два види «випрямлячів»: проміжний та додатковий.

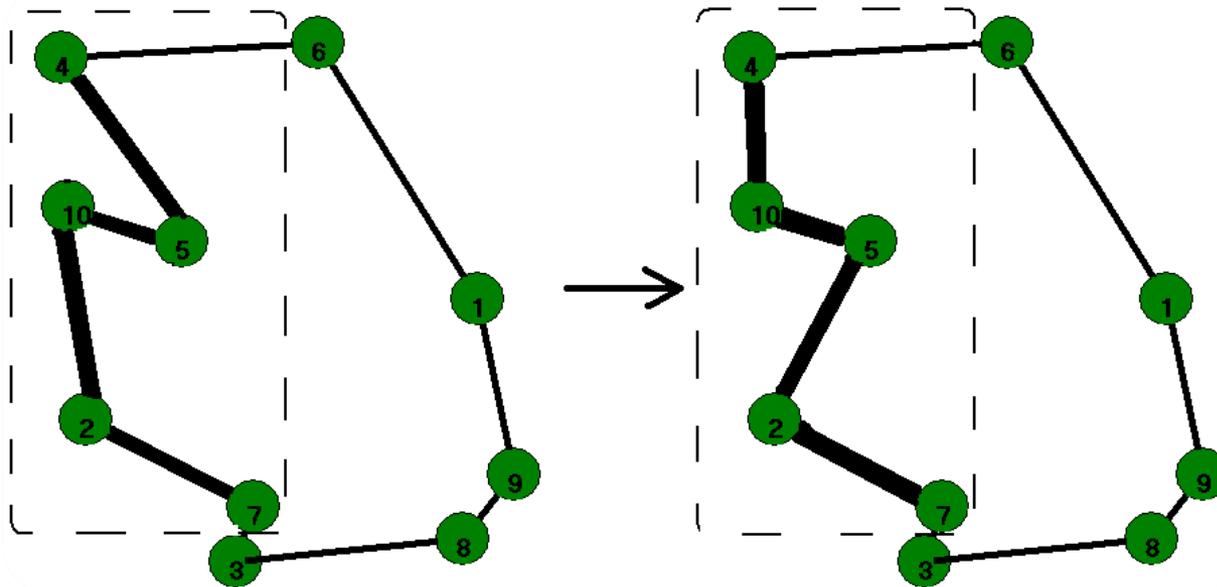
приклад

Нехай заданий наступний обхід вершин:

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

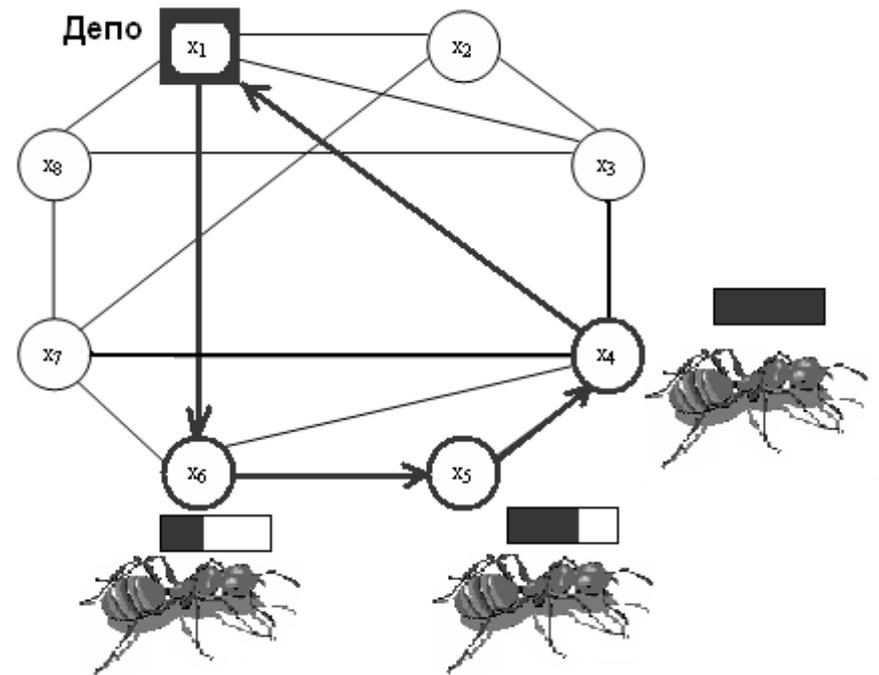
Застосуємо локальний пошук для підмаршруту розмірністю $l=5$, з 4 позиції по 8:

вершини (7,2,10,5,4) > маршрут (7-2-5-10-4)



Завдання маршрутизації автотранспорту

При збиранні їжі мураха намагається обходити всі точки (розташування їжі) найкоротшим шляхом. Мураха здатна «брати вантаж», обмеженої ваги. Набравши максимально можливий вантаж, мураха повертається до колонії і «розвантажується». Далі він повторює ці дії на не відвіданих місцях харчування, поки вона вся не буде зібрана.



