WydziaĆ Elektroniki i Technik Informacyjnych

Politechnika Warszawska

Przeszukiwanie i optymalizacja

Semestr 23Z

Sprawozdanie z projektu

Temat nr 6
Algorytm roju czstek z modyfikacjami

Natalia Iwaska, MikoĆaj Wewir

Warszawa, 2024

Spis treści

1. WstŹp

Celem naszego projektu jest implementacja algorytmu roju czstek z modyfikacjami dotyczcymi wspĆczynnika bezwĆadnoŻci, a nastŹpnie porwnanie wynikw dziaĆania algorytmu z jego podstawow wersj.

PracŹ rozpoczŹliŻmy nad zaimplementowaniem standardowej wersji algorytmu. Gdy udaĆo siŹ zrealizowa tŹ czŹŻ, musieliŻmy siŹ zastanowi nad moliwymi modyfikacjami wspĆczynnika bezwĆadnoŻci. Po odpowiednim przejrzeniu literatury doszliŻmy do dwch gĆwnych sposobw zmiany tego parametru.

Pierwszym z nich okazaĆa siŹ zmiana wartoŻci wspĆczynnika wraz z kolejnymi iteracjami. MieliŻmy tutaj do rozwaenia dwie kwestie. Pierwsz z nich byĆ wybr funkcji, ktra opisywaĆaby zmianŹ parametru w czasie - w naszym przypadku padĆo na najprostsz realizacjŹ, czyli funkcjŹ liniow. Drugim kryterium byĆo to czy funkcja jest rosnca czy malejca w czasie; sprawdziliŻmy dwie opcje.

Innym sposobem na zmiennoŻ wspĆczynnika bezwĆadnoŻci w czasie byĆa modyfikacja, ktra przypisywaĆa odpowiednie wartoŻci w zalenoŻci od jakoŻci rozwizania. Tak jak w pierwszym przypadku tutaj te mogliŻmy wybra to, czy parametr ma siŹ zwiŹksza, czy zmniejsza w zalenoŻci od wyniku. ZdecydowaliŻmy siŹ na jedn z opcji mianowicie na malejcy wspĆczynnik, dla lepszych rozwiza.

1.1. Opis algorytmu roju czstek

Algorytm Roju Czstek (PSO, ang. *particle swarm optimization*) jest metaheurystycznym algorytmem optymalizacyjnym, inspirowanym zachowaniem stad w przyrodzie, ktry na pocztku sĆuyĆ do symulowania zachowa spoĆecznych wystŹpujcych wŻrd Ćawic ryb, czy kluczy ptakw.

Dla danej funkcji celu $f(\mathbf{x})$, gdzie $\mathbf{x}$ to wektor zmiennych decyzyjnych, PSO korzysta z populacji czstek poruszajcych siŹ w przestrzeni poszukiwa, w celu znalezienia optymalnego rozwizania problemu. Kada czstka reprezentuje pozycjŹ oraz potencjalne rozwizanie (wartoŻ w tym punkcie) problemu.

Pozycja czstki w przestrzeni poszukiwa jest okreŻlana przez wektor $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, gdzie n to liczba zmiennych decyzyjnych.

Ruch czstki jest aktualizowany na podstawie dwch wpĆyww: jej wĆasnego najlepszego rozwizania (lokalne) oraz najlepszego rozwizania w caĆej populacji (globalne).

Aktualizacja pozycji czstki jest opisana rwnaniem:

$$\mathbf{v}_i^{t+1} = w \cdot \mathbf{v}_i^t + c_1 \cdot r_1 \cdot (\mathbf{p}_i^t - \mathbf{x}_i^t) + c_2 \cdot r_2 \cdot (\mathbf{g}^t - \mathbf{x}_i^t) \tag{1.1}$$

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{v}_i^{t+1} \tag{1.2}$$

gdzie:

— $\mathbf{v}_i^t$ to prŹdkoŻ czstki w chwili t,
— $\mathbf{p}_i^t$ to najlepsze dotd osigniŹte przez czstkŹ rozwizanie,
— $\mathbf{g}^t$ to najlepsze dotd osigniŹte w populacji globalne rozwizanie,
— w to wspĆczynnik bezwĆadnoŻci,
— c_1 i c_2 to wspĆczynniki akceleracji,
— r_1 i r_2 to losowe liczby z przedziaĆu $(0, 1)$.

1.2. Implementacja algorytmu

Algorytm zostaĆ zaimplementowany gĆwnie w funkcji `PSO.py` w kodach rdĆowych doĆczonych do dokumentacji projektowej.

1.2.1. Inicjalizacja

Inicjalizacja algorytmu obejmuje wygenerowanie 25 czstek, a nastŹpnie przypisanie kadej czsteczce losowej pocztkowej pozycji w przestrzeni poszukiwa, gdzie kada zmienna decyzyjna jest ustawiana na losow wartoŻ z zakresu dziedziny badanej funkcji. Dodatkowo, inicjalizowane s rwnie losowe pocztkowe prŹdkoŻci czsteczki, a take ustawiany jest atrybut reprezentujcy najlepszy osigniŹty wynik, ktry pocztkowo jest ustawiony na nieskoczonoŻ. Te inicjalne parametry stanowi punkt wyjŻcia dla procesu optymalizacji rojem czsteczek. PozostaĆe parametry algorytmu:

— wspĆczynnik bezwĆadnoŻci : 0.72984
— liczba czsteczek: 25
— liczba testw: 500
— maksymalna liczba iteracji: 60
— wspĆczynniki akceleracji (c_1 i c_2): 1.496

1.2.2. Ograniczenia

W celu skutecznej obsĆugi naruszonych ogranicze dziedziny badanych funkcji w algorytmie optymalizacyjnym, zdecydowaliŻmy siŹ zaimplementowa technikŹ rzutowania. Ta metoda umoliwiĆa utrzymanie rozwizania w obszarze dopuszczalnym, poprzez rzutowanie wspĆrzŹdnych znajdujcych siŹ poza dziedzin funkcji na wspĆrzŹdne graniczne. PrzyczyniĆo siŹ to do poprawy dziaĆania algorytmu.

1.3. Modyfikacje

Poniewa projekt dotyczy badania wpĆywu zmiany parametru bezwĆadnoŻci czsteczek na wyniki dziaĆania algorytmu, wszystkie pozostaĆe parametry pozostawiliŻmy jednakowe. PrzyjŹte one zostaĆy na podstawie rekomendowanych parametrw, opracowanych w badaniu przez M. Clerc'a oraz J. Kennedy'ego. W podstawowej wersji algorytmu wspĆczynnik bezwĆadnoŻci jest staĆy. ZaimplementowaliŻmy trzy rne modyfikacje algorytmu, ktre sprawiaj, e wspĆczynnik ten jest zmienny w czasie.

