

Algorytmy Ewolucyjne

Projekt 1 - sprawozdanie

Tomasz Indeka, 293457

Wstęp	2
Wyniki	3
Liczba przedmiotów: 32	3
Wpływ metody selekcji	4
Wpływ metody krzyżowania	6
Wpływ metody mutacji	8
Liczba przedmiotów: 64	10
Wpływ metody selekcji	12
Wpływ metody krzyżowania	14
Wpływ metody mutacji	16
Wnioski	18

1. Wstęp

Do rozwiązania zadany był standardowy problem plecakowy, który cechuje się następującymi równaniami:

$$f_{celu} = \max \sum_{i=1}^N p_i x_i ,$$
$$\sum_{i=1}^N w_i x_i \leq W ,$$

gdzie:

$$p_i \in < 0, 1; 1 > ,$$

$$w_i \in < 1; 100 > ,$$

$$x_i \in \{0; 1\} ,$$

$$W = 0,3 * \sum_{i=1}^N w_i ,$$

$N = 32$ lub $N = 64$ (zależnie od rozpatrywanego przypadku).

Same wektory p i w ze względu na swój rozmiar znajdują się w tabelach w dalszej części sprawozdania i zostały wygenerowane losowo na podstawie nr albumu. Wektor p odpowiada wartościom przedmiotów, a wektor w wadze przedmiotów w problemie plecakowym.

Do rozwiązania zadania użyłem funkcji *ga* (algorytm genetyczny) programu MATLAB z ustaleniem populacji na wartości bitowe.

W zadaniu skupiłem się na przeanalizowaniu wpływu parametrów algorytmu genetycznego (metody selekcji, mutacji i krzyżowania) na szybkość obliczenia maximum dla podanych danych. Obliczenia dokonywane przez algorytm genetyczny były w dużej mierze oparte na losowości dlatego przedstawione wyniki są uśrednieniem otrzymanych wyników z 25 wywołań.

Parametry takie jak warunek zatrzymania algorytmu były dobrane odpowiednio daleko, tak aby nie ograniczać algorytmu ewolucyjnego w obliczeniach, ale jednocześnie skrócić obliczenia o złym uwarunkowaniu. Mimo dobrania zadowalającej maksymalnej ilości iteracji nie wszystkie wywołania znalazły optimum w ograniczonej iteracji.

We wszystkich przypadkach przybrałem licznosc populacji 100, liczba elitarnych jednostek populacji 1, maksymalna liczba generacji 200, tolerancja końcowa funkcji 1e-6.

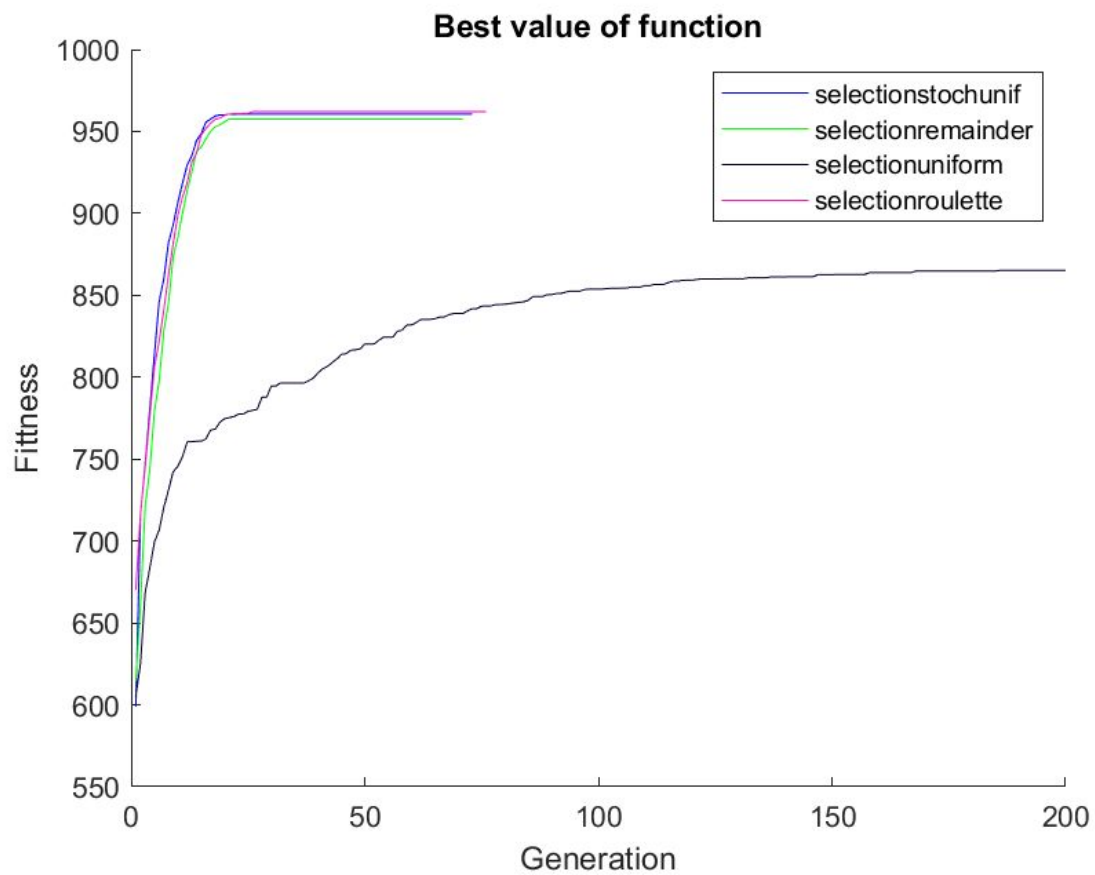
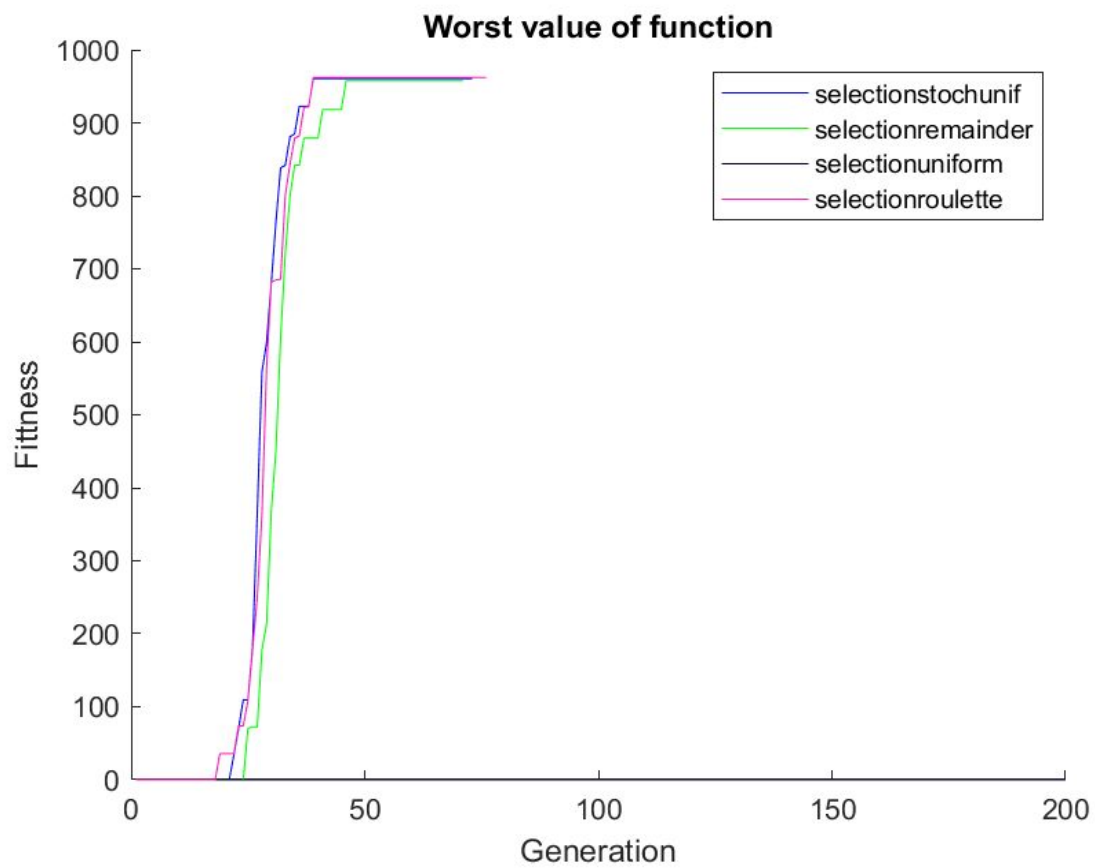
Najlepsze znalezione rozwiązanie jest przedstawione jest w poszczególnych tabelach, a wybrane przedmioty zaznaczone są na niebiesko.