Аналіз ТСА

- ТСА по М.Дориго залежить від кількості вершин n , розміру колонії m та числа ітерацій T (час життя колонії) - $O(T*m*n^2)$.
- Витікає, що ТСА залежить від параметрів α і β , внаслідок зведення в ступінь - операція, ТСА якого $O(\log(\alpha))$, де α - значення ступеня. Тоді ТСА визначається, як:

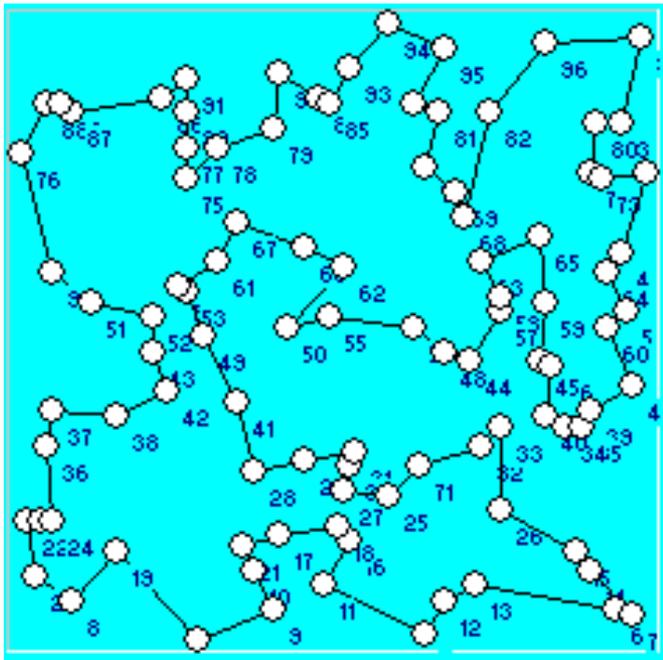
$$O(T*m*n^2 * (\log(\alpha) + \log(\beta)))$$

- Відмітимо, що η_{ij} і β є константою, отже η_{ij}^β - константа. Тоді для підвищення швидкодії η_{ij} зручно обчислювати як $\eta_{ij} = (1/D_{ij})^\beta$. Тоді ТСА:

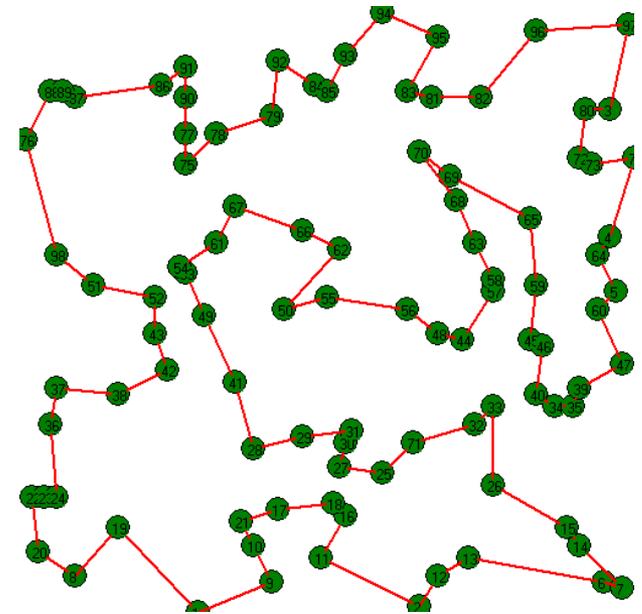
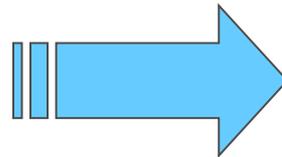
$$O(T*m*n^2 * \log(\alpha))$$



98 вершин



783,723



783,447

Бджолині алгоритми

Цей клас алгоритмів розроблявся у межах наукового напрямки «природні обчислення».

Дослідження у цій галузі почалися на початку ХХІ століття.

Авторами ідеї є Lučić P., Teodorović D.

Основна ідея – моделювання поведінки колонії бджіл.

Основні положення

Додатній зворотній зв'язок

- Бджоли виходячи з інформації, що отримана від іншої бджоли, починають літати до вказаного джерела нектару

Від'ємний зворотній зв'язок

- Бджола, виходячи з інформації, що отримана від інших бджіл, може вирішити, що знайдене нею джерело значно гірше в порівнянні з іншими знайденими джерелами

Нестійкість

- Бджоли-розвідники виконують випадковий пошук нових джерел ресурсів

Множинність взаємодії

- Інформація від джерела ресурсів, що знайдене однією бджолою, є доступною для усіх інших у вулику шляхом виконання так званого виляючого танцю

Властивості бджоли

i -та бджола здатна по танцю j -й визначити кількість нектару $F(h_j)$ у джерелі h_j

i -та бджола, знайшовши нове джерело h_i , може змінити роль на фуражиря або розвідника

i -та бджола здійснює випадковий пошук на ділянці p_i радіусом r

i -та бджола повідомляє колонії тільки про краще джерело h_i на ділянці p_i

Параметри бджолиного алгоритму

p – число фуражирів

l – число розвідників

$m=p+l$ – розмір колонії

k – кількість ділянок для фуружування

r – розмір ділянки

T – кількість ітерацій

Відносини об'єктів бджолиного алгоритму

Колонія

Фуражири

Розвідники

Бджола

Пошук
джерела
нектару

Танецок

Ділянка

Джерело

Джерело

Колонія складається з бджіл двох типів: фуражири та розвідники. Бджоли можуть здійснювати танці та шукати нові джерела нектару на різних ділянках.