1.3.1. Modyfikacja 1

W pierwszej modyfikacji wartoŻ wspĆczynnika roŻnie wraz z kad iteracj algorytmu.

$$\mathbf{w} = w_{\text{init}} \cdot \frac{t}{t_{\text{max}}} + 0.2 \tag{1.3}$$

gdzie:

— w_{init} to pocztkowa wartoŻ wspĆczynnika bezwĆadnoŻci rwna 0.72984
— t to numer aktualnej iteracji
— t_{max} to maksymalna liczba iteracji w naszym przypadku rwna 60

WspĆczynnik startuje ze wspĆczynnikiem bezwĆadnoŻci rwnym 0.2 i z kada iteracj roŻnie, a do wartoŻci rwnej 0.72984+0.2 czyli 0.92984

1.3.2. Modyfikacja 2

W drugiej modyfikacji wartoŻ wspĆczynnika bezwĆadnoŻci zaley od jakoŻci rozwizania. Im jakoŻ rozwizania jest blisza najlepszemu globalnemu rozwizaniu tym inercja ma mniejsza wartoŻ.

$$\mathbf{w} = (w_{\text{init}} + 0.1) \cdot \frac{dif}{dif + 1} \tag{1.4}$$

gdzie:

— dif to rnica miŹdzy najlepszym globalnym wynikiem, a najlepszym dotychczasowym wynikiem czstki.
— w_{init} to pocztkowa wartoŻ wspĆczynnika bezwĆadnoŻci rwna 0.72984

1.3.3. Modyfikacja 3

W drugiej modyfikacji wartoŻ wspĆczynnika maleje wraz z kad iteracj algorytmu.

$$\mathbf{w} = w_{\text{init}} \cdot \frac{t_{\max} - t}{t_{\max}} + 0.2 \tag{1.5}$$

gdzie:

— w_{init} to pocztkowa wartoŻ wspĆczynnika bezwĆadnoŻci rwna 0.72984

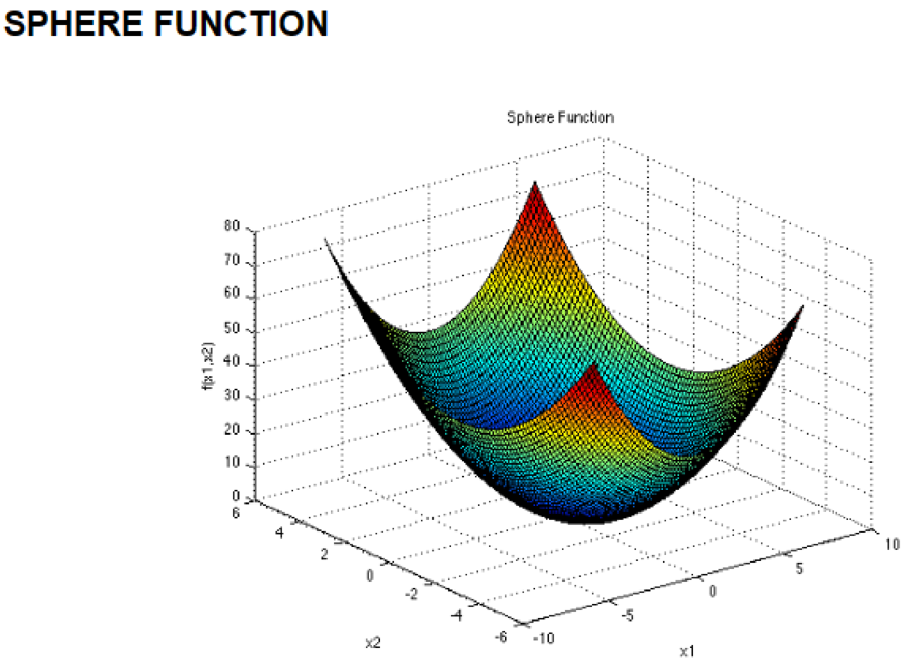
WspĆczynnik startuje ze wspĆczynnikiem bezwĆadnoŻci rwnym 0.92984 i z kad iteracj maleje, a do wartoŻci rwnej 0.2.

W dalszej czŹŻci dokumentu rozrnienie poszczeglnych trybw wyglda nastŹpujco:

— tryb 1 - brak modyfikacji - staĆa inercja rwna 0.72984
— tryb 2 - modyfikacja 1. inercja rosnca liniowo w czasie
— tryb 3 - modyfikacja 2. inercja malejca wraz z popraw jakoŻci rozwizania.
— tryb 4 - modyfikacja 3. inercja malejca liniowo w czasie

2. Opis testw i funkcji

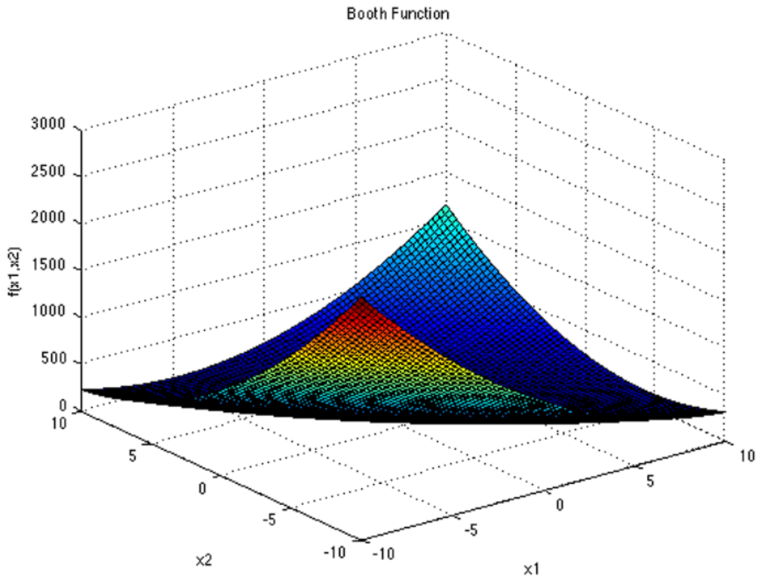
Testy dla algorytmu wykonaliŻmy dla szeŻciu rnych funkcji dwuwymiarowych, tj. wielomianowej (sferycznej), Bootha, Rosenbrocka, Ackley'a, Michalewicza oraz funkcji holder-table. Kada z modyfikacji algorytmu szukaĆa najlepszego rozwizania dla kadej z powyszych funkcji. Po zakoczonych testach dla kadej z funkcji analizowane byĆy wyniki oraz porwnywane ich efekty. Poniej znajduj siŹ wykresy tych funkcji:



$$f(\mathbf{x}) = \sum_{i=1}^{d} x_i^2$$

Rys. 2.1: funkcja wielomianowa

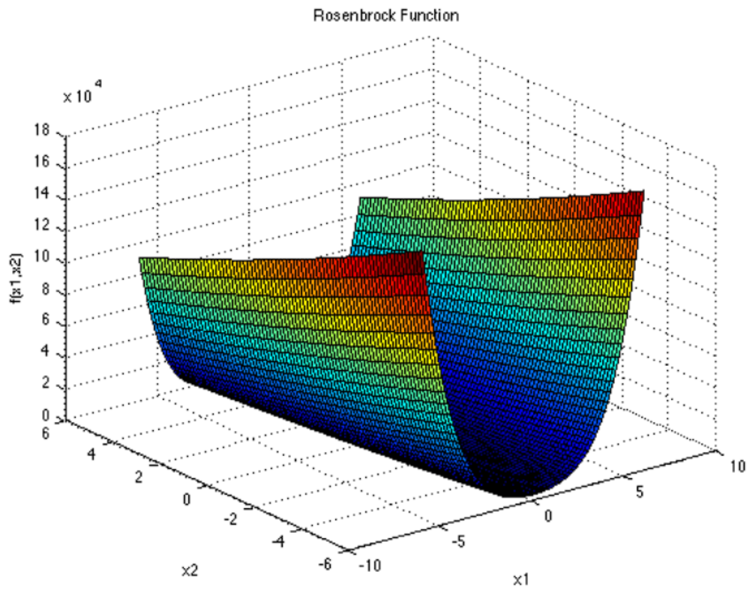
BOOTH FUNCTION



$$f(\mathbf{x}) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$$

Rys. 2.2: funkcja Bootha

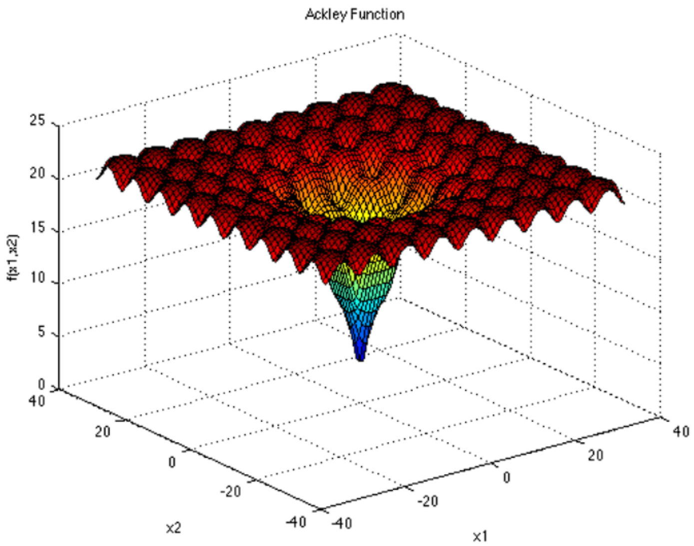
ROSENBROCK FUNCTION



$$f(\mathbf{x}) = \sum_{i=1}^{d-1} \left[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right]$$

