

Лабораторна робота №3

Тема: ‘Застосування генетичних алгоритмів для вирішення задач оптимізації

1

Мета роботи: навчитися розв’язувати задачі оптимізації за допомогою генетичного алгоритму, навчитися програмно реалізовувати генетичний алгоритм.

Генетичний алгоритм – потужний інструмент для вирішення складних задач. Він знайшов своє застосування серед задач оптимізації, інженерії, штучного інтелекту. В основу алгоритму покладено принципи біології та генетики – створення популяції індивідів, кожен з яких представляється у вигляді хромосоми. Своїм існуванням генетичні алгоритми завдячують спробам наслідування природних процесів, які відбуваються серед живих організмів – селекції та еволюції популяції живих істот.

Ідею генетичного алгоритму у 1975 році запропонував Джон Холланд (J.H.Holland). Він припустив, що складений відповідним чином алгоритм біологічної еволюції може бути покладений в основу розв’язання складних проблем.

Представлення об’єктів.

Біологічно кожен організм може бути представлений своїм фенотипом (котрий визначає чим є об’єкт у реальному світі) та генотипом (котрий містить інформацію про цей об’єкт на рівні хромосомного набору). Кожен ген, тобто елемент інформації генотипу, містить своє відображення у фенотипі. Генетичні алгоритми моделюють природну еволюцію та використовуються здебільшого для розв’язання задач оптимізації. Генетичний алгоритм перейняв термінологію з генетики: хромосома (елемент популяції), популяція (випадковий набір допустимих розв’язків), алель (значення гена), ген (елемент хромосоми, двійковий розряд), локус (позиція гена у хромосомі), алель (значення гена), покоління (результат наступної ітерації роботи алгоритму), фенотип (числове значення хромосоми).

Генетичний алгоритм.

Класичний генетичний алгоритм зображений на рис.1

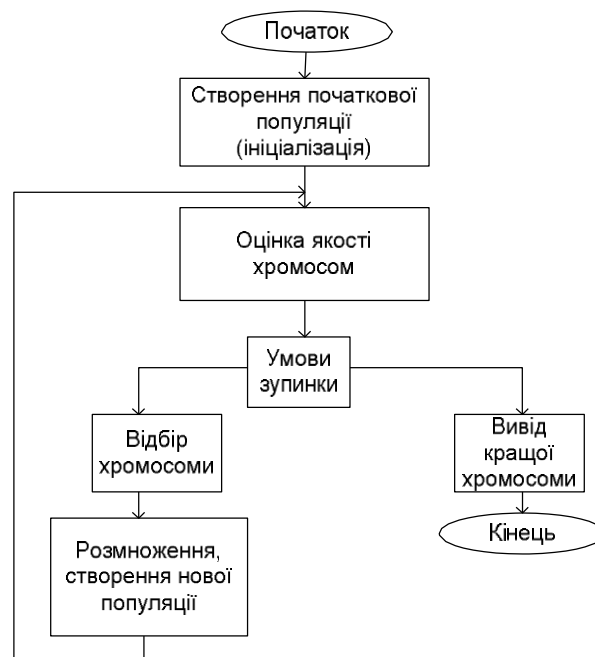


Рис.1 Блок-схема роботи генетичного алгоритму.

1. *Створення початкової популяції.* На першому етапі роботи створюємо певну початкову популяцію. Навіть якщо вона буде неконкурентоспроможною алгоритм сам перетворить її у «живу» популяцію. З популяції випадково вибираємо N хромосом у вигляді бітових ланцюжків довжиною L . Вводимо поняття номеру популяції K (для початкової $K=0$).
2. На даному етапі необхідно обчислити значення функції відповідності кожної хромосоми. Чим вище значення функції, тим вища якість хромосоми.
3. *Умови зупинки алгоритму* . Алгоритм припиняє свою роботу у наступних випадках: коли досягнуто шуканого значення(із заданою точністю); коли подальша робота алгоритму не може покращити отриманий результат; коли був вичерпаний заданий ліміт часу чи кількості ітерацій. У випадку зупинки алгоритму виводимо найкращу хромосому.
4. *Відбір хромосоми.* Обчисливши значення функції відповідності відбираємо нащадків хромосом із найбільшим значенням. Зазвичай відбір проводиться методом рулетки. Результатом роботи операції є батьківський пул з N хромосом. Відповідно його розмір збігається із розміром початкової популяції.
5. *Створення нової популяції.* Створення нової популяції досягається застосуванням генетичних операторів. За їх допомогою ми утворюємо нову популяцію нащадків. В класичному алгоритмі використовують два генетичних оператори – схрещування та мутація. При цьому схрещування відбувається завжди, а мутація доволі рідко. При реалізації генетичного алгоритму задається ймовірність схрещування P_c ($0,5 \leq P_c \leq 1$) та ймовірність мутації P_m ($0 \leq P_m \leq 0,5$) .