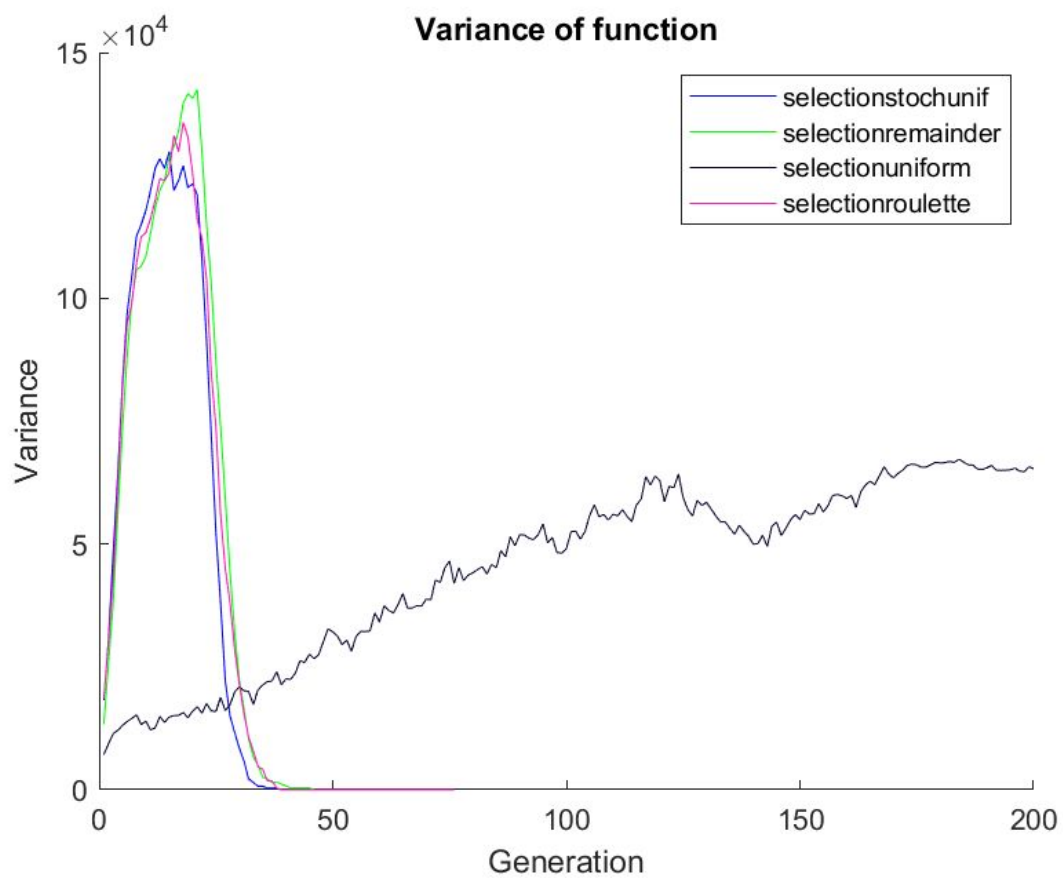
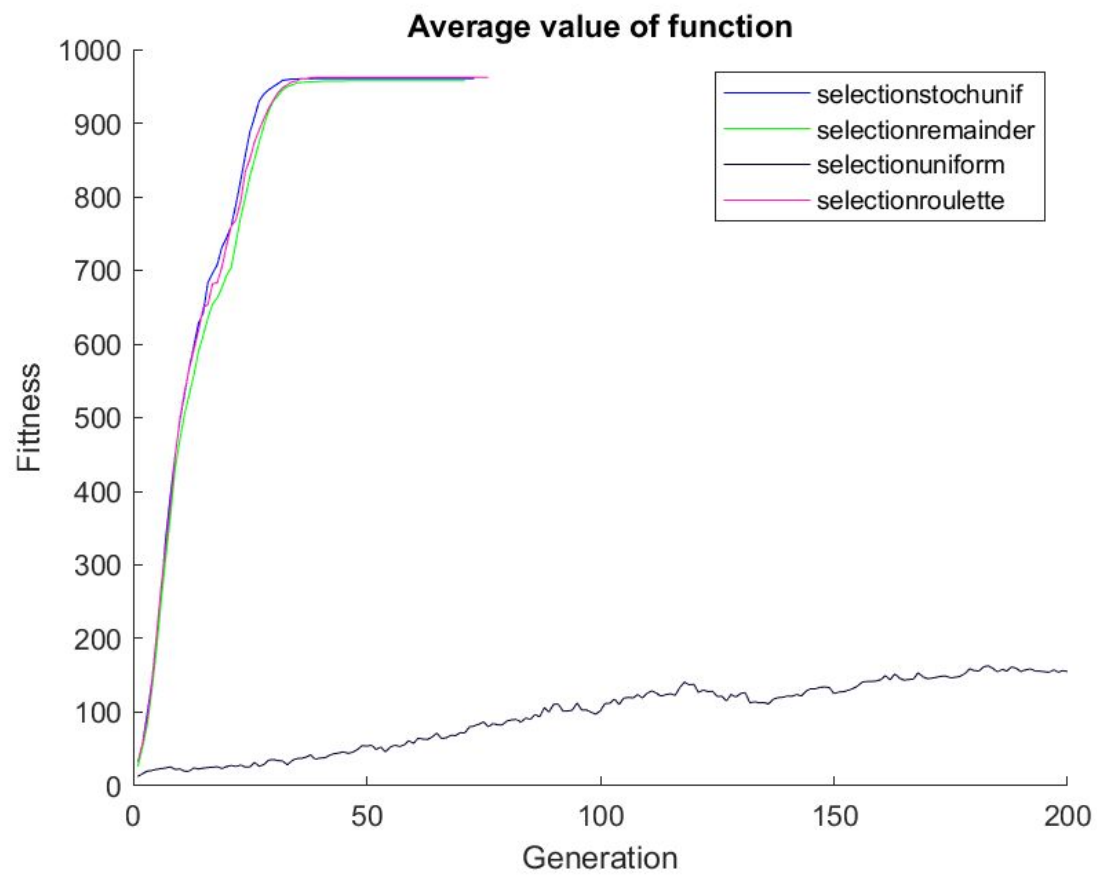
2. Wyniki

2.1. Liczba przedmiotów: 32

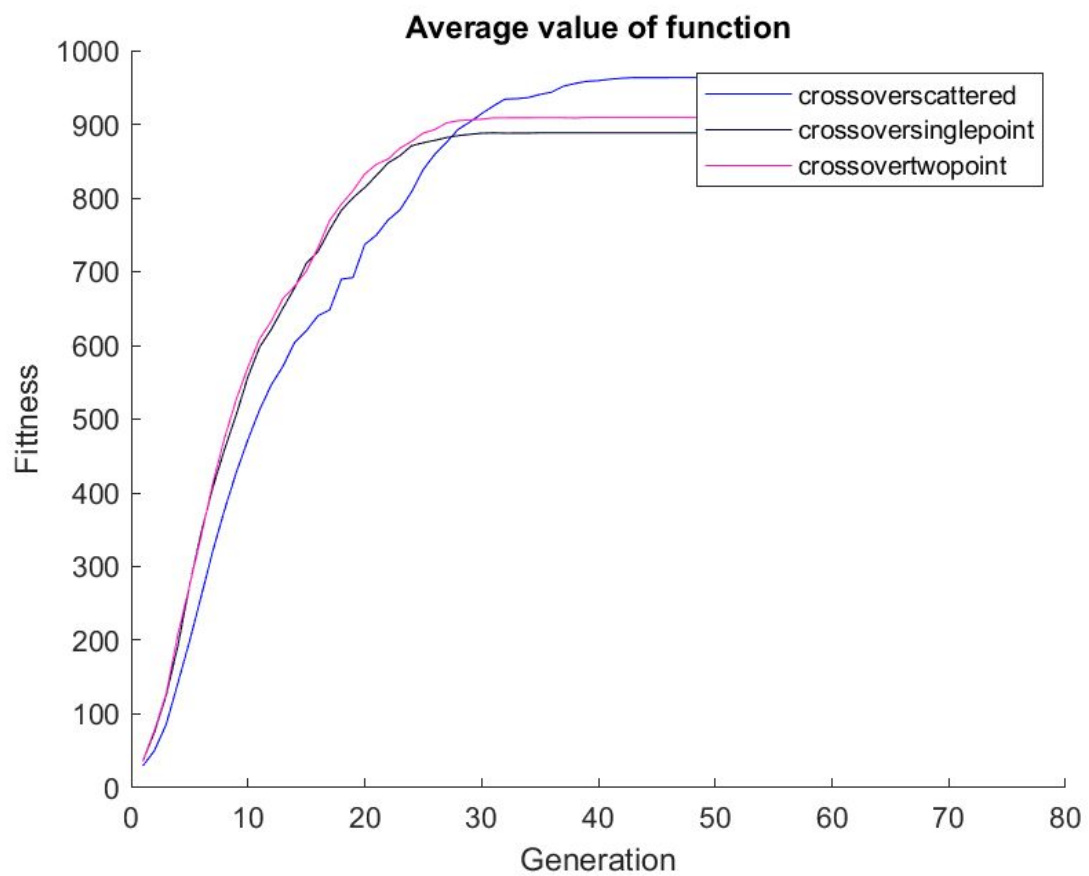
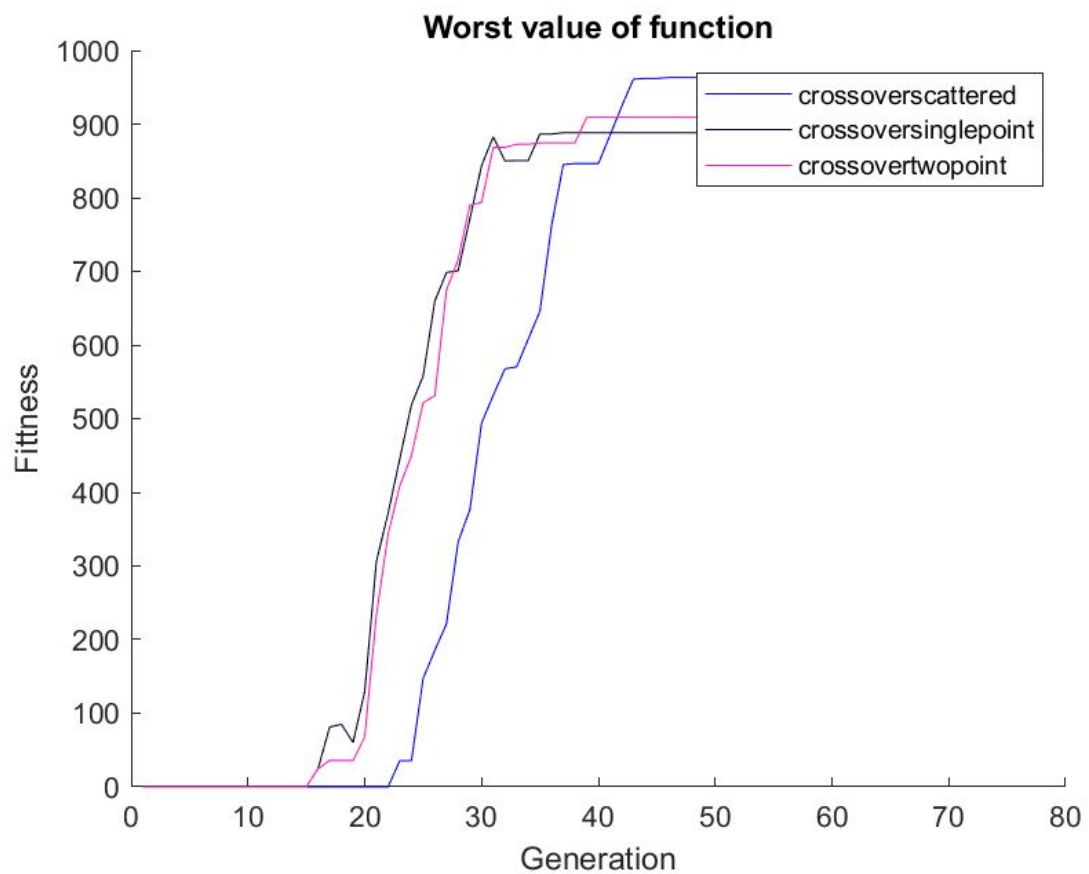
i	p_i	w_i	x_i	wartość	waga
1	44	0,2	1	44	0,2
2	88	0,7	1	88	0,7
3	84	0,5	1	84	0,5
4	58	0,5	0	0	0
5	49	0,7	0	0	0
6	48	0,4	0	0	0
7	32	0,2	1	32	0,2
8	97	0,5	1	97	0,5
9	16	0,6	0	0	0
10	30	0,9	0	0	0
11	86	0,2	1	86	0,2
12	61	1	0	0	0
13	22	0,6	0	0	0
14	74	0,4	1	74	0,4
15	18	0,3	0	0	0
16	86	0,5	1	86	0,5
17	8	0,1	1	8	0,1
18	35	0,4	0	0	0
19	99	0,3	1	99	0,3
20	59	0,9	0	0	0
21	44	0,3	1	44	0,3
22	12	0,3	0	0	0
23	46	0,4	0	0	0
24	73	0,2	1	73	0,2
25	83	0,8	0	0	0
26	49	0,9	0	0	0
27	58	0,5	0	0	0
28	9	0,7	0	0	0
29	40	0,9	0	0	0
30	99	0,2	1	99	0,2
31	69	0,6	1	69	0,6
32	40	0,8	0	0	0
		W = 4,95		983	4,9