Приклад поведінки бджоли

Повернувшись у вулик бджола або змінює роль, або вербує, інформуючи про знайдене джерело



Фуражир

- Вибір елітної ділянки
- Пошук
- Повернення до вулика

Розвідник

- Випадковий вибір ділянки
- Пошук
- Повернення до вулика

Вербовка

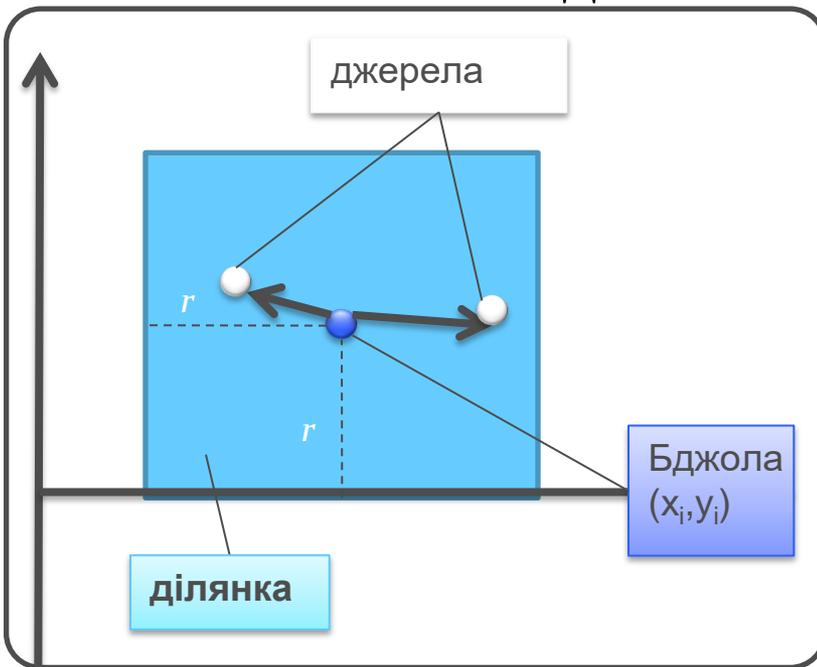
- Фуражировка на попередній ділянці

Пошук на ділянці

Реальна модель



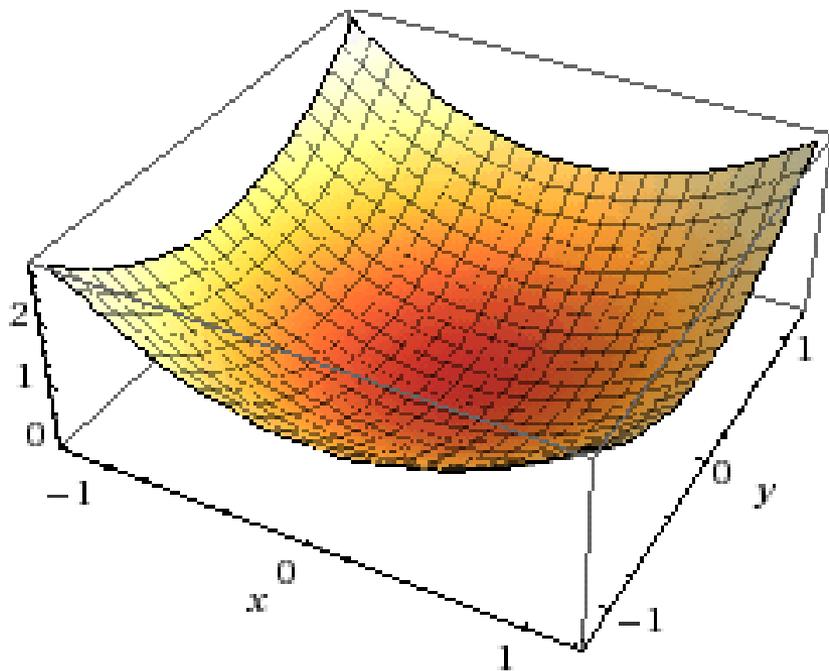
Математична модель



При вирішенні завдань знаходження глобальних екстремумів складних багатовимірних функцій межі ділянки визначаються як $[x_1-r; x_1+r][x_2-r; x_2+r] \dots [x_n-r; x_n+r]$, де x_i - координата бджоли (параметр функції), r - радіус (розмір) ділянки, n - кількість параметрів функції