Сутність методу рулетки для селекції хромосом (roulette wheel selection)

Метод моделює випадковий вибір, за аналогією рулетки в казино. Рулетка ділиться на частини, кожна з яких відповідає кожній хромосомі. Розмір секції рулетки є пропорційним до значення функції відповідності цієї хромосоми. Чим більше значення функції відповідності, тим більша частина кола відповідає йому на рулетці і навпаки. Математично це можна зобразити так:

$$v(ch_i) = p_s(ch_i) * 100\% ,$$

де ch_i - хромосома, $v(ch_i)$ - відрізок кола, який становить частину цілого кола, $i = 1, 2, \dots, N$, N - кількість елементів популяції, $p_s(ch_i)$ -- ймовірність селекції

$$p_s(ch_i) = \frac{F(ch_i)}{\sum_{i=1}^N F(ch_i)} ,$$

де $F(ch_i)$ -- значення функції відповідності для хромосоми ch_i , хромосоми ch_i .

Для вибору хромосом проводиться обертання рулетки. Ймовірність вибору певної хромосоми є тим більшою, чим більше значення її функції відповідності.

Вибір способу кодування

Для використання генетичного алгоритму хромосому необхідно представити у відповідному числовому значенні. Найпростіший варіант – бітове представлення. Бітове представлення хромосоми є досить зручне та просте. Проте, проблема такого кодування полягає у тому, що сусідні числа послідовності відрізняються у значеннях кількох бітів. Наприклад, числа 7 та 8

у бітовому представленні відрізняються у 4-х позиціях, а це відповідно затрудняє функціонування генетичного алгоритму. Для вирішення цієї проблеми краще використовувати кодування, при якому сусідні числа відрізняються меншою кількістю позицій, а в ідеалі значенням одного біта. Таким є код Грея, що представлений нижче у таблиці:

Таблиця 1. Відповідність десяткових кодів і кодів Грея.

Ціле	Двійковий код	Шістнадцятковий код	Код Грея
0	0000	0h	0000
1	0001	1h	0001
2	0010	2h	0011
3	0011	3h	0010
4	0100	4h	0110
5	0101	5h	0111
6	0110	6h	0101
7	0111	7h	0100
8	1000	8h	1100
9	1001	9h	1101
10	1010	Ah	1111
11	1011	Bh	1110
12	1100	Ch	1010
13	1101	Dh	1011
14	1110	Eh	1001
15	1111	Fh	1000

Отже, при кодуванні цілочисельного значення ми розбиваємо його на тетради і кожен тетраду перетворюємо за кодом Грея. Задача декодування генів є тривіальною.

При кодуванні чисел із плаваючою крапкою можна використати наступний алгоритм.

1. Розбивають весь інтервал допустимих значень фенотипу на частини із заданою точністю;
2. Розглядають значення гена як цілочисельне значення, визначають номер інтервалу (використовуючи код Грея);
3. В якості значення параметру приймають число, що є серединою цього інтервалу.

Наприклад, нехай простір пошуку лежить у діапазоні $[0,1]$. При кодуванні використовувалось розбивання простору на 256 інтервалів. Для кодування їх номера нам потрібно буде 8 біт. Припустимо, що значення гена є 00100101bG (літера G показує, що використовується кодування Грея). Спочатку, використовуючи код Грея, знайдемо відповідний йому номер інтервалу – 25hG-> 36hG-> 54hG. Тепер отримуємо інтервал $[0,207031, 0,210937]$. відповідно значення нашого параметру буде $(0,207031+0,210937) / 2 = 0,2089843$

При кодуванні нечислових даних потрібно попередньо перетворити їх у числа.

Оператор схрещування

Для використання операції схрещування необхідно вибрати пару хромосом з батьківського пулу, які випадково групуються у пари за заданою ймовірністю схрещування P_c . Алгоритм роботи оператора приймає вигляд:

1. з популяції вибирається дві особини, які і будуть батьками;
2. випадковим чином визначається точка розриву;

3. нащадок визначається як конкатенація частини першого та другого батьків. Наприклад, хромосома_1: 000000 , хромосома_2: 111111. Випадковим чином проводимо лінію схрещування (випадково генерується число, що менше довжини хромосоми). Нехай лінія схрещування проходить після 3-го біта, тоді:

