1. АНАЛІЗ РОБОТИ ГЕНЕТИЧНИХ АЛГОРИТМІВ НА ТЕСТОВИХ ПРИКЛАДАХ

4.1 **Перевірка коректності роботи генетичних операторів**

Перед тим, як перевіряти роботу генетичних алгоритмів багатокритеріальної оптимізації, необхідно перевірити роботу загального генетичного алгоритму та усіх генетичних операторів, що приймають участь у його роботі.

Основні етапи роботи класичного генетичного алгоритму такі:

1. ініціалізація, або вибір вихідної популяції хромосом;
2. оцінка пристосованості хромосом в популяції;
3. перевірка умови зупинки алгоритму;
4. селекція хромосом;
5. застосування генетичних операторів;
6. формування нової популяції;
7. вибір «найкращої» хромосоми.

Для перевірки роботи загального генетичного алгоритму були використані функції для яких відоме точне значення мінімуму.

Перша функція, за якою перевірялася робота генетичного алгоритму - це сферична модель, яка вираховується за формулою

Ця функція вважаться досить простою для оптимізації, та має мінімум, що дорівнює 0 у точці *xi* = (0, 0).

Графік функції наведений на рисунку 4.1.

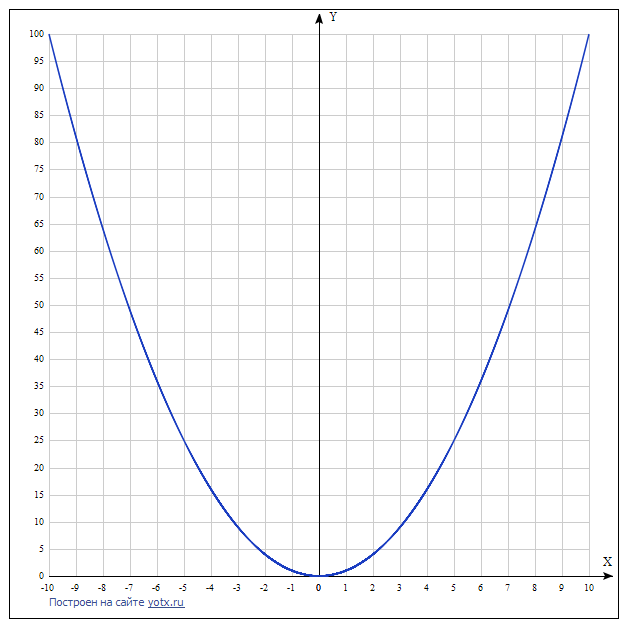


Рисунок 4.1 – Сферична модель для однієї змінної

Перевірка здійснювалася для популяції розміром в 100 особин, з вірогідністю мутації 0.5 та умовою завершення роботи алгоритму після 100 ітерацій. Результати експерименту наведені у таблиці 4.1.

Таблиця 4.1 – Результати роботи алгоритму за першою функцією

|  |  |  |  |
| --- | --- | --- | --- |
| Номер експерименту | Значення цільової функції | Точка мінімуму | Ітерація, на якій був знайдений мінімум |
| 1 | 0 | (0,0) | 14 |
| 2 | 0 | (0,0) | 42 |
| 3 | 0 | (0,0) | 25 |
| 4 | 0 | (0,0) | 22 |
| 5 | 0 | (0,0) | 23 |
| 6 | 0 | (0,0) | 33 |
| 7 | 0 | (0,0) | 14 |
| 8 | 0 | (0,0) | 47 |
| 9 | 0 | (0,0) | 55 |
| 10 | 0 | (0,0) | 16 |

Графік десятого експерименту наведений на рисунку 4.2.



Рисунок 4.2 – Графік значення цільової функції

Як видно, обраних початкових даних було достатньо, щоб на відповідній ітерації знайти точне значення мінімуму перевірочної функції. Результати кожного експерименту співпали з реальним значенням мінімуму функції, отже можна вважати, що перевірка генетичного алгоритму за допомогою сферичної функції завершилася успішно, та усі генетичні оператори працюють коректно.

Друга використана для перевірки алгоритму функція - це функція Растрігіна, яка будується за формулою

Ця функція має складний рельєф, тому вважається досить складною для оптимізації за допомогою генетичних алгоритмів. Мінімальне значення функції дорівнює 0 у точці (0,0).

Графік функції Растрігіна наведений на рисунку 4.3.