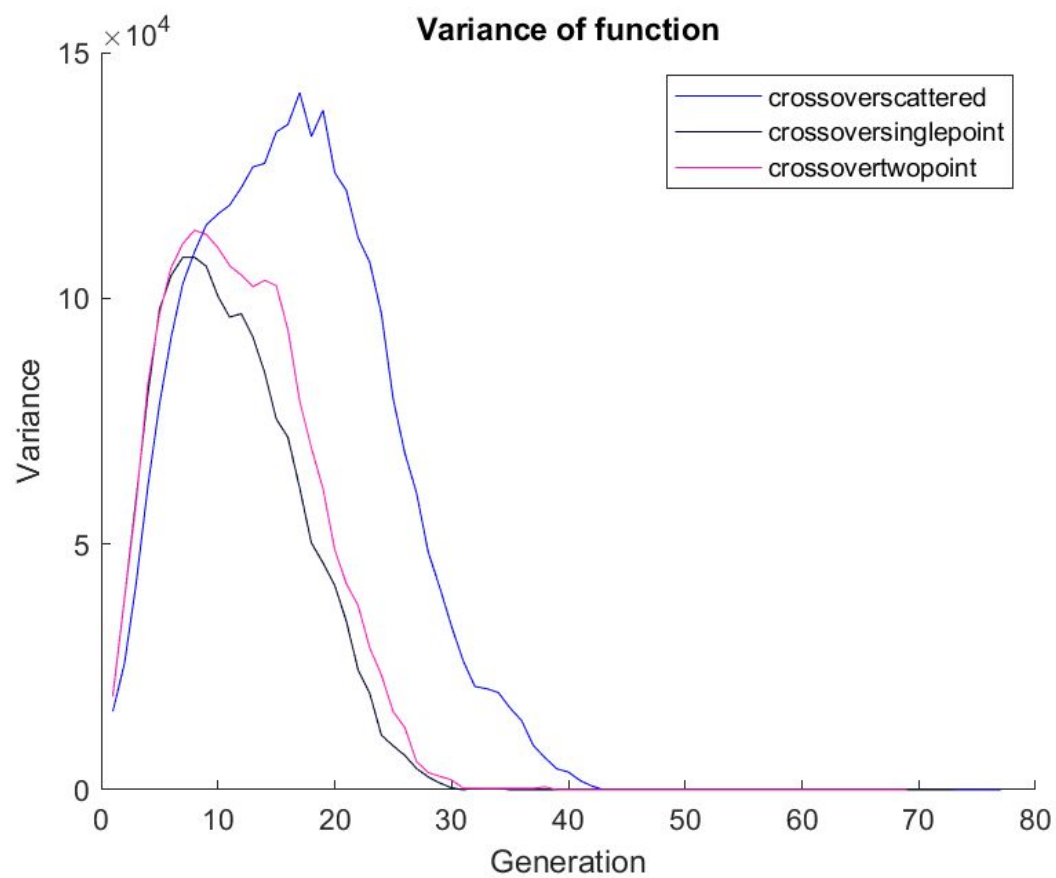
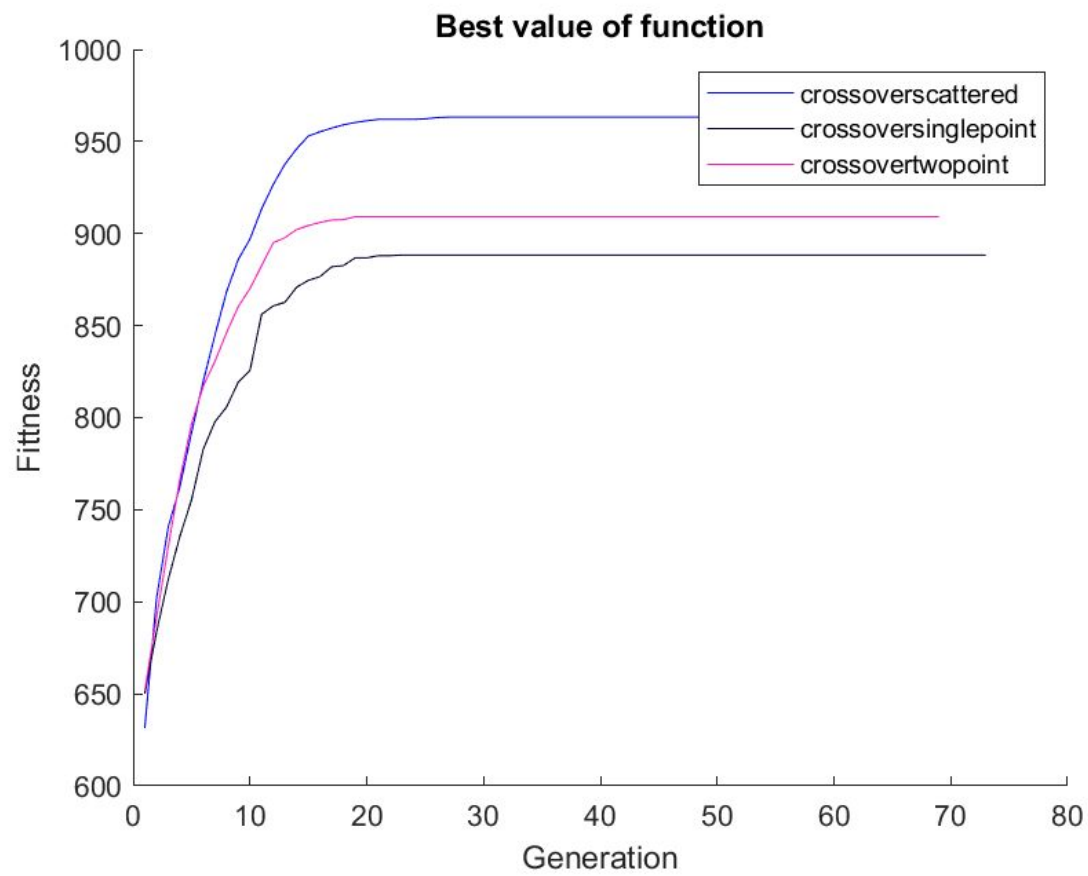
2.1.1. Wpływ metody selekcji