Rys. 2.3: funkcja Rosenbrocka

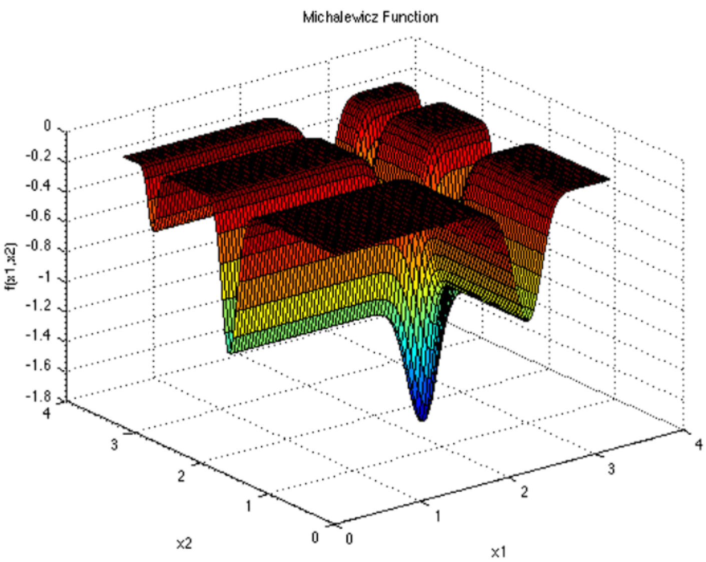
ACKLEY FUNCTION



$$f(\mathbf{x}) = -a \exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right) + a + \exp(1)$$

Rys. 2.4: funkcja Ackley'a

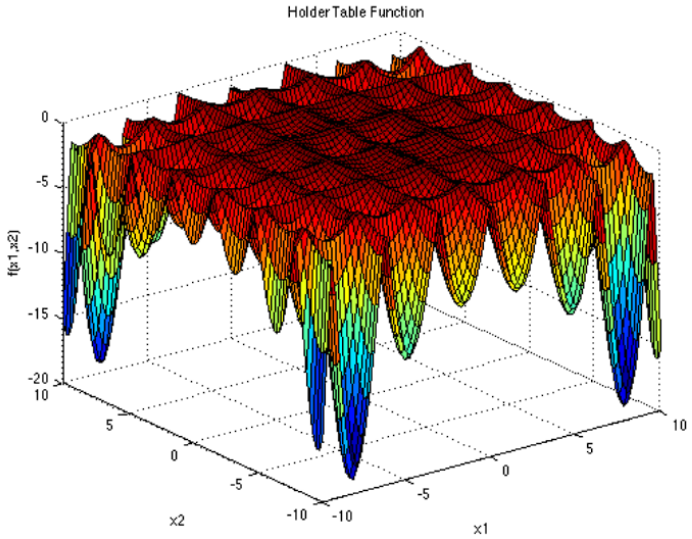
MICHALEWICZ FUNCTION



$$f(\mathbf{x}) = -\sum_{i=1}^{d} \sin(x_i) \sin^{2m}\left(\frac{ix_i^2}{\pi}\right)$$

Rys. 2.5: funkcja Michalewicza

HOLDER TABLE FUNCTION



$$f(\mathbf{x}) = -\left|\sin(x_1)\cos(x_2)\exp\left(\left|1-\frac{\sqrt{x_1^2+x_2^2}}{\pi}\right|\right)\right|$$

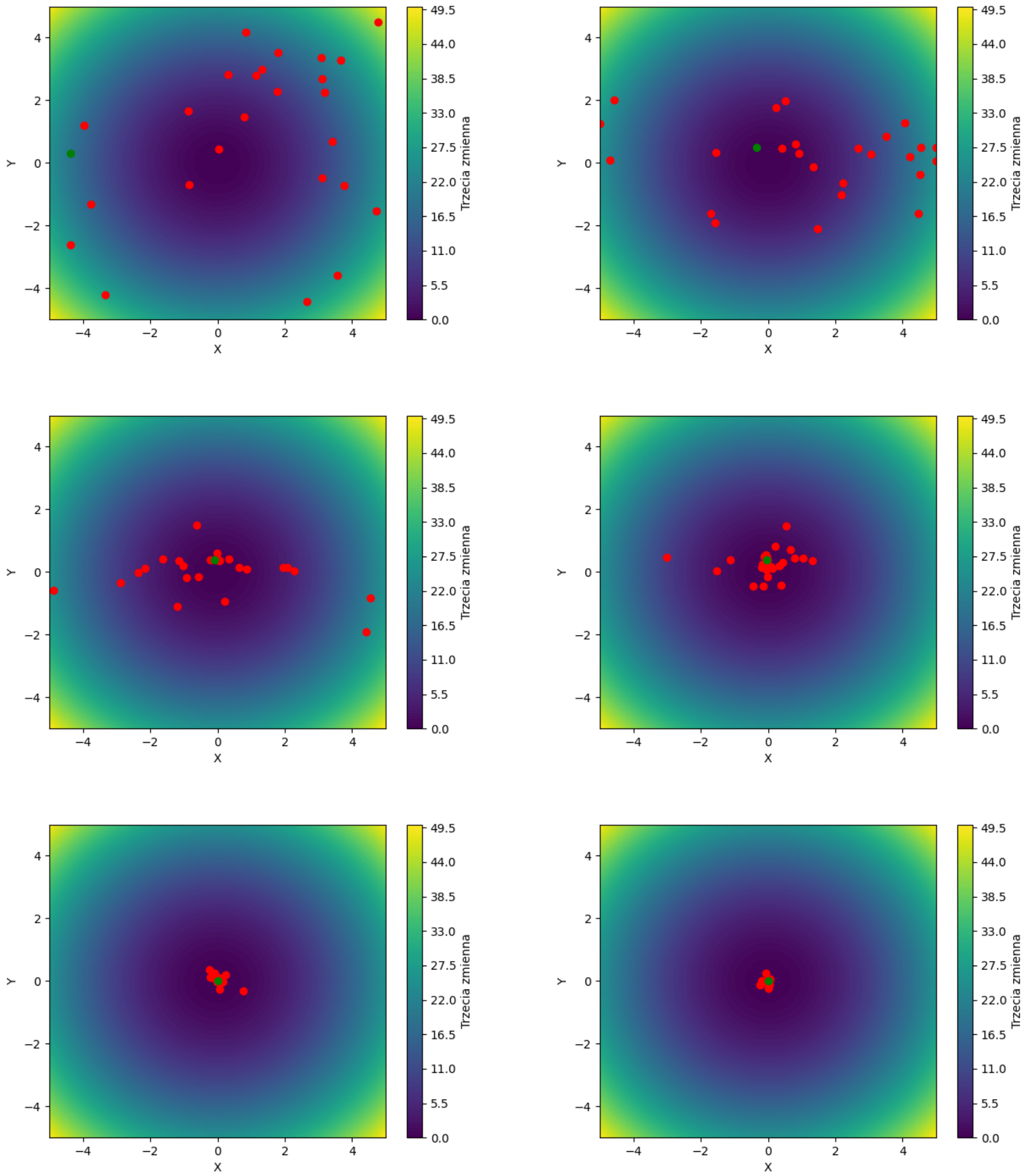
Rys. 2.6: funkcja Holder-Table

3. Przebiegi

Poniej zostaĆy przedstawione fragmenty wizualizacji przebiegw algorytmu w formie zrzutw wykresu funkcji z zaznaczonymi czsteczkami w danej iteracji (czerwone punkty) oraz najlepszego globalnego rozwizania z poprzedniej iteracji (zielony punkt).