Приклад поведінки колонії

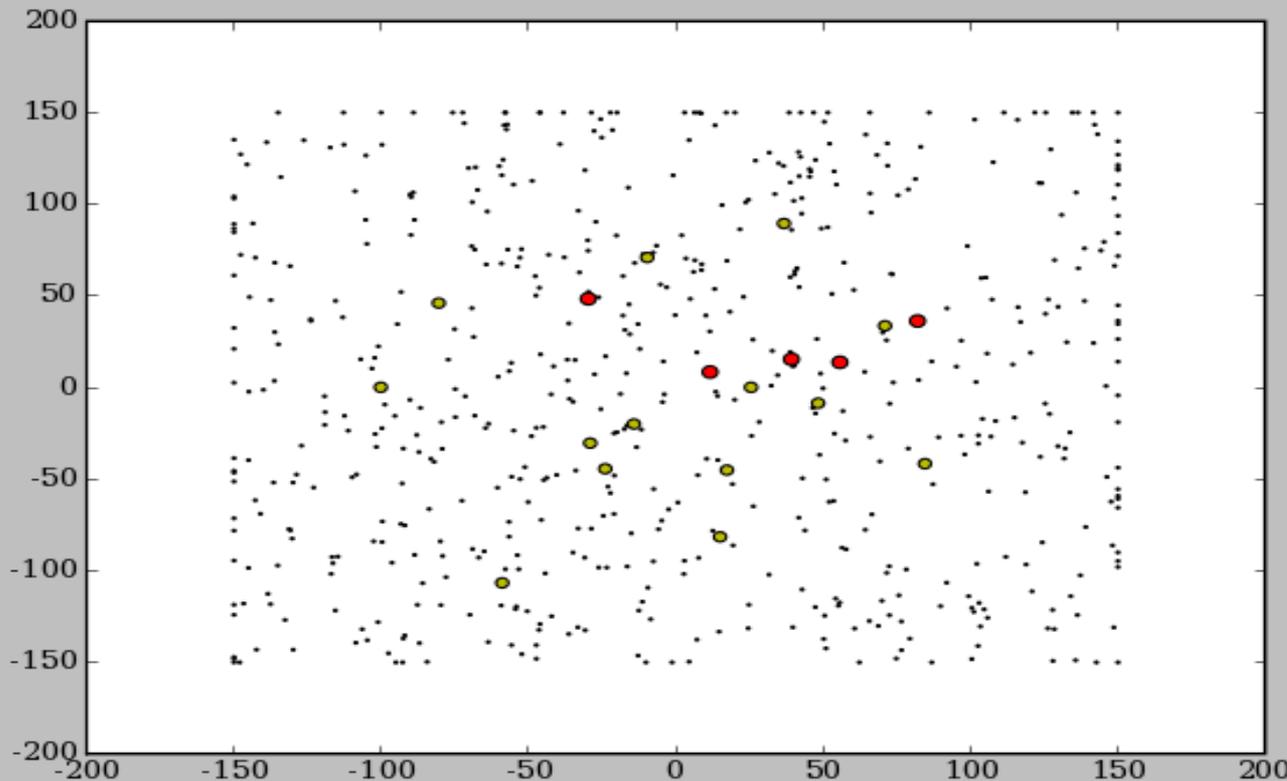
Цільова функція: $f(x, y) = x^2 + y^2$. Ціль: мінімізація ЦФ.



Рішення знаходиться
в координатах (0; 0)

Приклад поведінки колонії

Цільова функція: $f(x, y) = x^2 + y^2$. Ціль: мінімізація ЦФ.

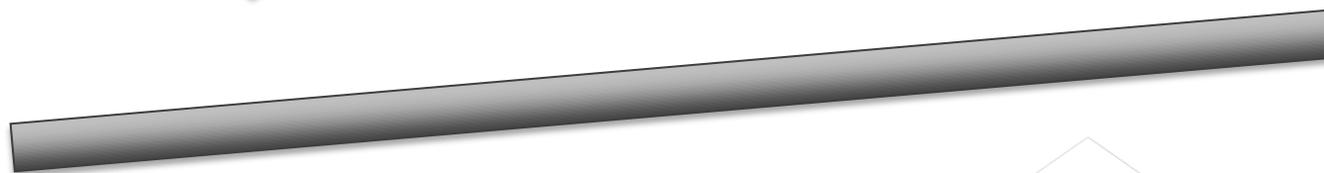


Бджоли поступово накопичуються навколо одного найкращого рішення. Червоні крапки - бджоли, які знайшли найкращі рішення. Зелені крапки - вибрані рішення. Чорні крапки - бджоли-розвідники

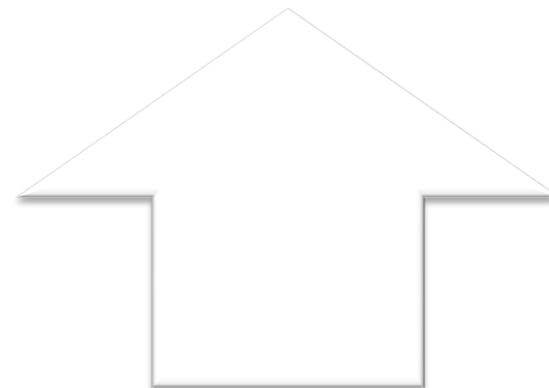
Використання бджолиних алгоритмів під час вирішення комбінаторних завдань



Бджолині алгоритми добре вирішують задачі знаходження глобальних екстремумів в складних багатомірних функціях



Даний алгоритм погано адаптований для вирішення класичних NP-складних задач (комівоязера, упаковки, ...). Основна проблема – її інтерпретація для цих задач