Хромосома_1: 000000 \gg 000 | 111 , Результуюча хромосома : 000111

Хромосома_2: 111111 \gg 111 | 000 , Результуюча хромосома : 111000

Схрещування можна проводити за різними масками. Розглянемо різні схрещування на прикладах:



У „двоточковому схрещуванні”, результат утворюється заміною однієї частини батька іншою. Іншими словами, „маска схрещування” починається з n_0 0-ів, n_1 1, та завершується необхідною кількістю 0-ів. Кожен раз коли застосовується „двоточкове схрещування”, маска генерується за допомогою вибраних випадковим чином n_0 та n_1 . У прикладі результат утворюється за допомогою маски для якої n_0 2 та n_1 5. Знову результат утворюється змінюючи місцями батьків при застосування оператора „схрещування”.

„Універсальне схрещування” комбінує біти двох батьків, як показано у прикладі. В цьому випадку „маска схрещування” генерується, як випадкова бітова стрічка у якої кожен біт вибирається випадково і не залежить від інших.

Оператор мутації

За заданою ймовірністю мутації P_m виконується зміна значення деякого гена на протилежне. Наприклад, маючи ген 0000 у результаті мутації отримаємо ген 0001. Тобто, результатом

роботи оператора є зміна одного біта на протилежне йому значення. Розглянемо ще один приклад:

Точкова мутація

11101001000 —————> 11101011000

ПРИКЛАД 1

Для прикладу розв'яжемо задачу максимізації функції $f(x) = x^2$ на відрізку $[0;31]$. Для кодування виберемо п'ятибітовий двійковий код. Виберемо початкову популяцію ($k=0$), яка складатиметься із 4 бітових ланцюжків ($N=4$). Таку комбінацію можна отримати двома шляхами. Перший полягає у випадковому виборі 4 цифр із діапазону від 0 до 31. Інший полягає у підкиданні 20 разів монети (4 комбінації * 5 бітів).

Нехай першим шляхом ми отримали наступну популяцію: 15(01111b), 27(11011b), 10(01010b), 6(00110b). В даному випадку бітове представлення чисел називається популяцією, а числа в десятковому форматі – фенотипом. Кожну із утворених хромосом підставляємо у задану функцію $f(x)=x^2$. В результаті отримаємо $15^2=225$, $27^2=729$, $10^2=100$, $6^2=36$ - це будуть значення цільової функції. Тепер цю популяцію необхідно розложити на рулетці для проведення вибору хромосом. Для цього знайдемо суму значень цільової функції $225+729+100+36=1090$. Обраховуємо частину кола, яку займе хромосома(наприклад, перший бітовий ланцюжок має значення 225 при сумі 1090, що становить 21% чи 0,21) $225/1090=0,21$; $729/1090=0,67$; $100/1090=0,09$; $36/1090=0,03$. Як видно, сума отриманих значень повинна становити одиницю, перевіримо $0,21+0,67+0,09+0,03=1$. Порахуємо очікувану кількість копій кожної хромосоми. Для цього потрібно кожну із частин кола помножити на кількість хромосом, а саме $0,21*4=0,84$; $0,67*4=2,68$; $0,09*4=0,36$; $0,03*4=0,12$. Рулетка прийме вигляд

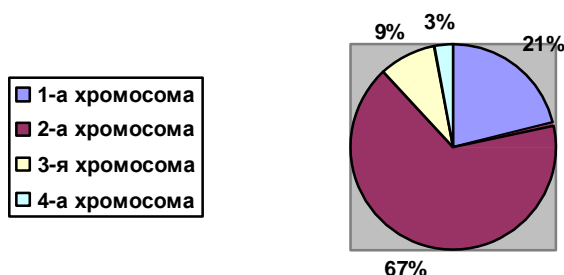


Рис.2 Рулетка, побудована для відбору хромосом

Оскільки у нас є 4 хромосоми, то рулетку будемо обертати 4 рази (щоб кожній хромосомі дати шанс на виживання). Нехай ми отримали наступний батьківський пул – одна копія першого та третього ланцюжків, дві других і жодного четвертого. Отримані результати і виправдали сподівання – найкращі (відповідно до значення цільової функції) хромосоми збільшили свою популяцію, середні залишились на тому ж рівні, а найгірші відмерли.