W tabelach znajduj siŹ otrzymane wyniki naszych testw. Testy polegaĆy na uruchomieniu algorytmu w kadym trybie i dla kadej funkcji po 500 razy. Z powodu pojedynczych znaczcych rozbienoŻci miŹdzy wynikami naszych testw a oczekiwanymi wartoŻciami, postanowiliŻmy eliminowa 5% najgorszych rezultatw w celu uzyskania bardziej reprezentatywnych danych. Ponisze tabele zawieraj kolumnŹ przedstawiajc najgorszy uzyskany wynik, jednak pozostaĆa analiza (miŹdzy innymi Żredni wynik) caĆkowicie pomija wyeliminowane wartoŻci.

3.1. Funkcja Wielomianowa

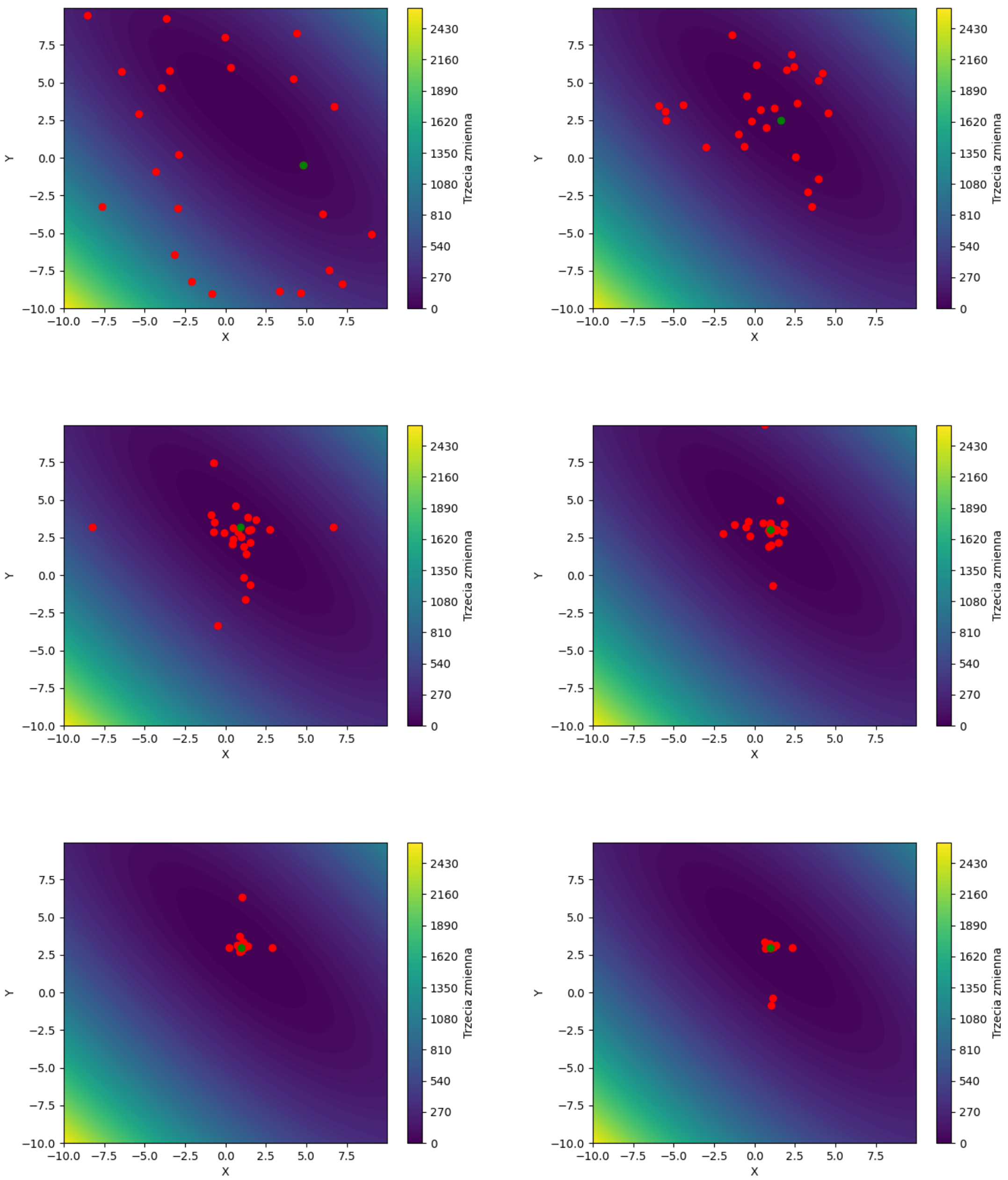


Dziedzina przeszukiwa = [-5, 5]. Optymalna wartoŻ = 0, dla punktu (0, 0).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$3{,}36 \cdot 10^{-13}$	$3{,}18 \cdot 10^{-7}$	$1{,}7 \cdot 10^{-2}$	$5{,}17 \cdot 10^{-7}$	1
$1{,}19 \cdot 10^{-18}$	$2{,}35 \cdot 10^{-14}$	3,71	$3{,}40 \cdot 10^{-14}$	2
$\mathbf{8{,}02 \cdot 10^{-42}}$	$\mathbf{4{,}59 \cdot 10^{-15}}$	1,13	$2{,}09 \cdot 10^{-13}$	3
$9{,}92 \cdot 10^{-10}$	$1{,}83 \cdot 10^{-12}$	$8{,}26 \cdot 10^{-5}$	$3{,}03 \cdot 10^{-12}$	4

Tab. 3.1: Porwnanie wynikw dla funkcji wielomianowej w rnych trybach.