Кодування рішення

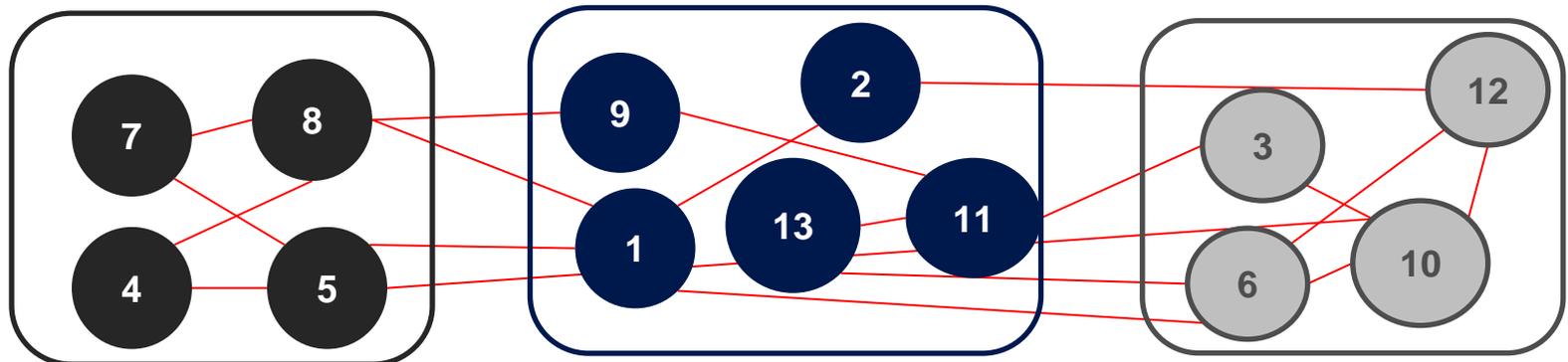
Потужності підграфів M_i :

i	1	2	3
$ M_i $	4	5	4

Рішення подається у вигляді перестановки вершин A_i , причому перші M_1 вершин відносяться до першого підграфу, наступні M_2 вершин відносяться до 2 підграфу і т.д.:

i	1	2	3	4	5	6	7	8	9	10	11	12	13
	підграф№1				підграф№2					підграф№3			
A_i	4	5	8	7	13	9	1	2	11	3	12	10	6

$F=8$



Інтерпретація бджолиного алгоритму на вирішення завдання розбиття графа

Простір пошуку нектару буде являти собою припустимі перестановки A з n чисел, а його розмірність дорівнює $n!$.

Кількість нектару в джерелі A обернено пропорційно ЦФ $F(A)$.

Джерело нектау визначається одним рішенням з простору пошуку та представляється у вигляді деякої перестановки

Два джерела (A, B) знаходяться на одній ділянці, якщо відстань $L(A, B) \leq r$, где r – заданий параметр перестановок B і A

Відстань L між перестановками A і B визначається, як відстань Хемінга та доравнює числу відмінностей в позиціях між ними

Приклад обчислення відстані

$$r = 4$$

i	1	2	3	4	5	6	7	8	9	10	11	12	13
	підграф№1				підграф№2					підграф№3			
A_i	4	5	8	7	13	9	1	2	11	3	12	10	6
B_i	4	11	8	7	13	9	1	2	3	5	12	10	6
$A_i \neq B_i$	0	1	0	0	0	0	0	0	1	1	0	0	0
$L(A,B)$	3												

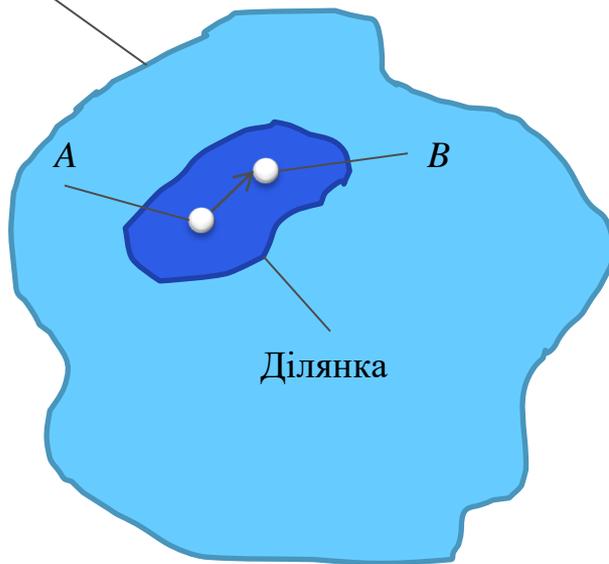
$$L(A,B) < r$$

Висновок: джерела нектарів A і B перебувають в одній ділянці, оскільки розташовуються у безпосередній близькості друг від друга просторі пошуку.

Переміщення з одного джерела нектару до іншого

Переміщення з джерела нектару A в джерело B однієї ділянки здійснюється з допомогою оператора обміну. Він полягає в обміні значеннями двох позицій у векторі рішення.