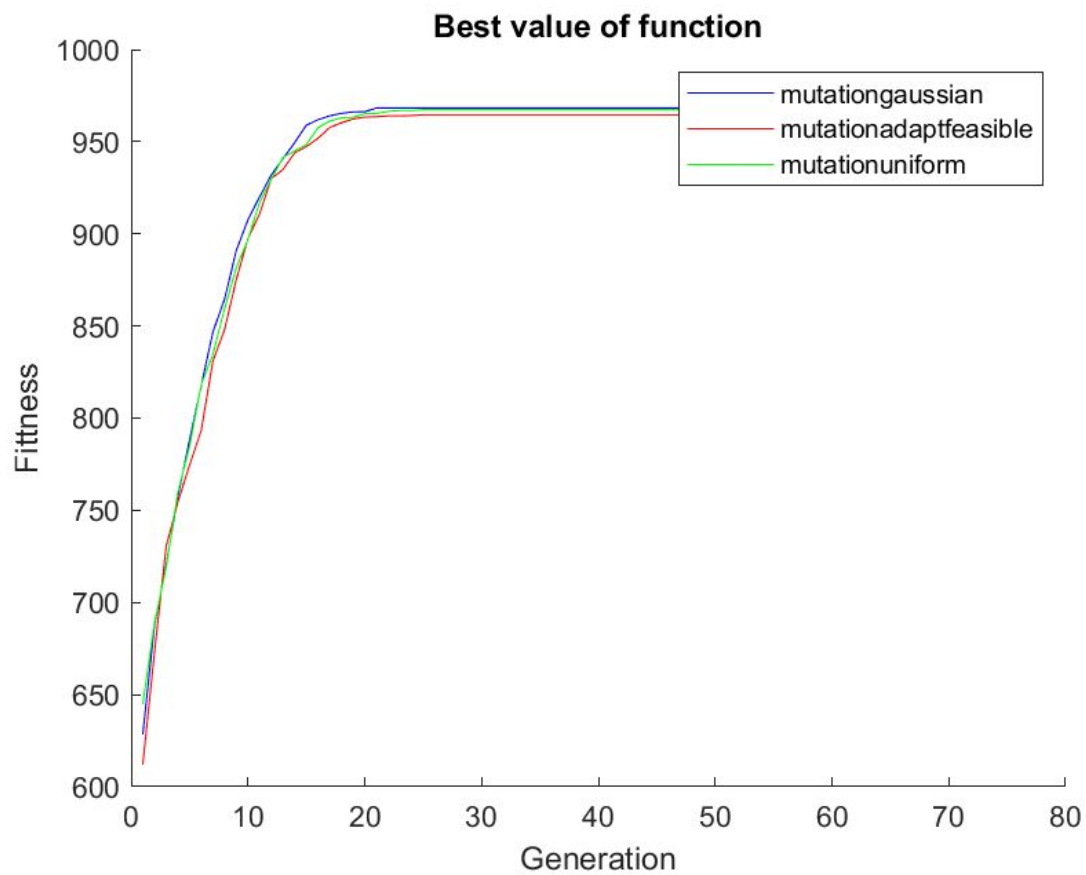
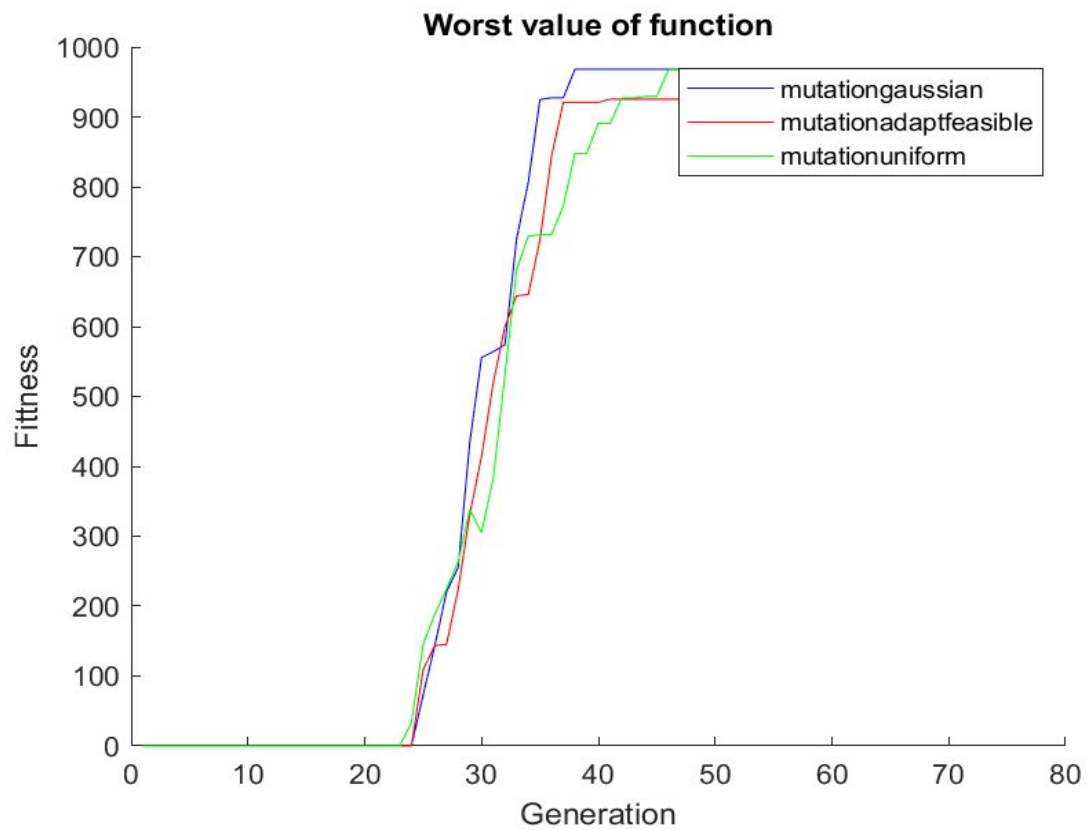


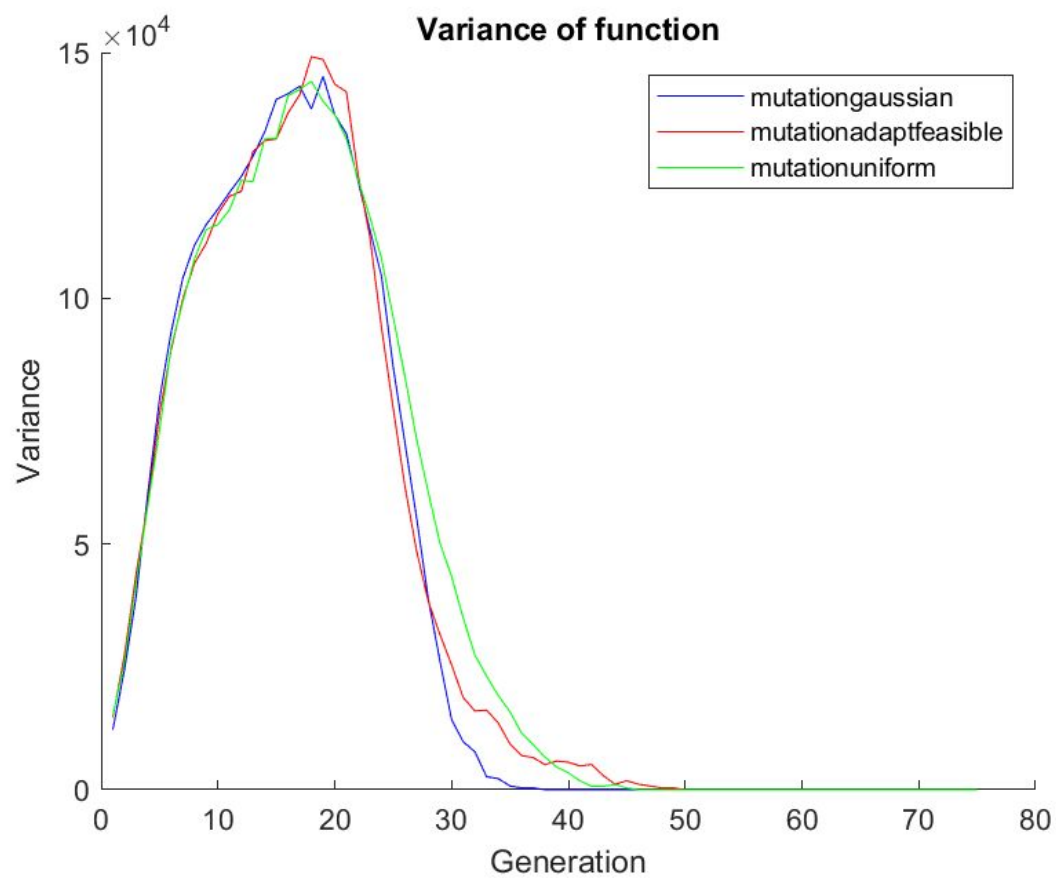
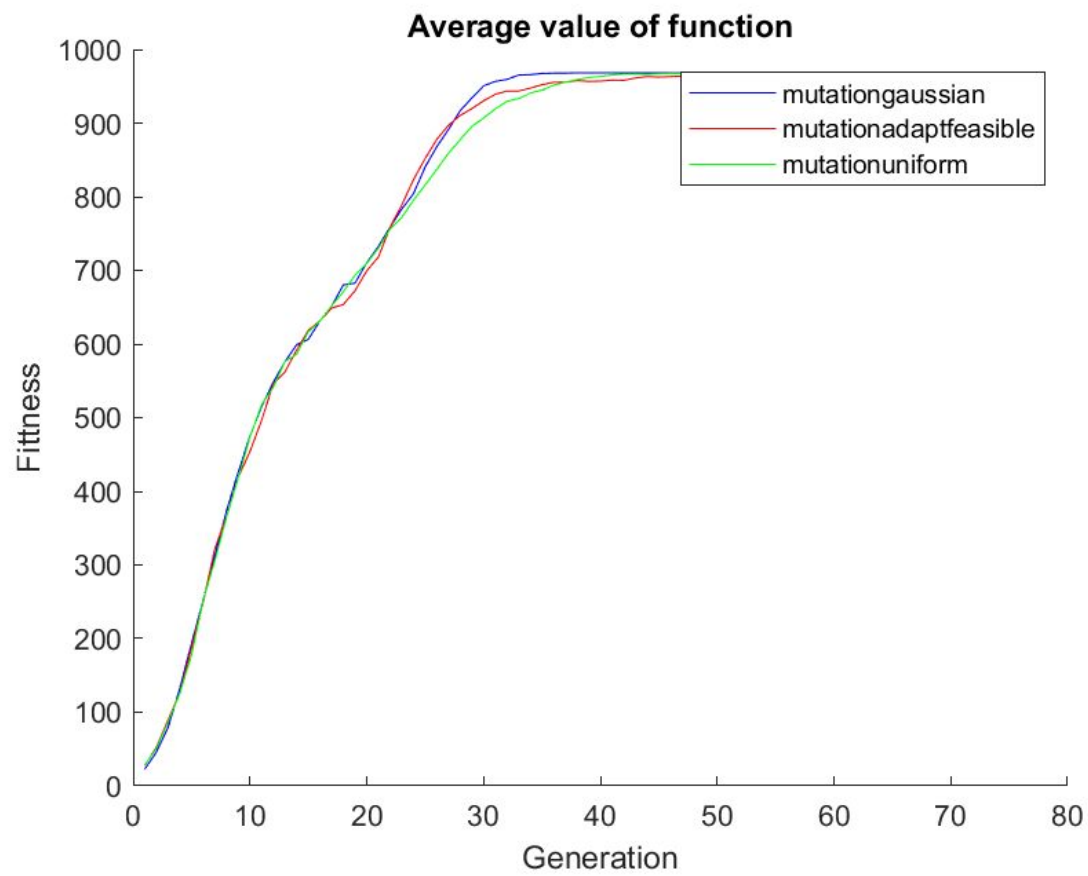
2.1.2. Wpływ metody krzyżowania