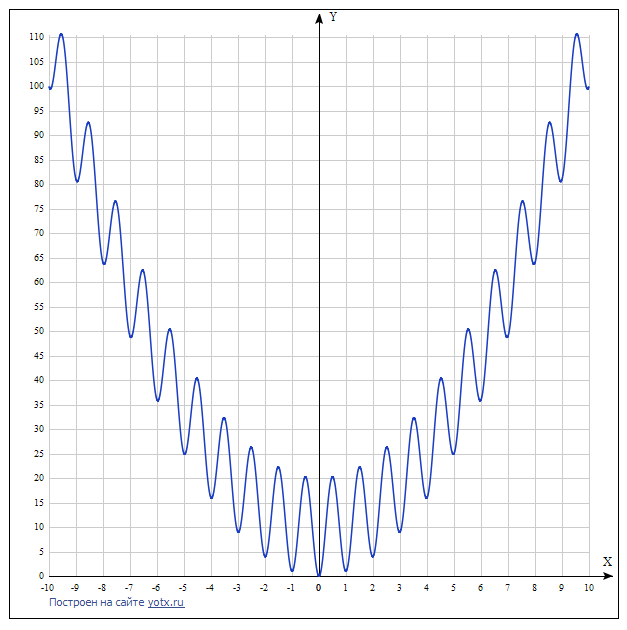


Рисунок 4.3 – Функція Растрігіна

Перевірка роботи алгоритму за допомогою функції Растрігіна здійснювалася для популяції розміром в 100 особин, з вірогідністю мутації 0.5 та умовою завершення роботи алгоритму після 100 ітерацій. Результати експерименту наведені у таблиці 4.2.

Таблиця 4.2 – Результати роботи алгоритму за другою функцією

|  |  |  |  |
| --- | --- | --- | --- |
| Номер експерименту | Значення цільової функції | Точка мінімуму | Ітерація, на якій був знайдений мінімум |
| 1 | 0 | (0,0) | 27 |
| 2 | 0 | (0,0) | 21 |
| 3 | 0 | (0,0) | 41 |
| 4 | 0 | (0,0) | 42 |
| 5 | 0 | (0,0) | 18 |
| 6 | 0 | (0,0) | 36 |
| 7 | 0 | (0,0) | 61 |
| 8 | 0 | (0,0) | 25 |
| 9 | 0 | (0,0) | 34 |
| 10 | 0 | (0,0) | 18 |

Графік десятого експерименту наведений на рисунку 4.4.



Рисунок 4.4 – Графік значення цільової функції

Як показали результати перевірки, обраних початкових даних достатньо, щоб на відповідній ітерації знайти точне значення мінімуму перевірочної функції. Результати кожного експерименту співпали з реальним значенням мінімуму функції Растрігіна, отже можна вважати, що перевірка генетичного алгоритму завершилася успішно.

Отже, був проведений аналіз коректності роботи генетичного алгоритму за допомогою сферичної моделі та функції Растрігіна. Результати перевірки показали, що загальний генетичний алгоритм та усі застосовані у ньому генетичні оператори буди побудовані коректно.

4.2 **Перевірка коректності побудови багатокритеріальних генетичних алгоритмів**

Завершивши перевірку генетичних операторів та загального генетичного алгоритму, переходимо до перевірки генетичних алгоритмів багатокритеріальної оптимізації.

У програмі були використані чотири методи багатокритеріальної оптимізації генетичними алгоритмами, що реалізують різні схеми призначення придатності та селекції:

1. RWGA – Random Weights Genetic Algorithm;
2. VEGA – Vector Evaluated Genetic Algorithm;
3. MOGA – Multiobjective Genetic Algorithm;
4. NPGA – Niched Pareto Genetic Algorithm.

Для перевірки роботи генетичних алгоритмів багатокритеріальної оптимізації були знайдені дві функції з відомими значеннями мінімуму.

Перша функція

,

де значення мінімуму на області [1; 10] по *x* та [1; 10] по *y* дорівнює 0.01 у точці (10, 10).

Друга функція

де значення мінімуму на області [1; 10] по *x* та [1; 10] по *y* дорівнює 0.5013 у точці (10, 10).

Спочатку була проведена перевірка генетичного алгоритму RWGA, результати якої зображено у таблиці 4.3.

Таблиця 4.3 – Результати перевірки методу RWGA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер експерименту | Значення першої цільової функції | Значення другої цільової функції | Точка мінімуму | Ітерація, на якій був знайдений результат |
| 1 | 0.01 | 0.5013 | (10,10) | 15 |
| 2 | 0.01 | 0.5013 | (10,10) | 8 |
| 3 | 0.01 | 0.5013 | (10,10) | 10 |
| 4 | 0.01 | 0.5013 | (10,10) | 8 |
| 5 | 0.01 | 0.5013 | (10,10) | 8 |
| 6 | 0.01 | 0.5013 | (10,10) | 13 |
| 7 | 0.01 | 0.5013 | (10,10) | 12 |
| 8 | 0.01 | 0.5013 | (10,10) | 7 |
| 9 | 0.01 | 0.5013 | (10,10) | 16 |
| 10 | 0.01 | 0.5013 | (10,10) | 12 |

Результати перевірки співпадають з відомими значеннями мінімумів функцій та точок мінімуму тобто метод RWGA працює коректно.