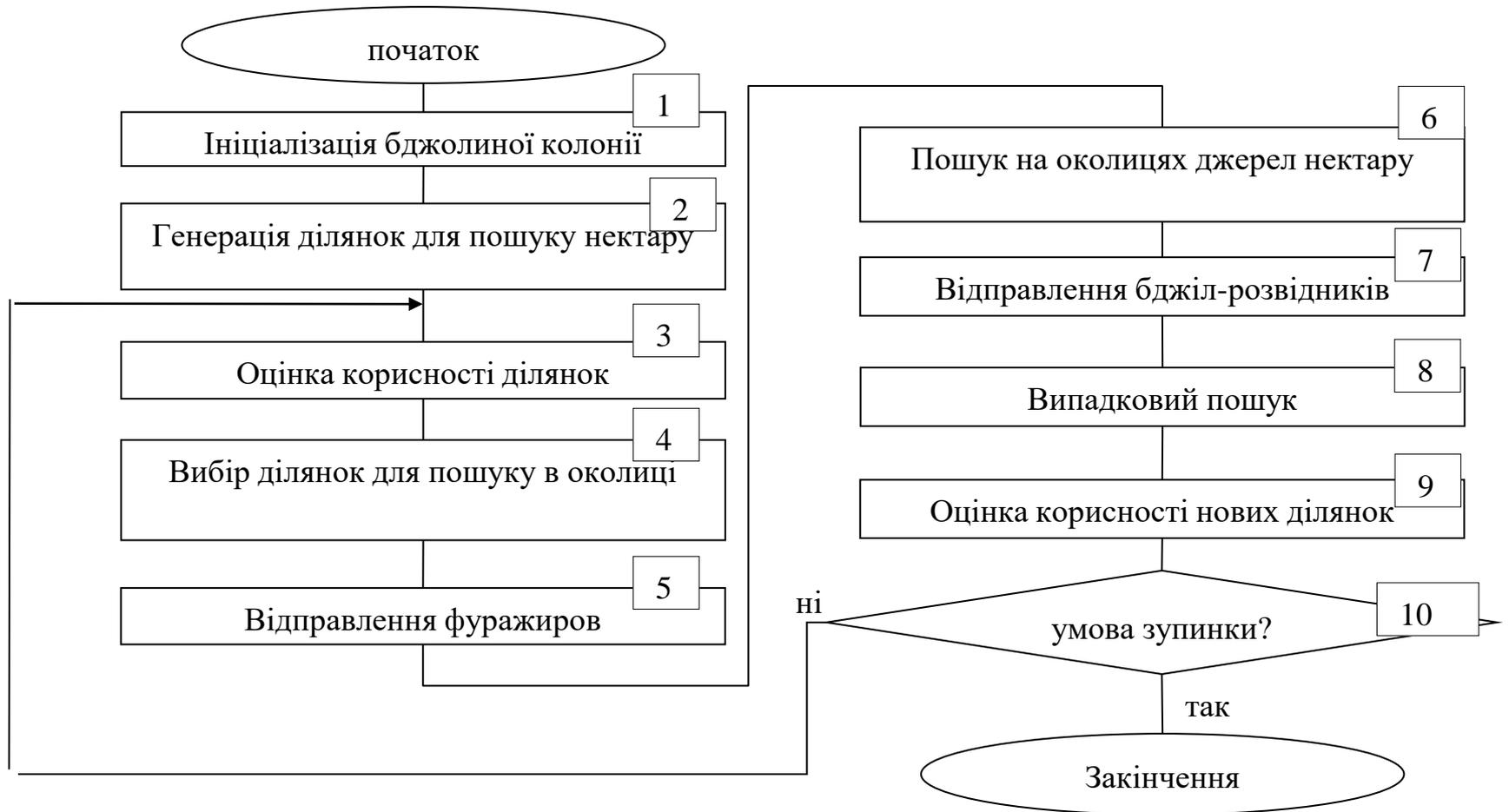
Простір пошуку



i	1	2	3	4	5	6
A_i	4	2	5	6	3	1
B_i	4	6	5	2	3	1

Використання простої операції обміну значеннями дозволяє моделювати пошук бджоли в рамках обмеженої ділянки.

Структурна схема бджолиного алгоритму



Тимчасова складність алгоритму (ТСА)