3.2. Booth

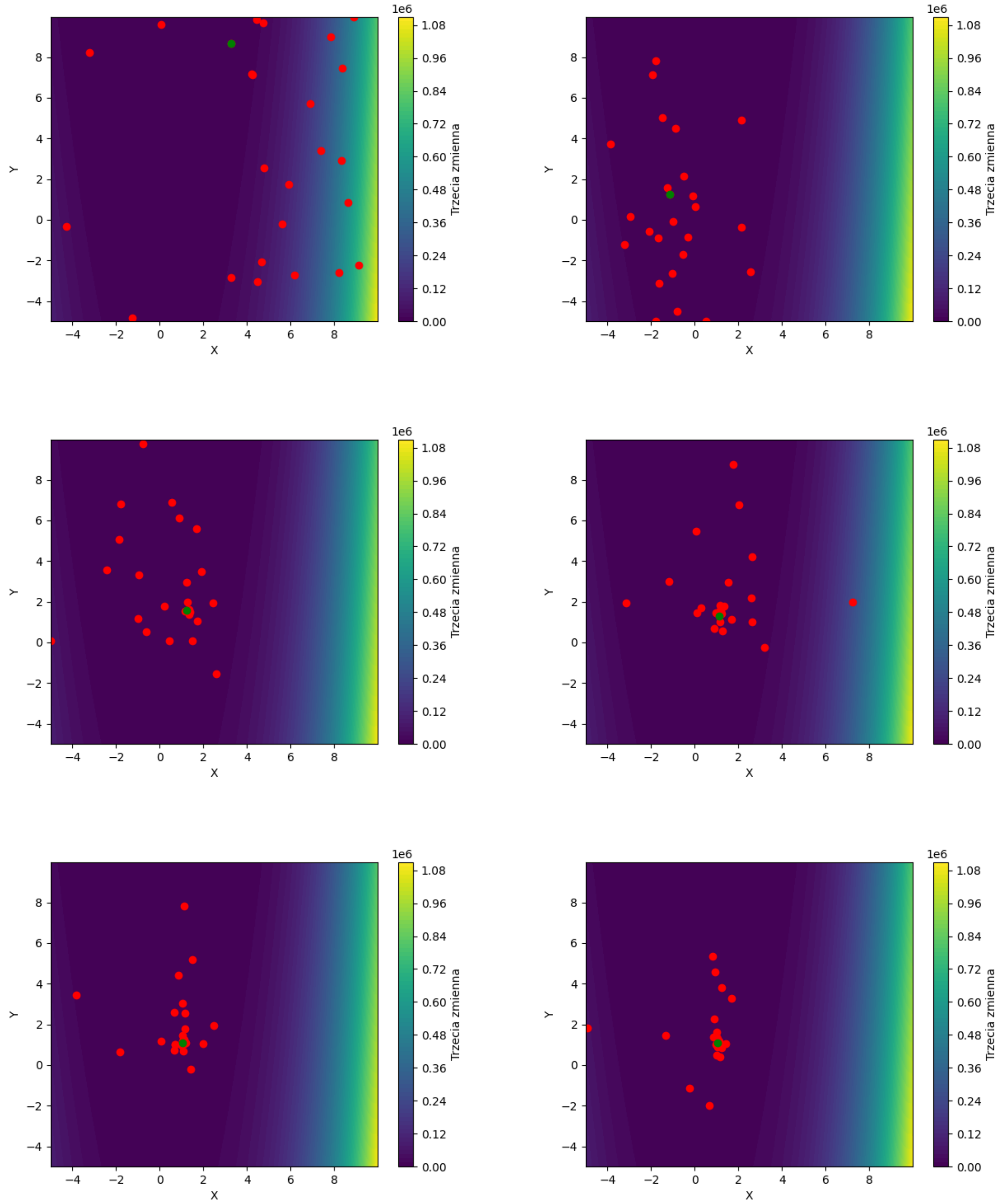


Dziedzina przeszukiwa = [-10, 10]. Optymalna wartoŻ = 0, dla punktu (1, 3).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$2{,}01 \cdot 10^{-9}$	$3{,}29 \cdot 10^{-6}$	$1{,}00 \cdot 10^{-4}$	$5{,}51 \cdot 10^{-6}$	1
$5{,}81 \cdot 10^{-16}$	$\mathbf{8{,}97 \cdot 10^{-13}}$	$5{,}25 \cdot 10^{-11}$	$1{,}43 \cdot 10^{-12}$	2
$\mathbf{3{,}32 \cdot 10^{-27}}$	$1{,}02 \cdot 10^{-6}$	$1{,}91 \cdot 10^{-3}$	$6{,}14 \cdot 10^{-6}$	3
$3{,}00 \cdot 10^{-14}$	$6{,}84 \cdot 10^{-11}$	$3{,}54 \cdot 10^{-9}$	$9{,}27 \cdot 10^{-11}$	4

Tab. 3.2: Porwnanie wynikw dla funkcji Booth w rnych trybach.

3.3. Rosenbrock

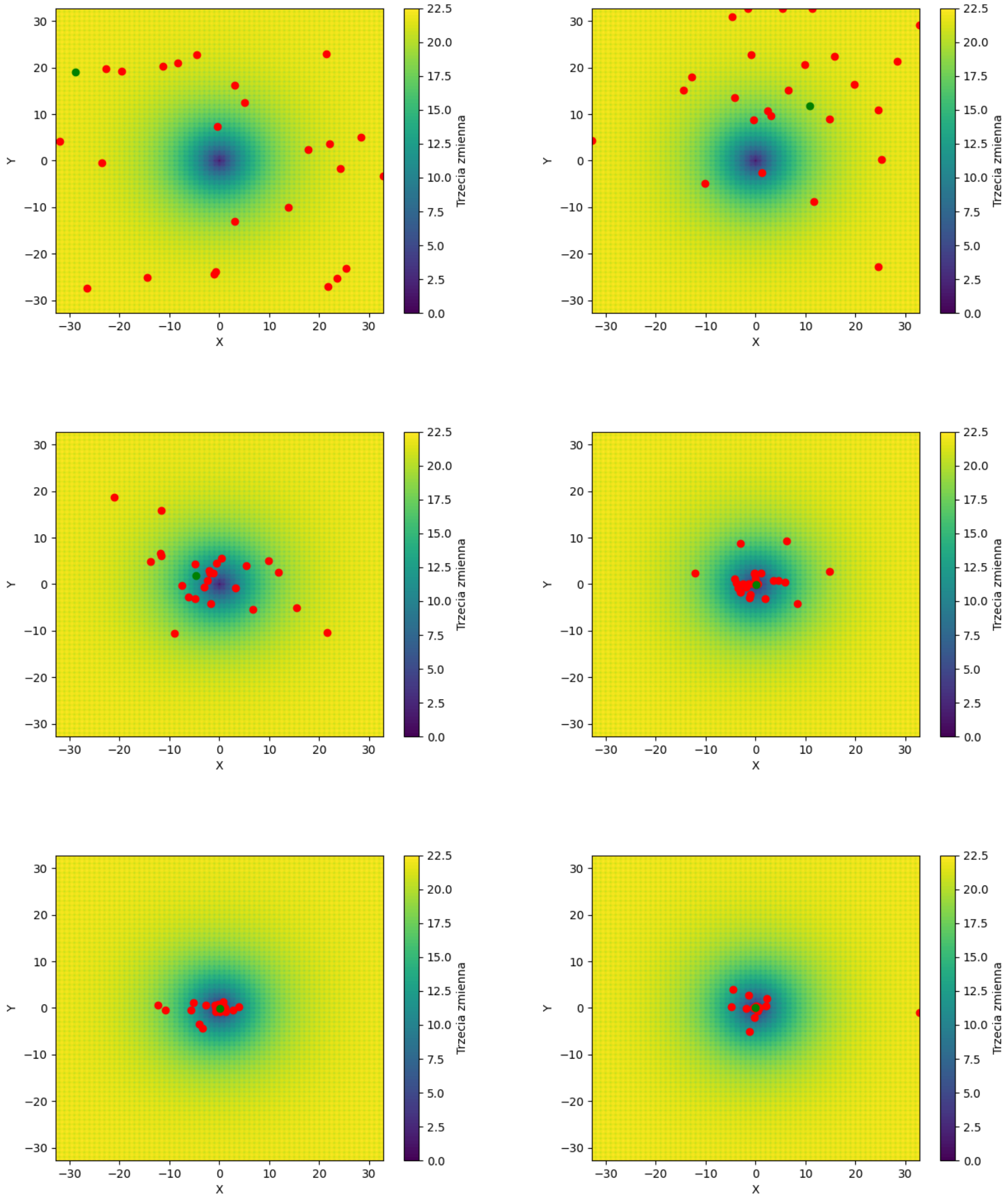


Dziedzina przeszukiwa = [-5, 10]. Optymalna wartoŻ = 0, dla punktu (1, 1).

Najlepszy	**redni**	**Najgorszy**	**Odchylenie standardowe**	**Tryb**
$1{,}27 \cdot 10^{-7}$	$\mathbf{1{,}36 \cdot 10^{-2}}$	$1{,}71 \cdot 10^{1}$	$3{,}59 \cdot 10^{-2}$	1
$6{,}81 \cdot 10^{-9}$	$2{,}35 \cdot 10^{-2}$	$1{,}74 \cdot 10^{1}$	$4{,}49 \cdot 10^{-2}$	2
$1{,}94 \cdot 10^{-8}$	$2{,}00 \cdot 10^{-2}$	$5{,}89$	$5{,}14 \cdot 10^{-2}$	3
$\mathbf{6{,}18 \cdot 10^{-9}}$	$2{,}48 \cdot 10^{-2}$	$1{,}48 \cdot 10^{1}$	$1{,}23 \cdot 10^{-1}$	4

Tab. 3.3: Porwnanie wynikw dla funkcji Rosenbrock w rnych trybach.