2.1.3. Wpływ metody mutacji



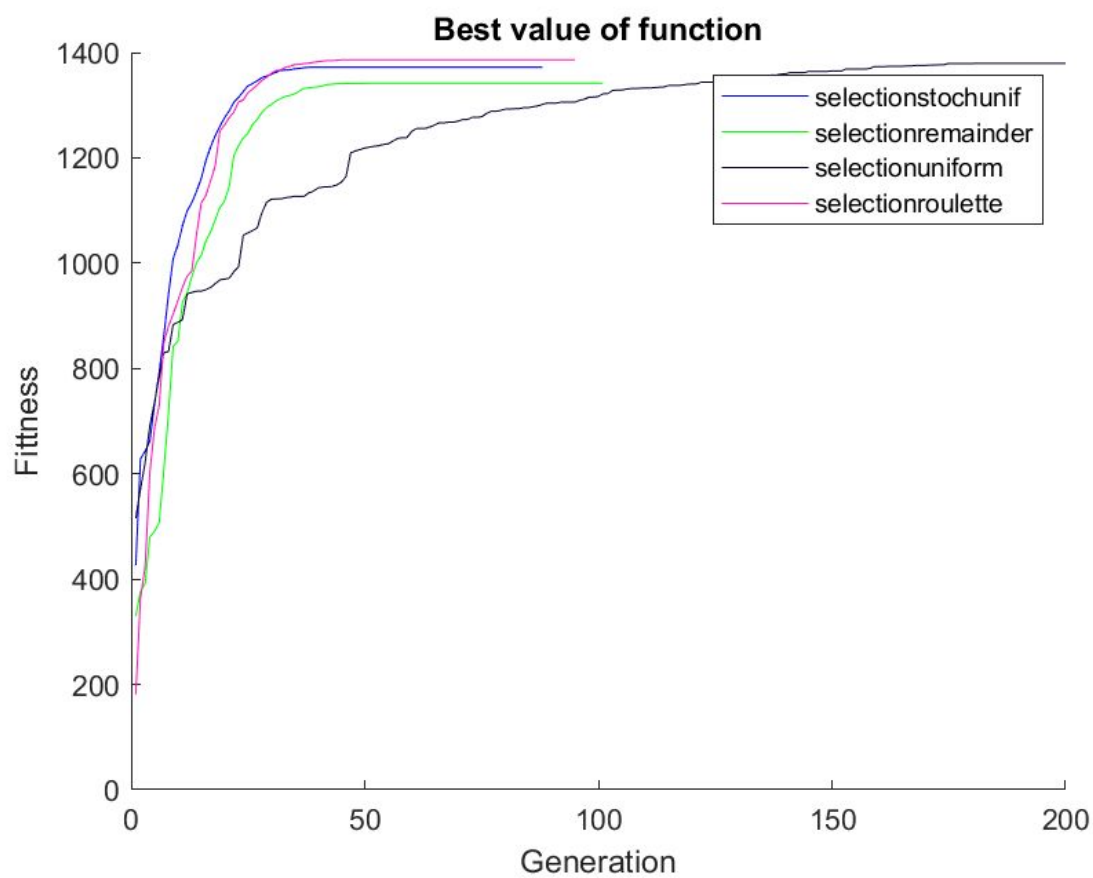
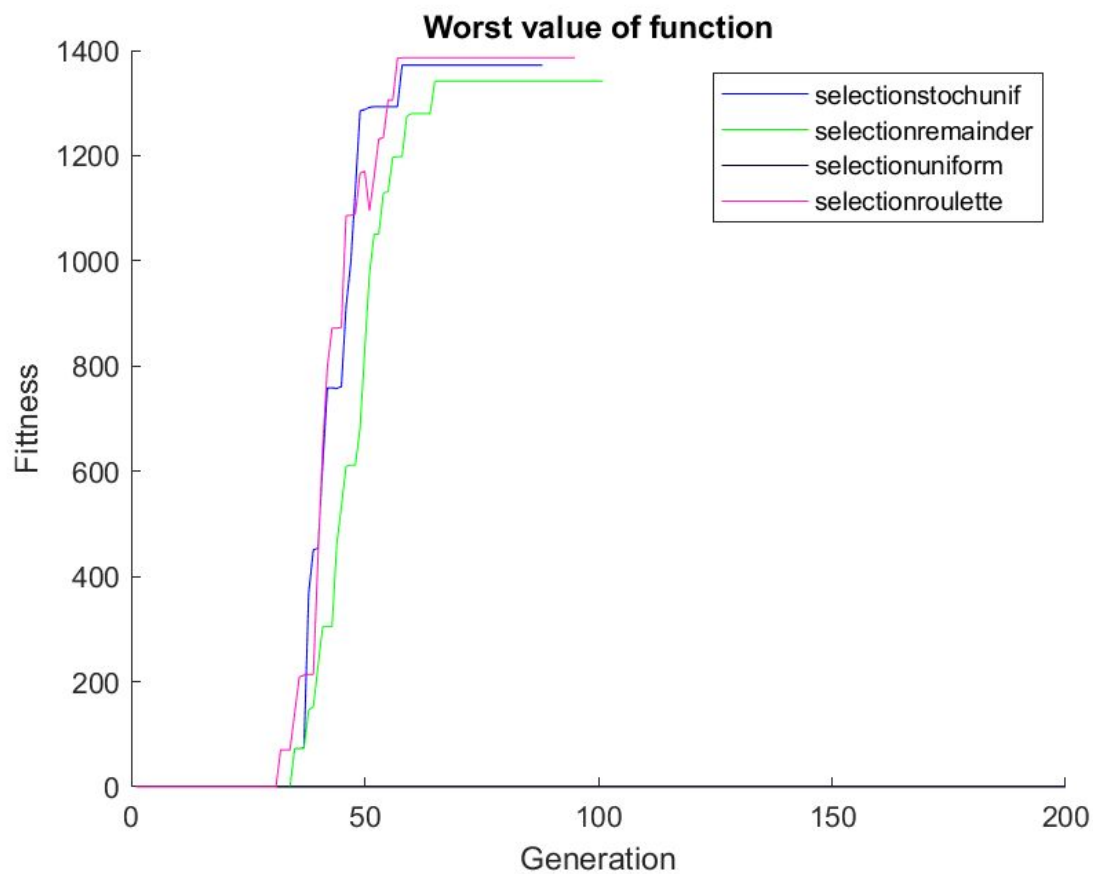


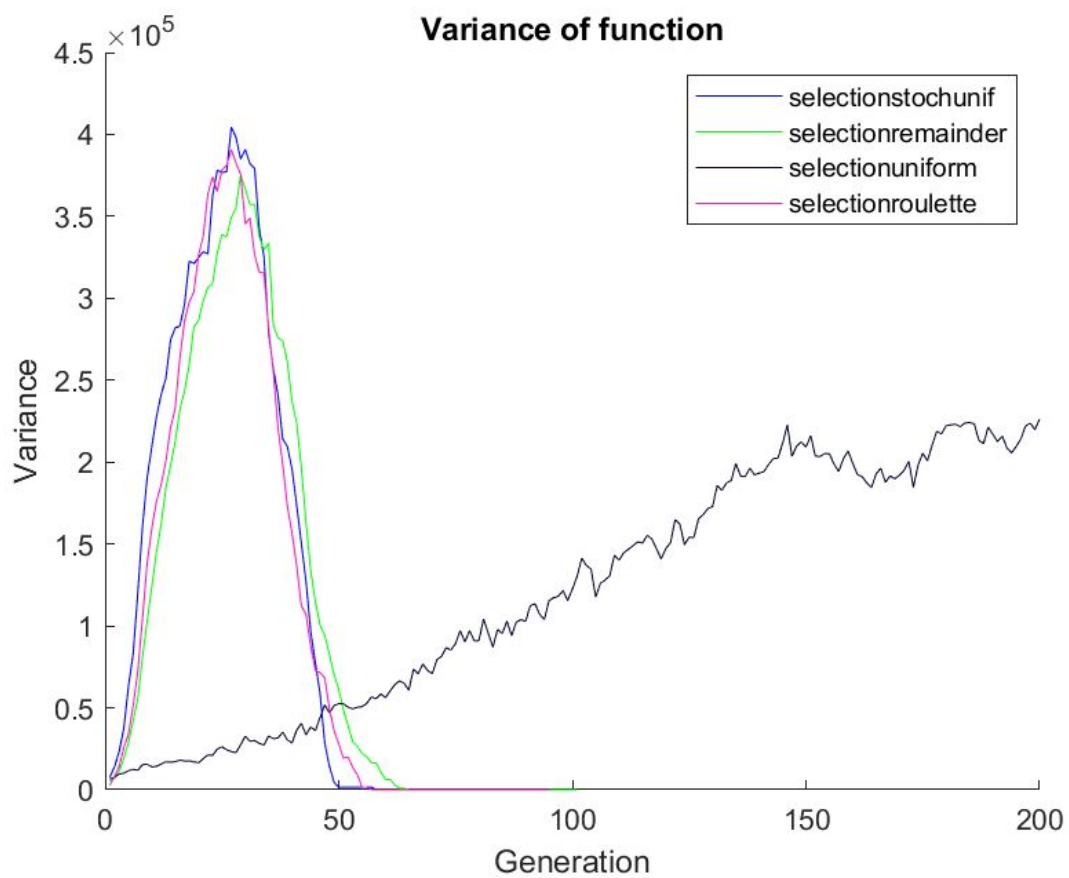
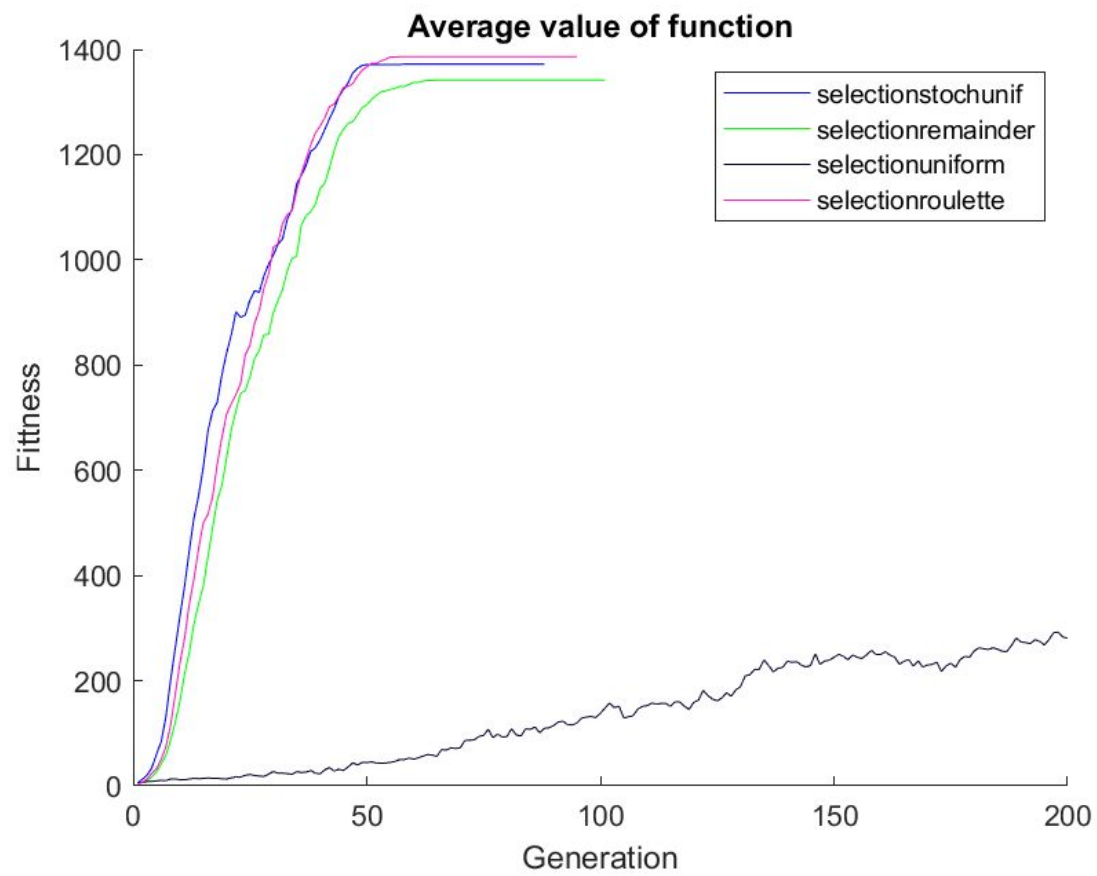
2.2. Liczba przedmiotów: 64