Часова складність бджолиного алгоритму $O(T*m*r*n^2)$, де

T – кількість ітерацій

m – розмір колонії

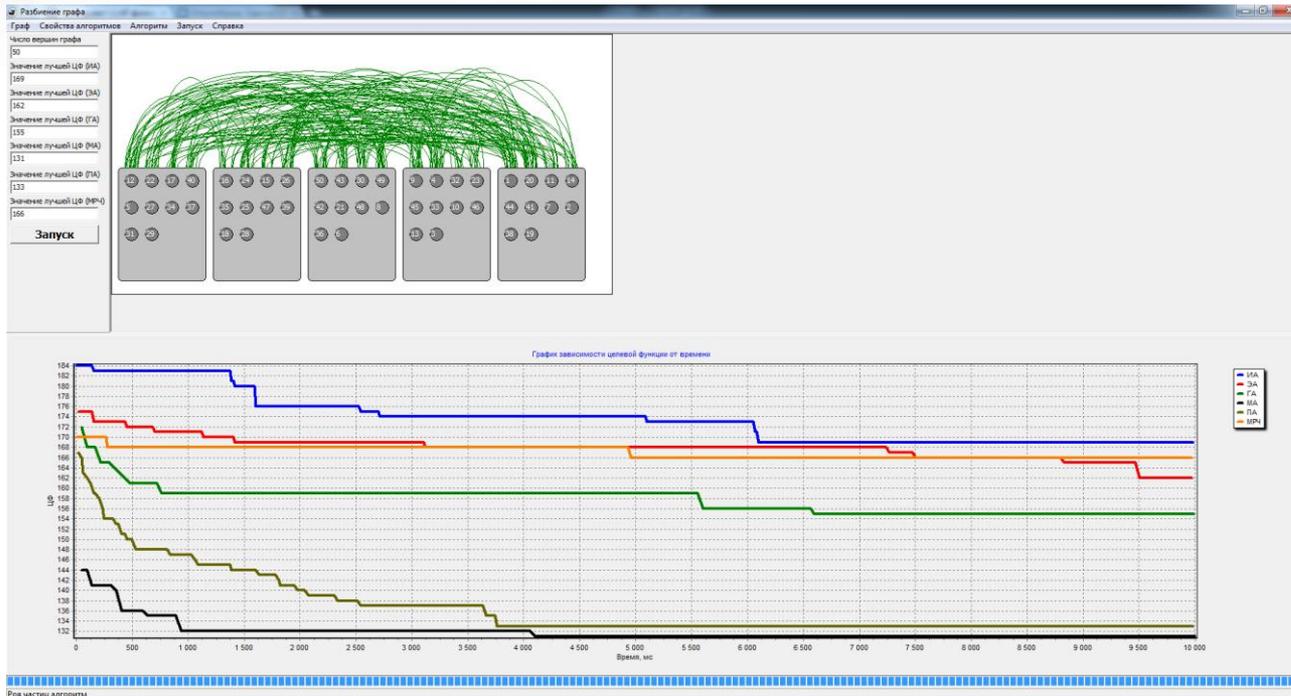
r – розмір ділянки

n – кількість вершин

Якщо прийняти параметри B A за константні значення, маємо ТСА:

$O(n^2)$

Експериментальні дослідження



Для графів невеликої розмірності (до 80 вершин) найкраще рішення виходило за допомогою мурашиного алгоритму, в інших випадках – за допомогою бджолиного.

