1. РОЗВ’ЯЗАННЯ ЗАДАЧІ ТОКАРНОЇ ОБРОБКИ ТИТАНУ

У попередньому розділі роботи завдяки моделюванню тестових функцій було встановлено, що алгоритм обчислювання складений вірно, а розроблений додаток працює коректно. Також був проведений пошук оптимальних початкових показників генетичного алгоритму для використаних прикладів. Тепер перейдемо до рішення задачі токарної обробки титану з субмікро- і нанокристалічною структурою за допомогою розробленого додатку, де мінімізуються функція розміру зерна металу *f1* та функція ефективності *f2*.

Зафіксуємо розмір популяції у чотири особини, а вірогідність мутації будемо змінювати як 0.1, 0.3 та 0.5. Результати експерименту наведені у таблиці 5.1.

Таблиця 5.1 – Результати експерименту

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Кількість ітерацій | Значення цільових функцій на останньому кроці алгоритму | | | |
| RWGA | VEGA | MOGA | NPGA |
| Вірогідність мутації 0.1 | | | | |
| 10 | 2,0127  0,000000083 | 4,5687  0,000000167 | 1,0792  0,000000065 | 1,1356  0,000000167 |
| 15 | 1,9755  0,000000169 | 5,4469  0,000000145 | 1,951  0,000000197 | 1,834  0,000000094 |
| 25 | 1,824  0,000000094 | 1,706  0,000000188 | 1,7866  0,000000068 | 2,0736  0,000000081 |
| 50 | 1,002  0,000000081 | 1,0564  0,000000085 | 0,977517  0,000000042 | 0,9856  0,000000116 |
| 100 | 0,9801  0,000000027 | 0,9791  0,000000027 | 0,9775  0,000000026 | 0,9775  0,000000033 |
| Вірогідність мутації 0.3 | | | | |
| 10 | 2,9372  0,000000113 | 2,6202  0,000000141 | 1,3849  0,000000099 | 3,1194  0,000000120 |
| 15 | 1,1200  0,000000090 | 1,0808  0,00000079 | 1,3411  0,000000096 | 0,9775  0,000000051 |
| 25 | 1,085  0,000000057 | 1,1794  0,000000094 | 0,9881  0,000000026 | 0,9775  0,00000045 |

Завершення таблиці 5.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Кількість ітерацій | RWGA | VEGA | MOGA | NPGA |
| 50 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9813  0,000000026 |
| 100 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 |
| Вірогідність мутації 0.5 | | | | |
| 10 | 1,9351 | 2,2609  0,000000080 | 1,8923  0,000000097 | 1,1608  0,000000059 |
| 15 | 0,9783 | 0,9779  0,000000071 | 0,9776  0,000000034 | 1,1338  0,000000125 |
| 25 | 0,9775 | 0,9775  0,000000032 | 0,9775  0,000000033 | 1,0120  0,000000026 |
| 50 | 0,9775 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 |
| 100 | 0,9775 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 |

Приклад графіку цільових функцій, що будує програма під час роботи, наведено на рисунку 5.1, де проводиться оптимізація за методом NPGA для

популяції у чотири особини, 50 ітерацій та вірогідності мутації 0,5.



Рисунок 5.1 - Графік значень цільових функцій за методом NPGA

Далі оберемо розмір популяції у 12 особин, а вірогідність мутації будемо змінювати як 0.1, 0.3 та 0.5. Результати експерименту наведені у таблиці 5.2.

Таблиця 5.2 – Результати експерименту

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Кількість ітерацій | Значення цільових функцій на останньому кроці алгоритму | | | |
| RWGA | VEGA | MOGA | NPGA |
| Вірогідність мутації 0.1 | | | | |
| 10 | 1,4211  0,000000123 | 1,3375  0,000000011 | 1,4360  0,000000100 | 2,407  0,000000131 |
| 15 | 1,2349  0,000000109 | 1,1690  0,000000115 | 1,1995  0,000000107 | 1,6314  0,000000164 |
| 25 | 1,1183  0,000000093 | 1,3812  0,000000060 | 1,1514  0,000000086 | 0,8996  0,000000088 |
| 50 | 0,9972  0,000000039 | 1,1001  0,000000044 | 0,9939  0,000000102 | 1,2427  0,000000040 |
| 100 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000044 |
| Вірогідність мутації 0.3 | | | | |
| 10 | 1,8531  0,000000054 | 1,5345  0,000000033 | 1,9592  0,000000086 | 1,7940  0,000000117 |
| 15 | 1,2117  0,000000042 | 1,0075  0,000000084 | 1,509  0,000000035 | 0,9775  0,000000059 |
| 25 | 0,9789  0,000000028 | 0,9787  0,000000036 | 0,9775  0,000000029 | 0,9775  0,000000044 |
| 50 | 0,9775  0,000000027 | 0,9775  0,000000027 | 0,9775  0,000000026 | 0,9775  0,000000027 |
| 100 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000027 |
| Вірогідність мутації 0.5 | | | | |
| 10 | 1,0918  0,000000030 | 1,0296  0,000000055 | 1,0435  0,000000070 | 1,1523  0,000000069 |
| 15 | 0,9775  0,000000041 | 1,0082  0,000000035 | 0,9775  0,000000028 | 0,9775  0,000000030 |
| 25 | 0,9775  0,000000028 | 0,9775  0,000000030 | 0,9775  0,000000026 | 0,9775  0,000000027 |
| 50 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 |
| 100 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 | 0,9775  0,000000026 |

Отже, після проведеного аналізу результатів оптимізації 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 12 особин, вірогідності мутації у 0.5 та кількості ітерацій біль ніж 50, значення цільових функцій для усіх генетичних алгоритмів співпадають при багатьох перевірках. Ці значення можна вважати знайденими мінімумами функцій, які дорівнюють *min f1* = 0,9775 та *min f2* = 0,000000026 у точці *s* = 0.0123, *v* = 100, *L* = 0.1, *d* = 0.01 *D0* = 2e-8.

Інтерфейс додатку з результатами роботи програми та побудованими графіками зображено на рисунку 5.2.



Рисунок 5.2 - Результати оптимізації

Далі проведемо порівняльний аналіз роботи різноманітних генетичних алгоритмів за швидкістю пошуку оптимального значення.

Для цього проведемо серію експериментів, встановивши розмір популяції у 12 особин, кількість ітерацій у 50 та вірогідність мутації у 0.5.

Результати наведені у таблиці 5.3.

Таблиця 5.3 - Порівняльний аналіз генетичних алгоритмів

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер експерименту | Ітерація, на якій був знайдений результат | | | |
| RWGA | VEGA | MOGA | NPGA |
| 1 | 18 | 16 | 12 | 20 |
| 2 | 16 | 22 | 18 | 37 |
| 3 | 15 | 24 | 21 | 22 |
| 4 | 26 | 25 | 15 | 24 |
| 5 | 19 | 16 | 22 | 32 |
| 6 | 23 | 18 | 16 | 20 |
| 7 | 18 | 24 | 23 | 18 |
| 8 | 19 | 21 | 21 | 22 |
| 9 | 20 | 22 | 13 | 19 |
| 10 | 19 | 20 | 14 | 26 |
| Середнє значення | 19,3 | 20,8 | 17,5 | 24 |

Отже, за результатами проведеної перевірки було встановлено, що найшвидше оптимальне рішення знаходить метод MOGA, що доводить високу продуктивність генетичних алгоритмів, заснованих на використанні рангів Парето. В той час найповільніше рішення знаходить метод NPGA, бо його механізм збереження різноманітності популяції та запобігання передчасного сходження алгоритму сповільнює процес оптимізації.