3.4. Ackley

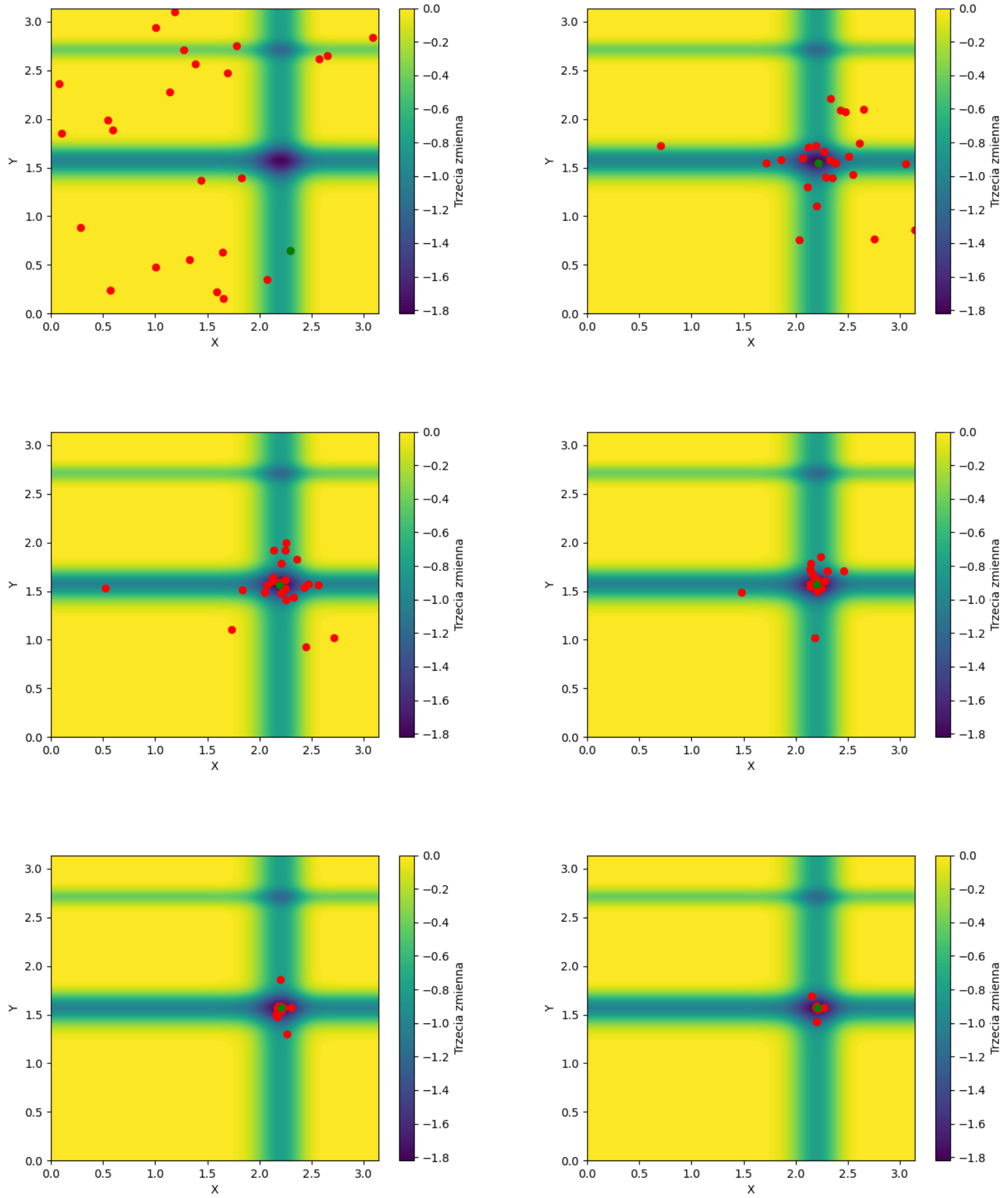


Dziedzina przeszukiwa = [-32.768, 32.768]. Optymalna wartoŻ = 0, dla punktu (0, 0).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$1{,}02 \cdot 10^{-4}$	$8{,}10 \cdot 10^{-3}$	$1{,}59 \cdot 10^{1}$	$7{,}54 \cdot 10^{-3}$	1
$2{,}78 \cdot 10^{-8}$	$\mathbf{1{,}95 \cdot 10^{-6}}$	$1{,}36 \cdot 10^{1}$	$1{,}82 \cdot 10^{-6}$	2
$\mathbf{4{,}94 \cdot 10^{-13}}$	$4{,}20 \cdot 10^{-6}$	$1{,}08 \cdot 10^{-2}$	$3{,}35 \cdot 10^{-5}$	3
$8{,}06 \cdot 10^{-7}$	$2{,}08 \cdot 10^{-5}$	$1{,}19 \cdot 10^{1}$	$1{,}88 \cdot 10^{-5}$	4

Tab. 3.4: Porwnanie wynikw dla funkcji Ackley w rnych trybach.