i	p_i	w_i	x_i	wartość	waga
1	26	0,2	1	26	0,2
2	46	0,7	0	0	0
3	79	0,5	1	79	0,5
4	55	0,5	0	0	0
5	16	0,7	0	0	0
6	68	0,4	1	68	0,4
7	39	0,2	1	39	0,2
8	51	0,5	0	0	0
9	84	0,6	1	84	0,6
10	74	0,9	0	0	0
11	69	0,2	1	69	0,2
12	40	1	0	0	0
13	19	0,6	0	0	0
14	4	0,4	0	0	0
15	39	0,3	1	39	0,3
16	66	0,5	1	66	0,5
17	80	0,1	1	80	0,1
18	6	0,4	0	0	0
19	58	0,3	1	58	0,3
20	38	0,9	0	0	0
21	47	0,3	1	47	0,3
22	89	0,3	1	89	0,3
23	99	0,4	1	99	0,4
24	84	0,2	1	84	0,2
25	53	0,8	0	0	0
26	37	0,9	0	0	0
27	72	0,5	1	72	0,5
28	25	0,7	0	0	0
29	64	0,9	0	0	0
30	43	0,2	0	0	0
31	52	0,6	0	0	0
32	37	0,8	0	0	0
33	20	0,5	0	0	0
34	9	0,9	0	0	0
35	23	0,9	0	0	0

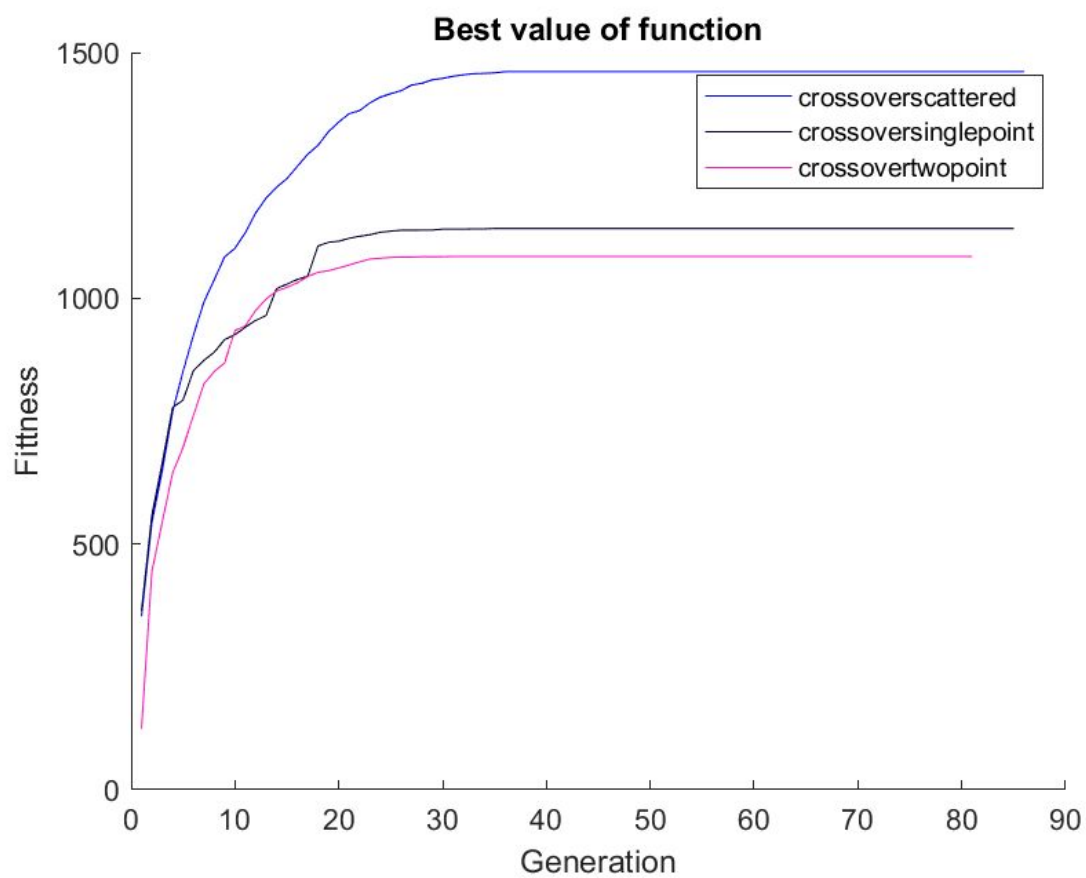
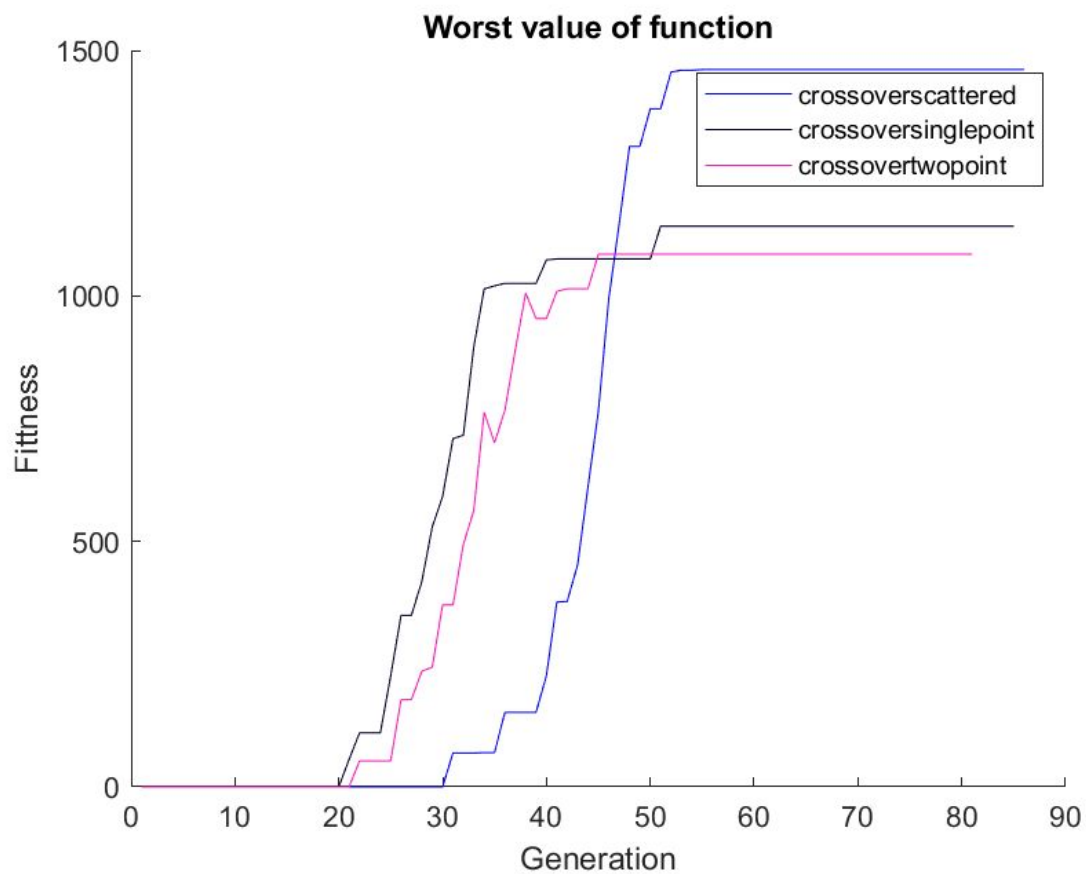
36	44	0,6	0	0	0
37	99	0,5	1	99	0,5
38	57	0,5	0	0	0
39	32	0,4	0	0	0
40	36	1	0	0	0
41	97	0,2	1	97	0,2
42	63	0,4	1	63	0,4
43	42	0,9	0	0	0
44	46	0,6	0	0	0
45	100	0,3	1	100	0,3
46	98	0,8	1	98	0,8
47	18	0,3	0	0	0
48	75	0,9	0	0	0
49	83	0,2	1	83	0,2
50	66	0,4	1	66	0,4
51	79	1	0	0	0
52	13	0,6	0	0	0
53	48	0,5	1	48	0,5
54	50	0,2	1	50	0,2
55	98	0,5	1	98	0,5
56	93	0,8	1	93	0,8
57	58	0,8	0	0	0
58	79	0,5	1	79	0,5
59	59	0,6	0	0	0
60	30	0,2	1	30	0,2
61	32	0,5	0	0	0
62	40	1	0	0	0
63	68	0,7	0	0	0
64	25	0,5	0	0	0
		W = 10,56		2003	10,5