Далі був перевірений метод VEGA, результати перевірки наведені у таблиці 4.4.

Таблиця 4.4 – Результати перевірки методу VEGA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер експерименту | Значення першої цільової функції | Значення другої цільової функції | Точка мінімуму | Ітерація, на якій був знайдений результат |
| 1 | 0.01 | 0.5013 | (10,10) | 6 |
| 2 | 0.01 | 0.5013 | (10,10) | 8 |
| 3 | 0.01 | 0.5013 | (10,10) | 8 |
| 4 | 0.01 | 0.5013 | (10,10) | 16 |
| 5 | 0.01 | 0.5013 | (10,10) | 9 |
| 6 | 0.01 | 0.5013 | (10,10) | 10 |
| 7 | 0.01 | 0.5013 | (10,10) | 13 |
| 8 | 0.01 | 0.5013 | (10,10) | 8 |
| 9 | 0.01 | 0.5013 | (10,10) | 11 |
| 10 | 0.01 | 0.5013 | (10,10) | 7 |

Результати усіх проведених експериментів співпадають з відомими значеннями мінімумів функцій та точок мінімуму, отже можна вважати, що генетичний алгоритм VEGA побудований коректно.

Наступний перевірений метод - генетичний алгоритм MOGA, результати експериментів наведені у таблиці 4.5.

Таблиця 4.5 – Результати перевірки методу MOGA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер експерименту | Значення першої цільової функції | Значення другої цільової функції | Точка мінімуму | Ітерація, на якій був знайдений результат |
| 1 | 0.01 | 0.5013 | (10,10) | 6 |
| 2 | 0.01 | 0.5013 | (10,10) | 16 |
| 3 | 0.01 | 0.5013 | (10,10) | 11 |
| 4 | 0.01 | 0.5013 | (10,10) | 11 |
| 5 | 0.01 | 0.5013 | (10,10) | 8 |
| 6 | 0.01 | 0.5013 | (10,10) | 7 |
| 7 | 0.01 | 0.5013 | (10,10) | 9 |
| 8 | 0.01 | 0.5013 | (10,10) | 10 |
| 9 | 0.01 | 0.5013 | (10,10) | 4 |
| 10 | 0.01 | 0.5013 | (10,10) | 6 |

Як бачимо, результати перевірки співпадають з відомими значеннями мінімумів функцій та точок мінімуму, отже генетичний алгоритм MOGA працює коректно.

Останній з методів, що був використаний у додатку, був метод NPGA. Результати його роботи наведені у таблиці 4.6.

Таблиця 4.6 – Результати перевірки методу NPGA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер експерименту | Значення першої цільової функції | Значення другої цільової функції | Точка мінімуму | Ітерація, на якій був знайдений результат |
| 1 | 0.01 | 0.5013 | (10,10) | 10 |
| 2 | 0.01 | 0.5013 | (10,10) | 7 |
| 3 | 0.01 | 0.5013 | (10,10) | 16 |
| 4 | 0.01 | 0.5013 | (10,10) | 12 |
| 5 | 0.01 | 0.5013 | (10,10) | 13 |
| 6 | 0.01 | 0.5013 | (10,10) | 19 |
| 7 | 0.01 | 0.5013 | (10,10) | 14 |
| 8 | 0.01 | 0.5013 | (10,10) | 11 |
| 9 | 0.01 | 0.5013 | (10,10) | 16 |
| 10 | 0.01 | 0.5013 | (10,10) | 8 |

У результати перевірки виявилося, що значення, вирахувані алгоритмом, співпадають з відомими значеннями мінімумів функцій та точок мінімуму. Отже генетичний алгоритм NPGA побудований коректно.

Як виявилося у результаті проведеної перевірки, усі чотири використані у додатку генетичні алгоритми багатокритеріальної оптимізації були побудовані коректно.

4.3 **Пошук оптимальних початкових даних для багатокритеріальних генетичних алгоритмів**

Для коректної роботи генетичного алгоритму дуже важливо знайти значення розміру популяції, кількості ітерацій та вірогідності мутації, що забезпечують правильність результатів. Для пошуку необхідних значень цих початкових даних був проведений аналіз.