Отримавши батьківський пул необхідно застосувати генетичні оператори – схрещування та мутація. Для схрещування кожній хромосомі необхідно підібрати пару. Для цього випадково

генеруємо числа від 1 до 4 (відповідно до кількості хромосом). Нехай ми отримали наступні результати:

Таблиця 2. Вибір партнера для схрещування

№ хром.	Отриманий батьківський пул	Номер партнера
1	01111	2
2	11011	1
3	11011	4
4	01010	3

Вибравши партнера, необхідно вибрати пункт поділу хромосоми. Лінія схрещування будується підкиданням монети 2 рази. Як результат, отримаємо двійковий код 00, 01, 10, 11, що в свою чергу і буде вказувати на 2, 3, 4 – ту позиції. Це відбудеться при ймовірності схрещування $P_c=1$. Нехай у результаті підкидання монети ми отримали такі дані

Таблиця 3. Проведення лінії схрещування

№ хром.	Отриманий батьківський пул	Лінія схрещування
1	01111	3
2	11011	3
3	11011	4
4	01010	4

Отже, проводимо схрещування. 1-ша хромосома схрещується з 2-ю і лінія схрещування проходить через 3 позицію, 3 з 4 і лінія проходить через 4 позицію:

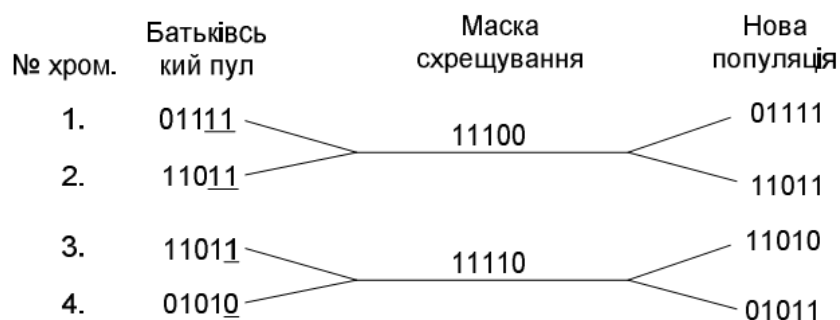


Рис.3 Застосування операції схрещування

Далі виконуємо операцію мутації. Операція виконується для кожної алелі окремо. Нехай ймовірність мутації становить $P_m=0,003$, то маючи 20 бітів, зміниться $20 \cdot 0,003 = 0,06$ біти. Це означає, що жоден з бітів не зазнає мутації. Але якщо ймовірність становить $P_m=0,3$ то зміниться $20 \cdot 0,3 = 6$ бітів. Тому випадковим чином генеруємо 4 номери позиції зміни позиції хромосоми і їх значення міняємо на протилежні. Наприклад, якщо випали позиції 3, 6, 18, 9, то хромосоми приймуть такі значення

1. 01011
2. 01001
3. 11010
4. 01111

Після закінчення операцій нове покоління піддається оцінюванню. Для цього обчислюємо значення функції.

Таблиця 4. Результат застосування генетичних операторів

№ хром.	Нова популяція	Фенотип	Значення функції
1	01111	15	225
2	11011	27	729
3	11010	26	676
4	01011	11	121

Отже, лише після першої генерації середнє значення функції відповідності збільшилось із 272 до 437. Найкраща хромосома з першого покоління отримала дві копії завдяки високій оцінці відповідності.

Алгоритм можна використовувати і для мінімізації функції. Зазначимо, що без втрати загальності, можна розв'язувати тільки задачі максимізації. Якщо потрібно функцію мінімізувати, то відповідно переходимо до максимізації функції g , де $g = -f$. При розв'язку задач максимізації функції багатьох змінних суть алгоритму не змінюється. Кожна

хромосома зображається бітовим кодом довжиною $m = \sum_{i=1}^k m_i$, де перші m_1 бітів відповідають змінній x_1 , другі m_2 бітів – змінній x_2 і так до останніх m_i бітів, що відповідають змінній x_k .

Для вибору початкової популяції можна згенерувати випадково біт за бітом N хромосом. При наявності інформації про розміщення оптимумів можна використати її для визначення початкових розв'язків. Решта алгоритму є очевидною.

При розв'язку задачі із заданою точністю, наприклад із 4 значущими десятковими цифрами для кожної змінної, кожний відрізок D , ділиться на $(b_i - a_i) \cdot 10^4$ рівних відрізків. Нехай m_i таке найменше ціле число, що $(b_i - a_i) \cdot 10^4 \leq 2^{m_i} - 1$. Тоді зображення, в якому кожна змінна x_i закодована у вигляді бітового ланцюжка довжини m_i буде задовольняти вимоги точності.

Значення цього ланцюжка можна представити у десятковій системі числення за допомогою наступного представлення $x_i = a_i + dec(ch) * (b_i - a_i) / (2^{mi} - 1)$, де $dec(ch)$ – десяткове представлення бітового ланцюжка ch .