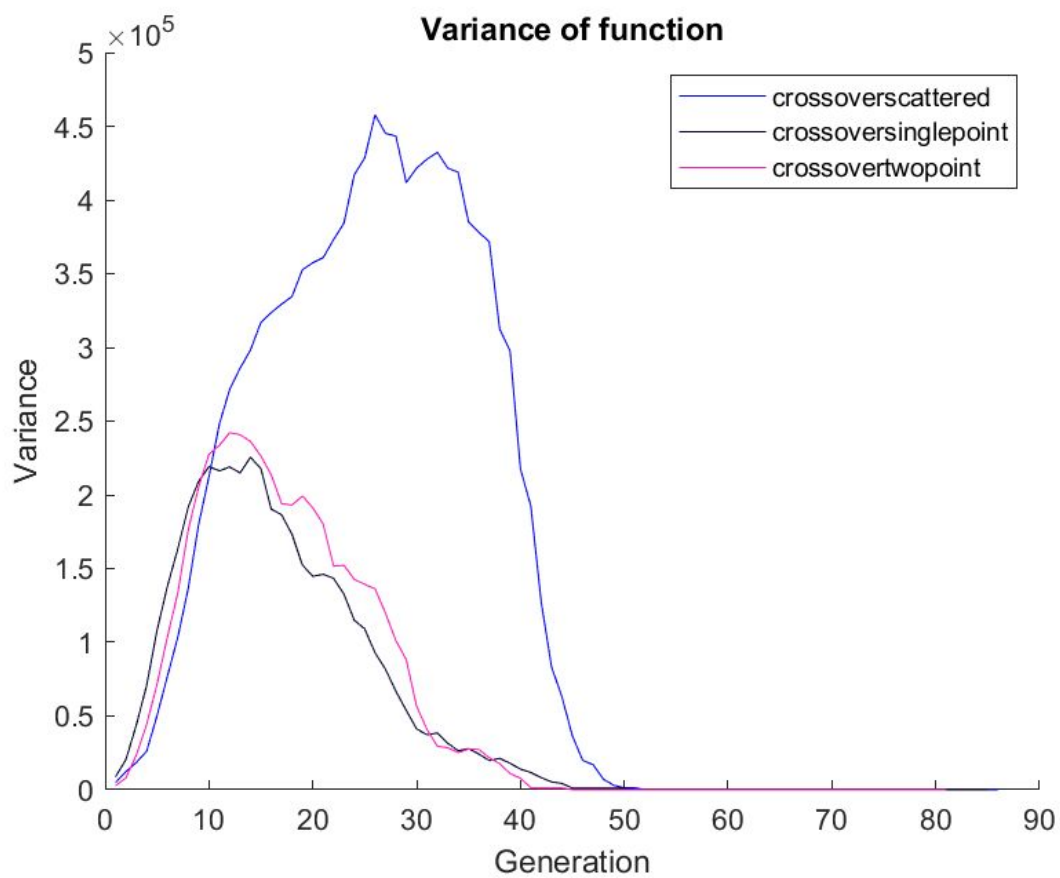
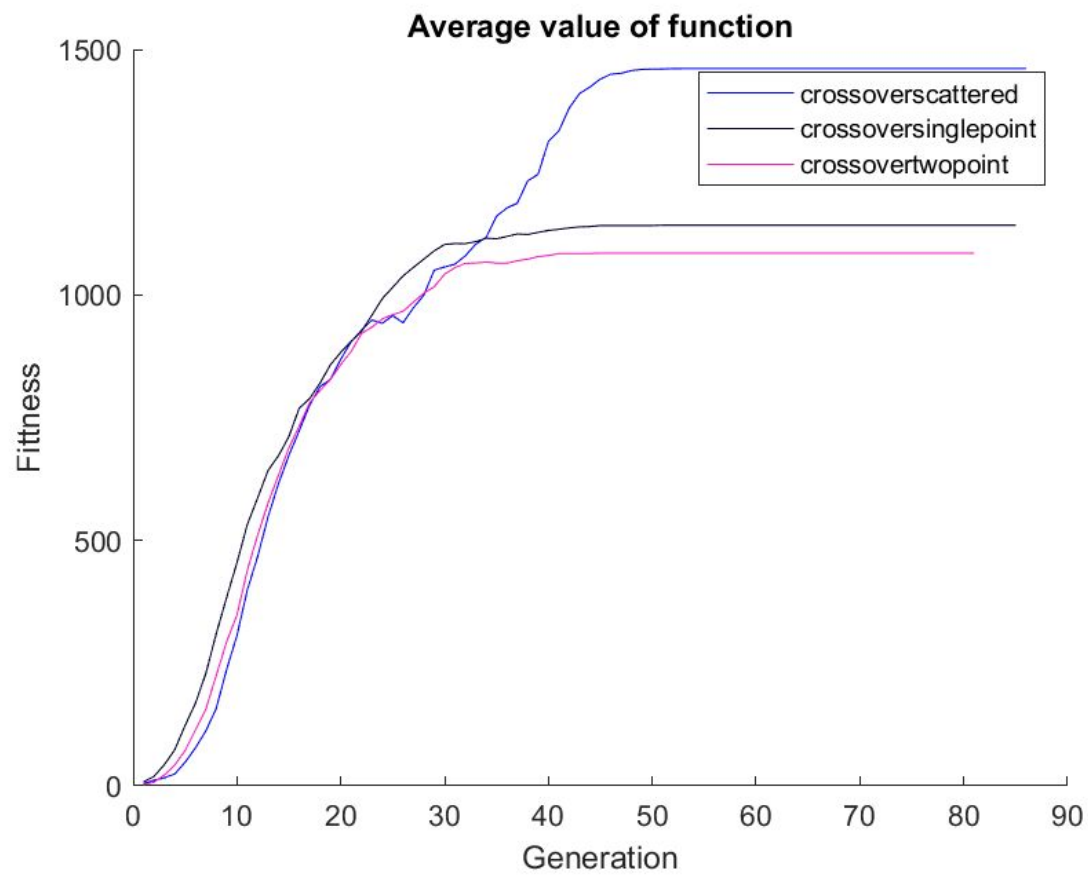
2.2.1. Wpływ metody selekcji



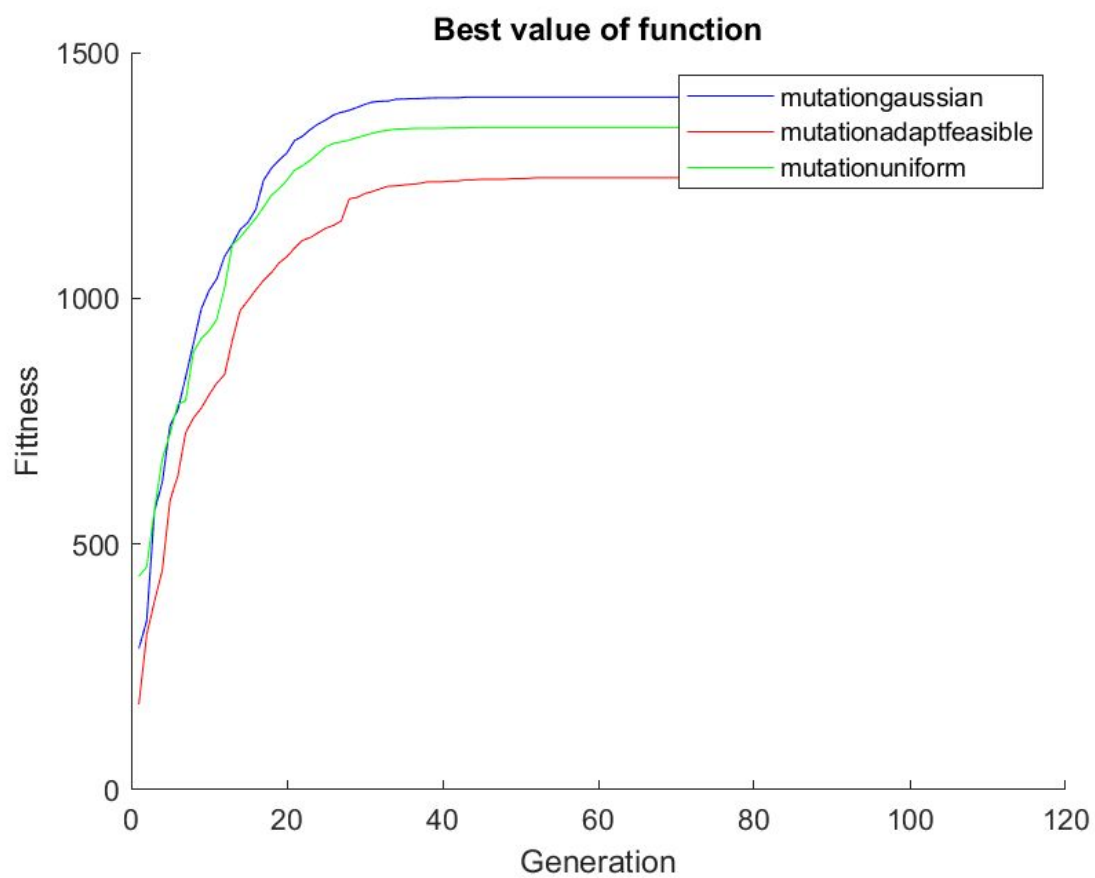
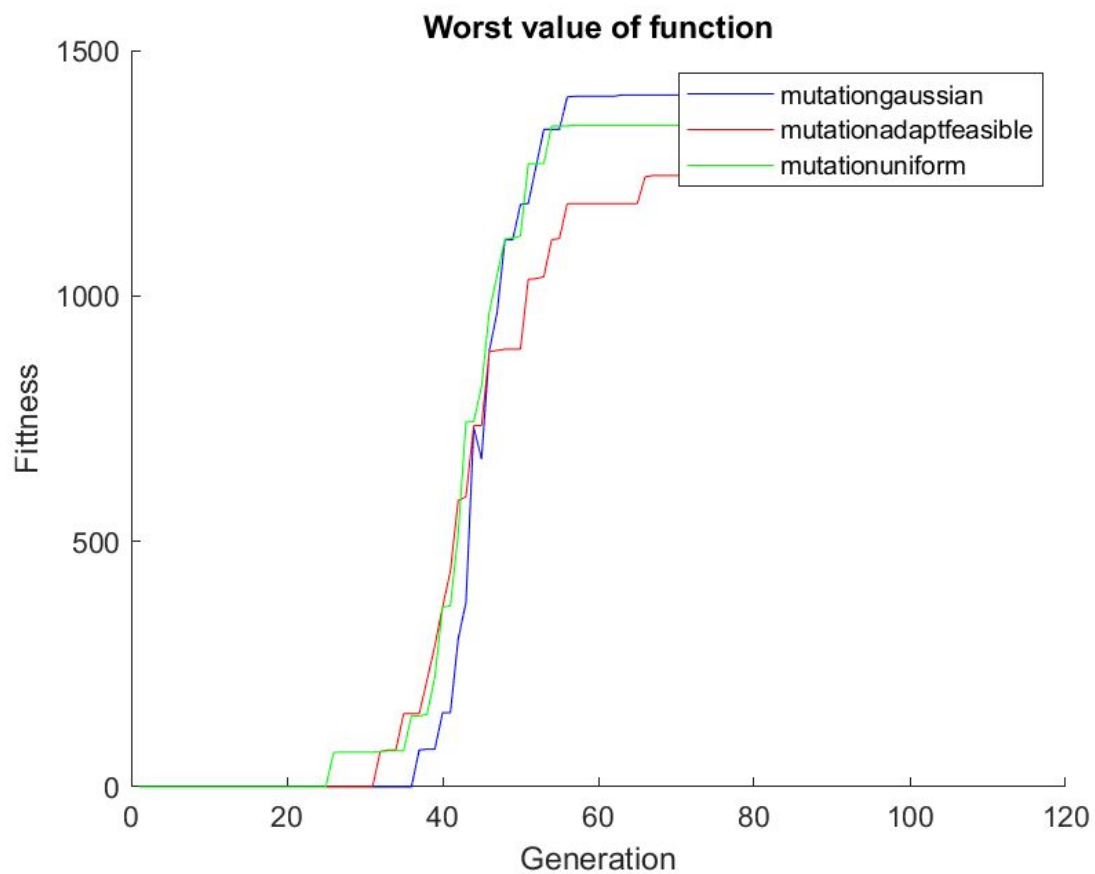