Зафіксуємо розмір популяції у 4 особини, а вірогідність мутації будемо змінювати як 0.1, 0.3 та 0.5. Результати експерименту наведені у таблиці 4.7.

Таблиця 4.7 – Результати експерименту

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Кількість ітерацій | Значення цільових функцій на останньому кроці алгоритму | | | |
| RWGA | VEGA | MOGA | NPGA |
| Вірогідність мутації 0.1 | | | | |
| 10 | 07915  0,0239 | 0,7577  0,0241 | 0,5696  0,0129 | 0,7632  0,231 |
| 15 | 0,7210  0,0201 | 0,6875  0,1865 | 0,7796  0,0241 | 0,8247  0,0270 |
| 25 | 0,5728  0,0130 | 0,5582  0,0133 | 0,0543  0,0117 | 0,899  0,0322 |
| 50 | 0,5016  0,0100 | 0,6717  0,0192 | 0,5052  0,0101 | 0,5014  0,0100 |
| 100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5014  0,0100 |
| Вірогідність мутації 0.3 | | | | |
| 10 | 0,6235  0,0157 | 0,6100  0,0142 | 0,5014  0,0100 | 0, 9535  0,0364 |
| 15 | 0,5487  0,0117 | 0,6149  0,0189 | 0,5568  0,0123 | 0,7028  0,0196 |
| 25 | 0,6589  0,1696 | 0,5786  0,0138 | 0,5098  0,0103 | 0,5104  0,0103 |
| 50 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |
| 100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |
| Вірогідність мутації 0.5 | | | | |
| 10 | 0,6661  0,0171 | 0,5466  0,0116 | 0,6112  0,0149 | 0,6652  0,0176 |
| 15 | 0,5826  0,0149 | 0,6226  0,0160 | 0,5065  0,0102 | 0,6040  0,0145 |
| 25 | 0,5019  0,0100 | 0,5013  0,0100 | 0,5214  0,0108 | 0,5912  0,0139 |
| 50 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |
| 100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |

Зафіксуємо розмір популяції у 8 особин, а вірогідність мутації будемо змінювати як 0.1, 0.3 та 0.5. Результати експерименту наведені у таблиці 4.8.

Таблиця 4.8 – Результати експерименту

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Кількість ітерацій | Значення цільових функцій на останньому кроці алгоритму | | | |
| RWGA | VEGA | MOGA | NPGA |
| Вірогідність мутації 0.1 | | | | |
| 10 | 0,6002  0,0146 | 0,6387  0,0162 | 0,7652  0,0232 | 0,6002  0,0144 |
| 15 | 0,6269  0,0154 | 0,5526  0,0124 | 0,6381  0,0162 | 0,6554  0,0170 |
| 25 | 0,5261  0,0105 | 0,5236  0,0107 | 0,5014  0,0100 | 0,5781  0,0132 |
| 50 | 0,5014  0,0100 | 0,5020  0,0100 | 0,5025  0,0100 | 0,5295  0,0111 |
| 100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |
| Вірогідність мутації 0.3 | | | | |
| 10 | 0,5113  0,0105 | 0,7006  0,0184 | 0,5328  0,0112 | 0,7738  0,0238 |
| 15 | 0,5536  0,0125 | 0,5508  0,0127 | 0,5218  0,0108 | 0,7047  0,0197 |
| 25 | 0,5148  0,0100 | 0,5102  0,0101 | 0,5013  0,0100 | 0,5354  0,0114 |
| 50 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |
| 100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |
| Вірогідність мутації 0.5 | | | | |
| 10 | 0,5891  0,0125 | 0,5299  0,113 | 0,6014  0,0143 | 0,6286  0,0157 |
| 15 | 0,5268  0,0111 | 0,5597  0,0127 | 0,5013  0,0100 | 0,5498  0,0120 |
| 25 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |
| 50 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |
| 100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 | 0,5013  0,0100 |

За результатами перевірки маємо, що при розмірі популяції у 4 особини, вірогідності мутації менше за 0.5 та кількості ітерацій менше за 50 можуть виникнути розбіжності у результатах. При розмірі популяції у 8 особин, вірогідності мутації у 0.5 та кількості ітерацій більш ніж 50, значення цільових функцій для усіх генетичних алгоритмів співпадають при багатьох перевірках. Ці значення можна вважати знайденими мінімумами функцій, які дорівнюють *min f1* = 0,5013 та *min f2* = 0,01 у точці *x* = 10, *y* = 10.

Приклад побудованого програмою графіку значень цільових функцій наведено на рисунку 4.5.



Рисунок 4.5 – Графік значень цільових функцій за методом MOGA