Завдання для виконання лабораторної роботи

Максимізувати та мінімізувати функцію однієї змінної. Обчислення провести з точністю до 10^{-n} , де $n=0,1,2 \dots$ Варіанти завдань.

№ варіанту	Функція
1	$x^3 + 2\sqrt{x}, x \in [-10;15]$
2	$-7x + x^7 - \sqrt{x}, x \in [0;50]$
3	$3y - y^2 - y^5, y \in [-10;0]$
4	$\frac{x^2}{5} - (x+5)^4, x \in [0;30]$
5	$\frac{z}{7} + z^2 + z^3, z \in (20;40)$
6	$x^3 - 35 + x, x \in (-15;15]$
7	$(x+3)^3 - 7, x \in [0;60]$
8	$y - (5 + y^2)^3, y \in [10;20]$
9	$\frac{y}{(3+y)^2} - 7, y \in [-10;10), y \neq -3$
10	$\frac{x-3}{x+5} - (x+3)(x-5), x \in [15;45], x \neq -5$
11	$x^2 + (x^3 - 7)(x^2 + 5), x \in [-10;0]$
12	$3x + 2x^2 - 7\sqrt{x}, x \in [0;15]$
13	$\frac{\sqrt{x}}{5} + (5 + \sqrt{x}), x \in [14;73)$
14	$\frac{z-5}{\sqrt{z}} - 7z, z \in (0;15)$
15	$y^7 - y^5 + 5\sqrt{y}, y \in (5;20)$
16	$-100(x^2 - 1)^2 - (1 - x^2)^3, x \in [0;15]$
17	$-(1 - x^2)^2 - \sqrt{x^2 - 2x}, x \in [-10;13]$
18	$\sqrt{\frac{x-5}{x+7}} + x^2, x \in [-15;15], x \neq -7$
19	$(x^3 + x^7)^3 - (x^2 - x)^6, x \in [0;30]$
20	$x^6 - x^4 + \sqrt{x^6 - x^4}, x \in [-50;-10]$
21	$x^3 + 7\sqrt{x} - \sqrt{x^3 - 7\sqrt{x}}, x \in [15;24]$
22	$\frac{x^7 - x^3}{x^4 - 5} \sqrt{3 - x^6}, x \in (1;15]$
23	$\frac{x^3 - x^{2x} - x}{\sqrt[3]{x-1}} \sqrt{x+6}, x \in [0;30), x \neq 1$

24	$30(x^3 - 7)^4 - (60 + x^2)^2, x \in (0;30)$
25	$\frac{17x^2 - 34x^3}{\sqrt{x^2 - 2x}}, x \in [-30;60], x \neq 0$
26	$(5x - x^2)^2 - \sqrt{x^2 - x^3}, x \in (0;100)$
27	$(x^8 - x^4)^2 - 5x^2, x \in [0;70]$
28	$x - 5\sqrt[3]{x^2 - x}, x \in [-7;9]$
29	$10(-x^3 - x^2) + x^6, x \in (0;47]$
30	$-10 - x^2 - x^3 + (x^4 + x^3)^2 \sqrt{x - 5}, x \in [-30;30]$

Вимоги до змісту роботи

Написати програму для реалізації генетичного алгоритму для розв'язування задач. Інтерфейс програми повинен дозволяти вводити такі дані: точність обчислення, кількість хромосом, вибір умови зупинки роботи алгоритму – після досягнення заданої точності або після досягнення кількості ітерацій, ймовірність схрещування, ймовірність мутації. Результатом роботи програми повинне бути значення всіх індивідумів у популяції, фенотип, значення функції відповідності для кожного індивідуума, середнє значення функції відповідності (див. приклад).

При захисті лабораторної роботи студент повинен знати генетичний алгоритм, вміти зробити необхідні зміни у тексті програми, пояснити зміст отриманих результатів.

Вимоги до оформлення звіту

Звіт повинен містити такі розділи:

1. Титульний аркуш із вказанням номером та назвою лабораторної роботи, особа що прийняла та виконала роботу;
2. Мета роботи;
3. Короткі теоретичні відомості;
4. Постановка задачі;
5. Алгоритм розв'язку задачі;
6. Інтерфейс програми із зазначеним розтлумаченням введених даних;
7. Результати роботи програми із розтлумаченням отриманих результатів;
8. Висновок;
9. У додатку помістити текст програми.

Контрольні запитання

1. Застосування генетичного алгоритму;
2. Чим відрізняється код Грея від звичайного бітового кодування?
3. Які основні кроки у класичному генетичному алгоритмі?
4. Які основні кроки методу рулетки?
5. Оператори схрещування та мутації?
6. Як застосувати генетичний алгоритм для знаходження мінімізації функції із заданою точністю?