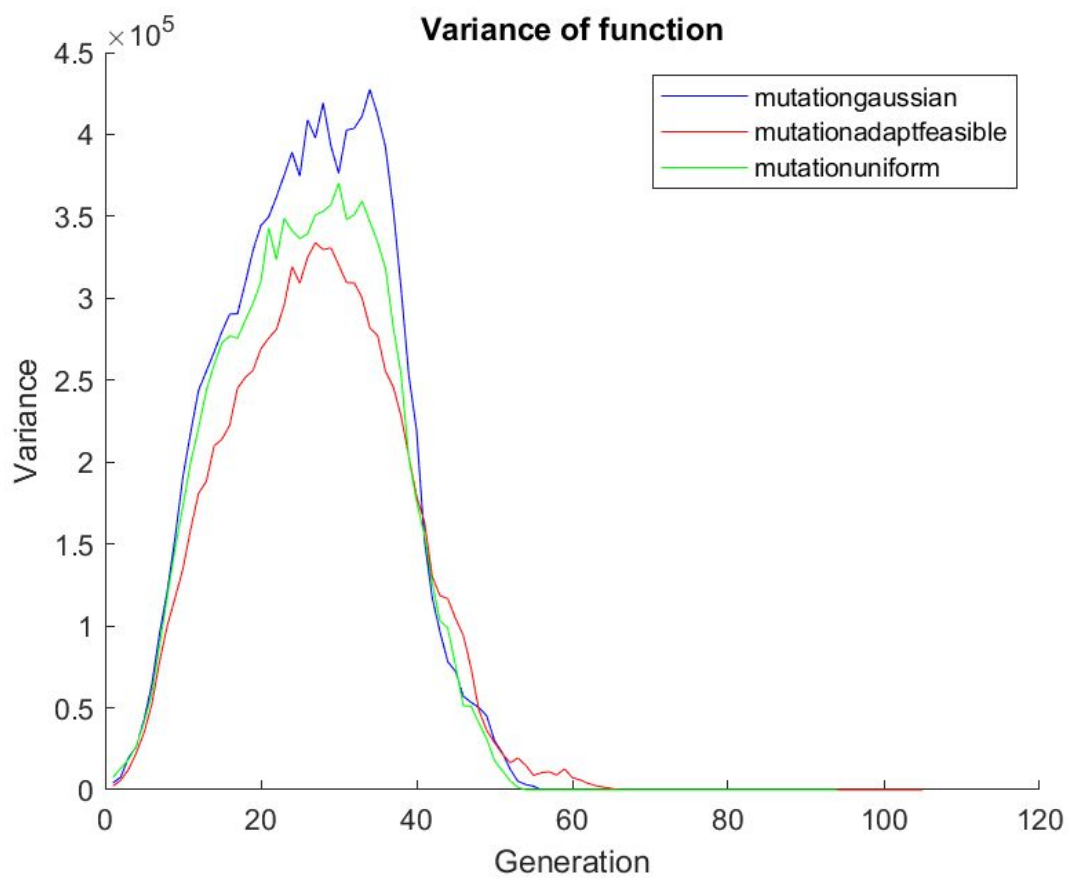
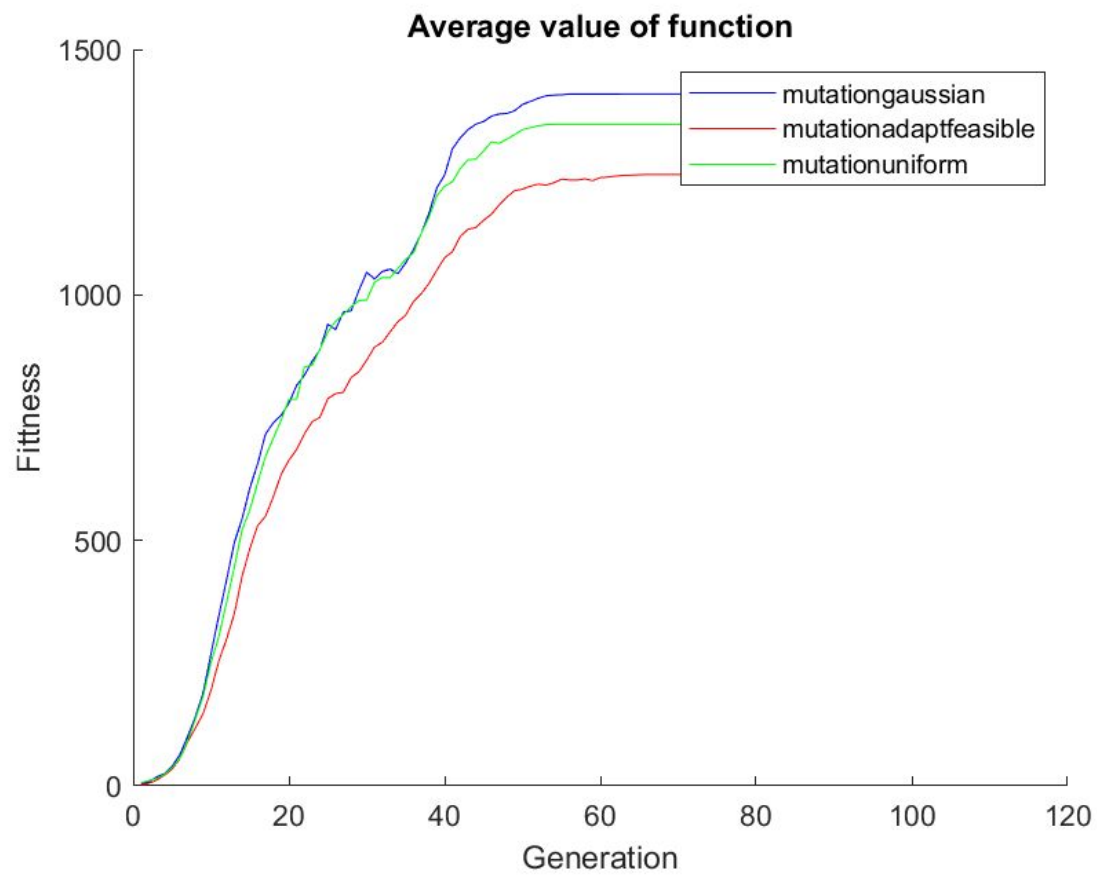
2.2.2. Wpływ metody krzyżowania





2.2.3. Wpływ metody mutacji





3. Wnioski

Największa różnica pomiędzy 32 elementami, a 64 elementami jest zauważalna w ilości optimumów lokalnych. Jest to zauważalne że na wykresach najlepsze rozwiązanie jest daleko od znalezionej optimum globalnego. W przypadku 64 elementów średnia z wywołań wynosi około 60–70% najlepszego znalezionej rozwiązania, a w przypadku 32 elementów średnia z wywołań wynosi około 90%. Dlatego aby znaleźć najlepsze rozwiązanie globalne należy uruchomić algorytm genetyczny wielokrotnie, tym więcej im więcej posiadamy zmiennych. W przypadku z 64 elementami zauważyłem też, że nawet wywoływanie kilkadziesiąt razy nie zawsze znajdowało najlepsze ze wszystkich znalezionych (we wszystkich testach - rozwiązanie podane jako globalne) rozwiązań.

Podczas wyboru najlepszej metody selekcji możemy zauważyć, że większość metod nie różni się znacząco między sobą wynikami. Wyjątkiem jest tylko selekcja równomierna, która w żaden sposób nie weryfikuje przydatności wybieranych jednostek do krzyżowania i mutacji. Ten rodzaj selekcji zapewnia znacznie gorsze wyniki i niejednokrotnie w trakcie testów nie był w stanie znaleźć optimum w wyznaczonej ilości generacji co świadczy o jego wolniejszej zbieżności. Do dalszych badań wybrałem selekcję stochastyczną równomierną.

Wybór najlepszej metody krzyżowania pokazał, że krzyżowanie jedno i dwu-punktowe daje bardzo podobne rezultaty, które zdają się różnić tylko ze względu na losowość algorytmu. Krzyżowanie rozproszone daje natomiast najlepsze rezultaty ze wszystkich testowanych metod. Znajduje statystycznie lepsze rozwiązania i szybciej znajduje lepsze wartości. Ponadto jego najgorsze dopasowanie wolniej zbiega do maksimum, a wariancja jest znacznie większa, co świadczy o szerszym przeszukiwaniu przed utworzeniem populacji homogenicznej. Do dalszych celów posługiwałem się metodą wielopunktowego krzyżowania.

Testowanie różnych metod mutacji w przypadku 32 elementów nie przyniosło żadnych wymiernych rezultatów. Wszystkie są w przybliżeniu takie same - z dokładnością do faktu iż algorytm genetyczny działał losowo. Natomiast w przypadku 64 elementów możemy zauważyć wyraźną przewagę mutacji o rozkładzie normalnym nad rozkładem równomiernym i adaptacyjnym. Mutacja gaussowska daje lepsze rezultaty pod każdym kątem: statystycznie lepsze rozwiązania, nieznacznie szybsze zbieranie i większa wariancja wśród populacji.

Jako najlepsze połączenie wybrałem selekcję stochastyczną równomierną, krzyżowanie wielopunktowe i mutację o rozkładzie gaussowskim. Warto zauważyć, że są to również domyślne wartości z jakimi uruchamiany jest w MATLABie algorytm genetyczny, jeśli nie podamy własnych preferencji.