3.5. Michalewicz

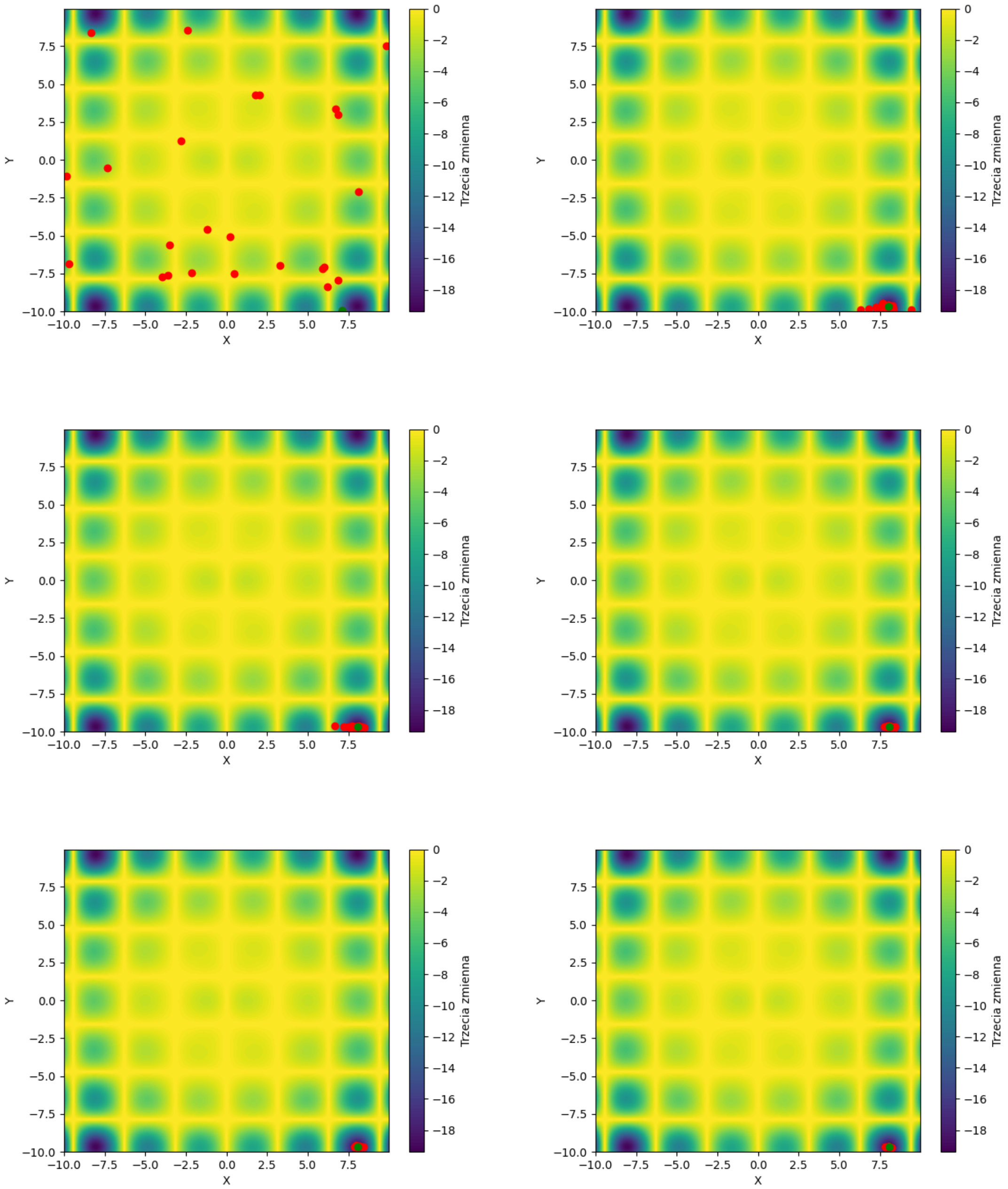


Dziedzina przeszukiwa = [-0, π]. Optymalna wartoŻ = -1.8013, dla punktu (2.20, 1.57).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
-1.801303	-1.801294	-1.0	$5{,}81 \cdot 10^{-7}$	1
-1.801303	**-1.801303**	$-1{,}0$	$6{,}64 \cdot 10^{-13}$	2
-1.801303	**-1.801303**	-1.0	$1{,}52 \cdot 10^{-13}$	3
-1.801303	**-1.801303**	-1.0	$5{,}16 \cdot 10^{-12}$	4

Tab. 3.5: Porwnanie wynikw dla funkcji Michalewicz w rnych trybach.

3.6. Holder Table



Dziedzina przeszukiwa = [-10, 10]. Optymalna wartoŻ = -19.2085, dla punktw: (8.05502, 9.66459), (-8.05502, 9.66459), (-8.05502, 9.66459), (-8.05502, -9.66459).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
-19.2085	-18.8182	-15.14	$5{,}38 \cdot 10^{-1}$	1
-19.2085	**-19.2085**	-11.0695	$1{,}09 \cdot 10^{-1}$	2
-19.2085	-18.9600	-16.27	$4{,}81 \cdot 10^{-1}$	3
-19.2085	-18.8305	-15.14	$5{,}49 \cdot 10^{-1}$	4

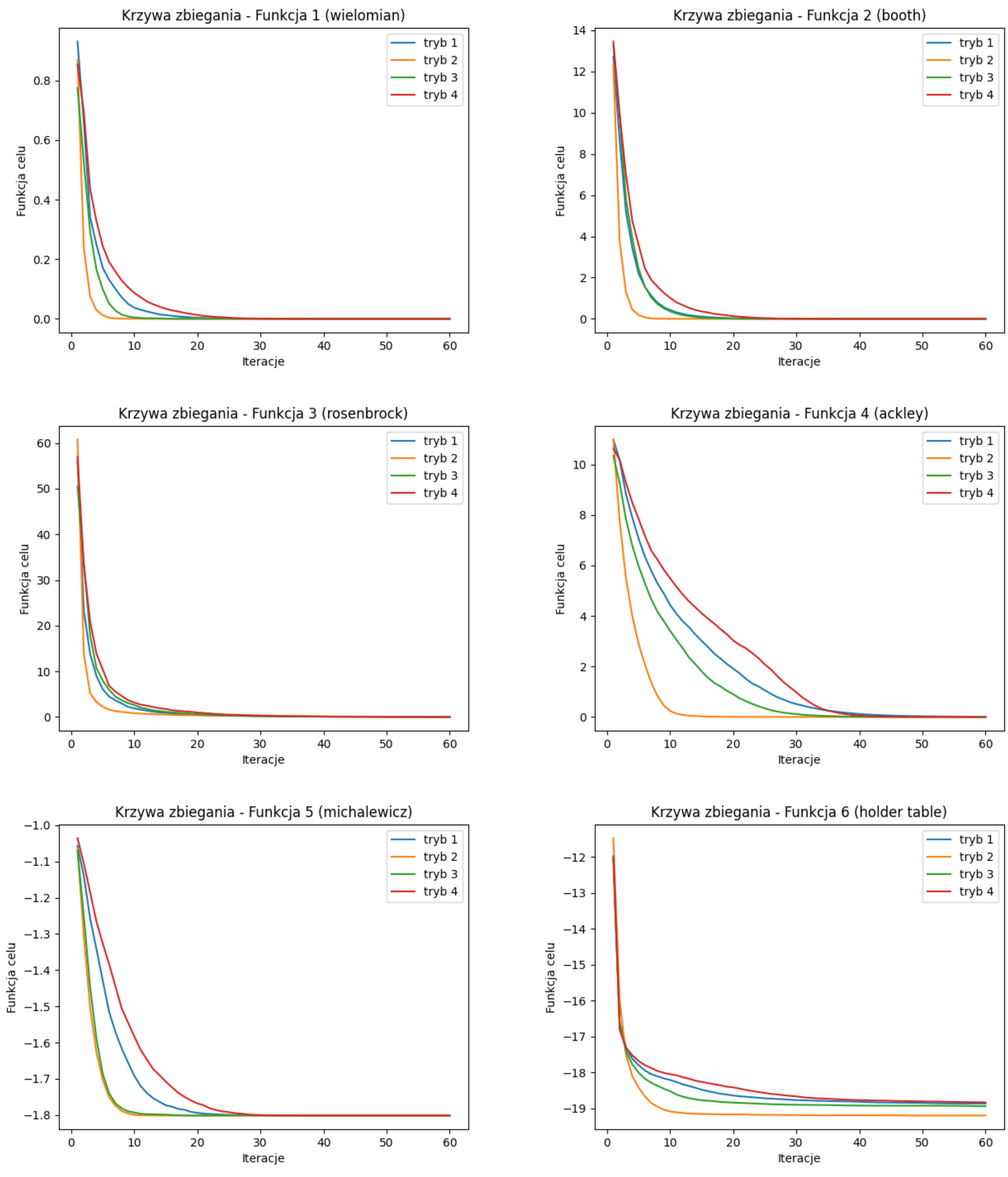
Tab. 3.6: Porwnanie wynikw dla funkcji Holder Table w rnych trybach.

Analiza powyszych wynikw pozwala stwierdzi, e najblisz oczekiwanej wartoŻci uzyskiwaĆ zazwyczaj algorytm dziaĆajcy w trybie 3. Jednak najbliszy oczekiwanej wartoŻci Żredni wynik miaĆ algorytm dziaĆajcy w trybie 2. Oznacza to, e tryb 3 pozwala na wiŹksz eksploatacjŹ przestrzeni przeszukiwa, jednak to trybowi 2 udaĆo siŹ czŹŻciej znale optimum.

4. Analiza wynikw

4.1. Krzywe zbienoŻci

Poniej graficzne przedstawienie zmian wartoŻci funkcji celu w kolejnych iteracjach, ilustrujce proces denia do minimum optymalnej wartoŻci przez algorytm.