Реалізовані алгоритми

Ітераційний

Еволюційний

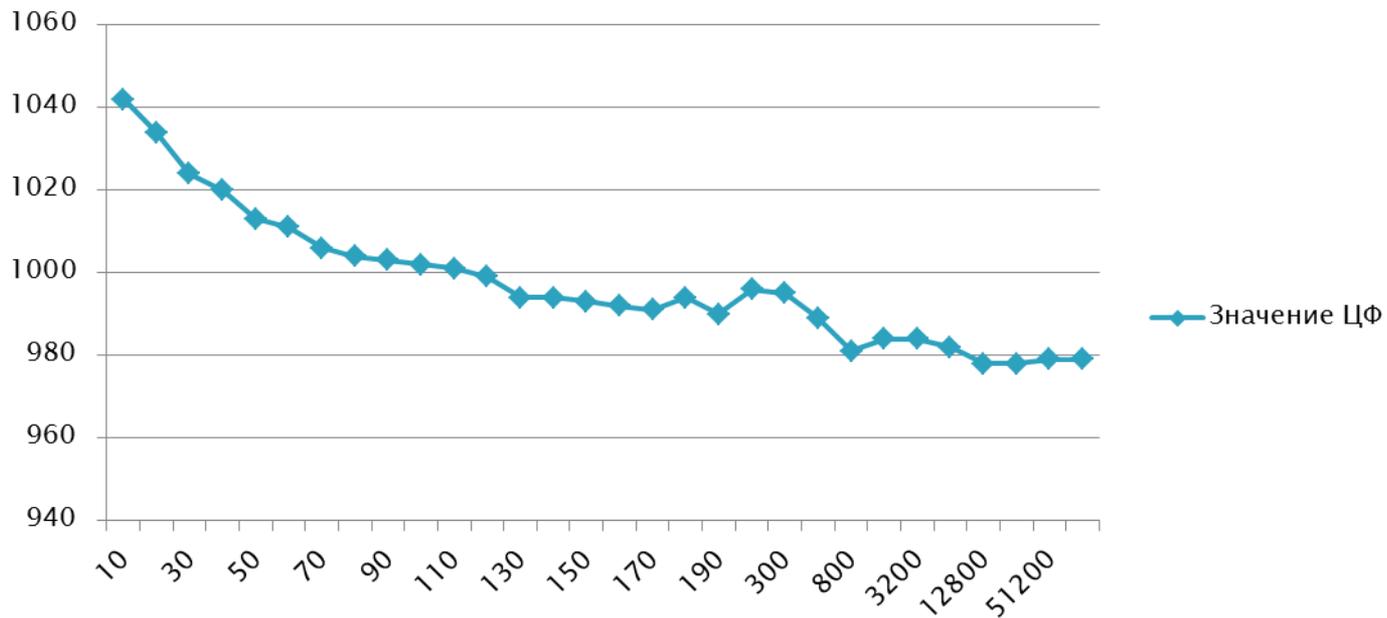
Генетичний

Мурашиний

Бджолиний

Експериментальні дослідження

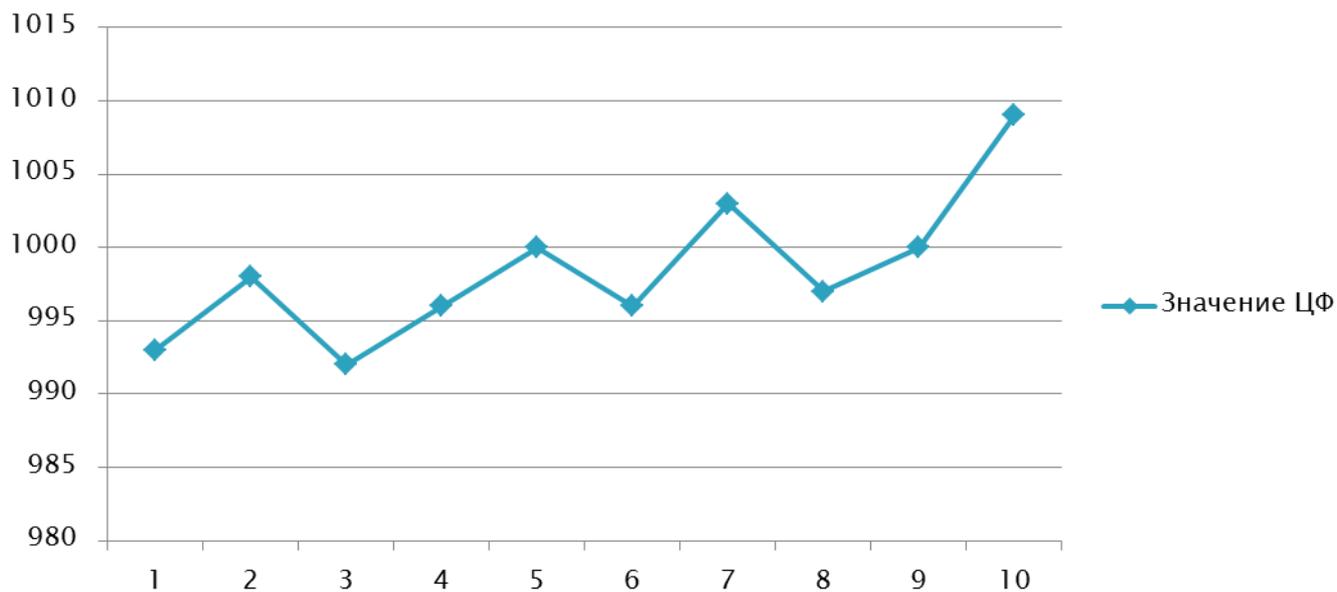
Значение ЦФ



Графік залежності ЦФ від числа фуражиров. Розмірність графа – 125 вершин

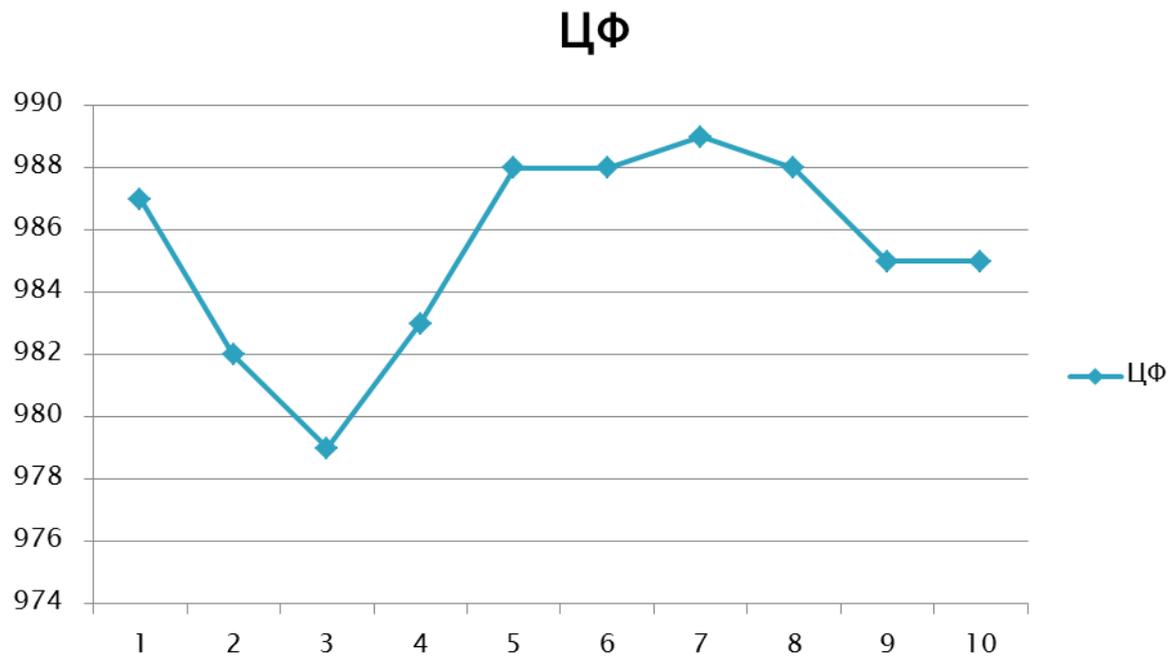
Експериментальні дослідження

Значение ЦФ



Графік залежності ЦФ від обсягу ділянок. Розмірність графа – 125 вершин

Експериментальні дослідження



Залежність ЦФ від кількості досліджуваних джерел нектару. Розмірність графа – 125
вершин

Експериментальні дослідження

<u>Развед.</u> фуражиры	20	40	60	80	100	150	200	300
5	1039	1011	1014	1003	1005	992	992	992
10	1029	1018	1011	1001	1001	999	996	993
15	1036	1016	1010	1004	999	993	993	993
20	1029	1017	1013	1013	1007	999	988	984
25	1024	1019	1018	1011	1005	996	977	990
30	1040	1019	1011	1005	994	994	994	999

Залежність ЦФ від числа фуражиров та розвідників. Розмірність графа – 125 вершин

Експериментальні дослідження

Кількість вершин	Кількість підграфів	Результат МА	Результат БА	
50	5	154	156	
100	10	815	824	
500	5	19681	19401	
2000	20	16696	19129	
2000	2	-	9893	

Порівняння алгоритмів розбиття для тестів різної розмірності