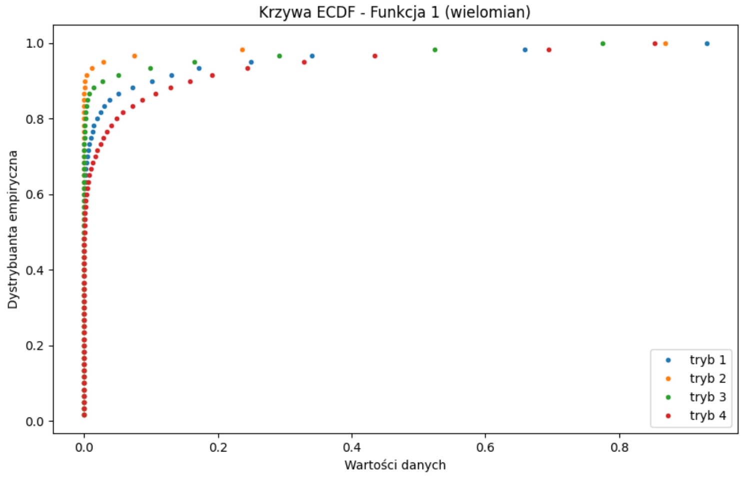


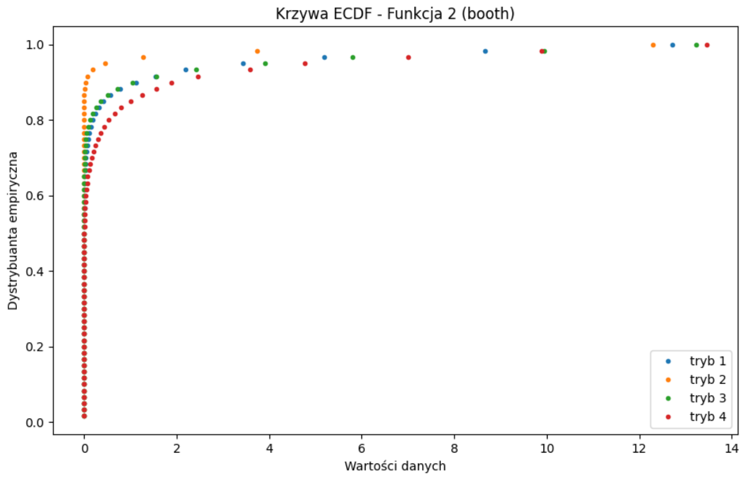
Rys. 4.1: Krzywe zbienoŻci

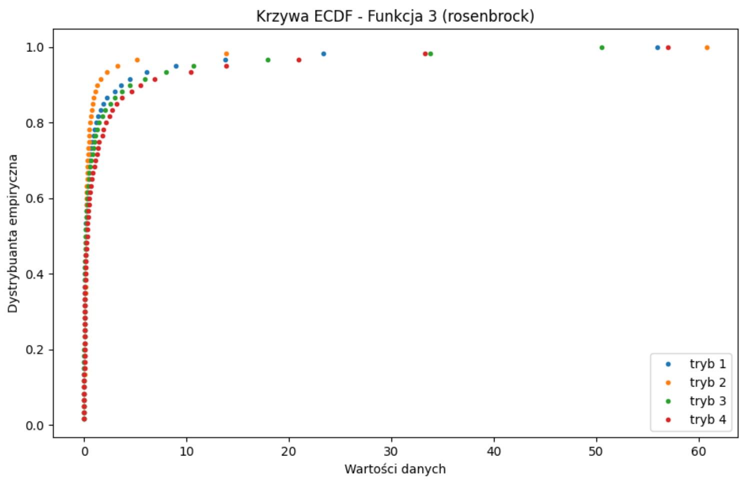
Powysze wykresy pozwalaj nam z ĆatwoŻci zauway, e najszybciej funkcja celu docieraĆa do optimum w algorytmie z trybem 2. W przypadku funkcji nr 6 (Holder Table) widzimy ciekawe zjawisko, poniewa tylko algorytm z trybem nr 2 skutecznie odnajdywaĆ optimum. PozostaĆe tryby miaĆy problem by zawsze odnale minimum funkcji. Prawdopodobnie spowodowane jest to sposobem obsĆugi ogranicze jakim jest rzutowanie. Poniewa Holder Table to jedyna funkcja, ktra posiada swoje optimum w pobliu granicy swojej dziedziny, rzutowanie moe zaburza proces eksploatacji przestrzeni wokĆ minimum.

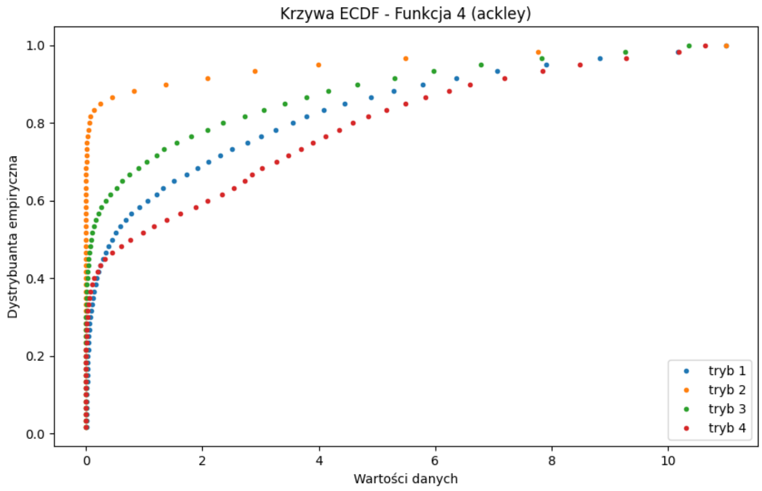
4.2. Krzywe ECDF

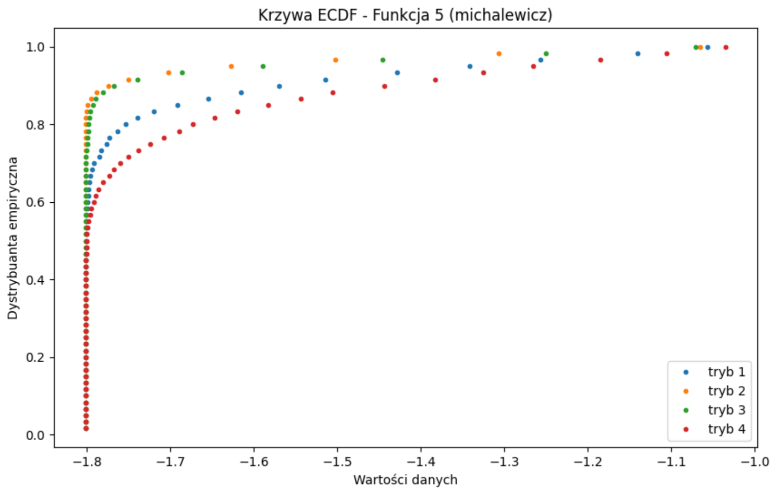
Krzywa empirycznej dystrybuanty skumulowanej (ECDF) to graficzna reprezentacja empirycznego rozkĆadu danych, przedstawiajca procentowy udziaĆ wartoŻci w prbce, ktre s mniejsze lub rwne danej wartoŻci.

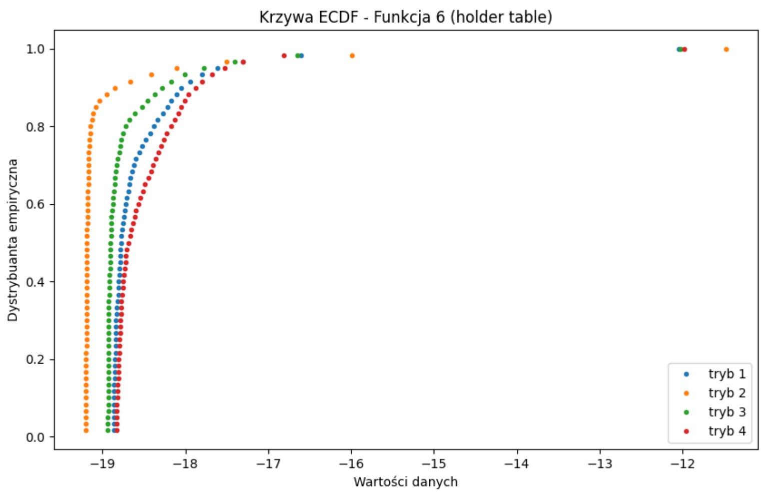








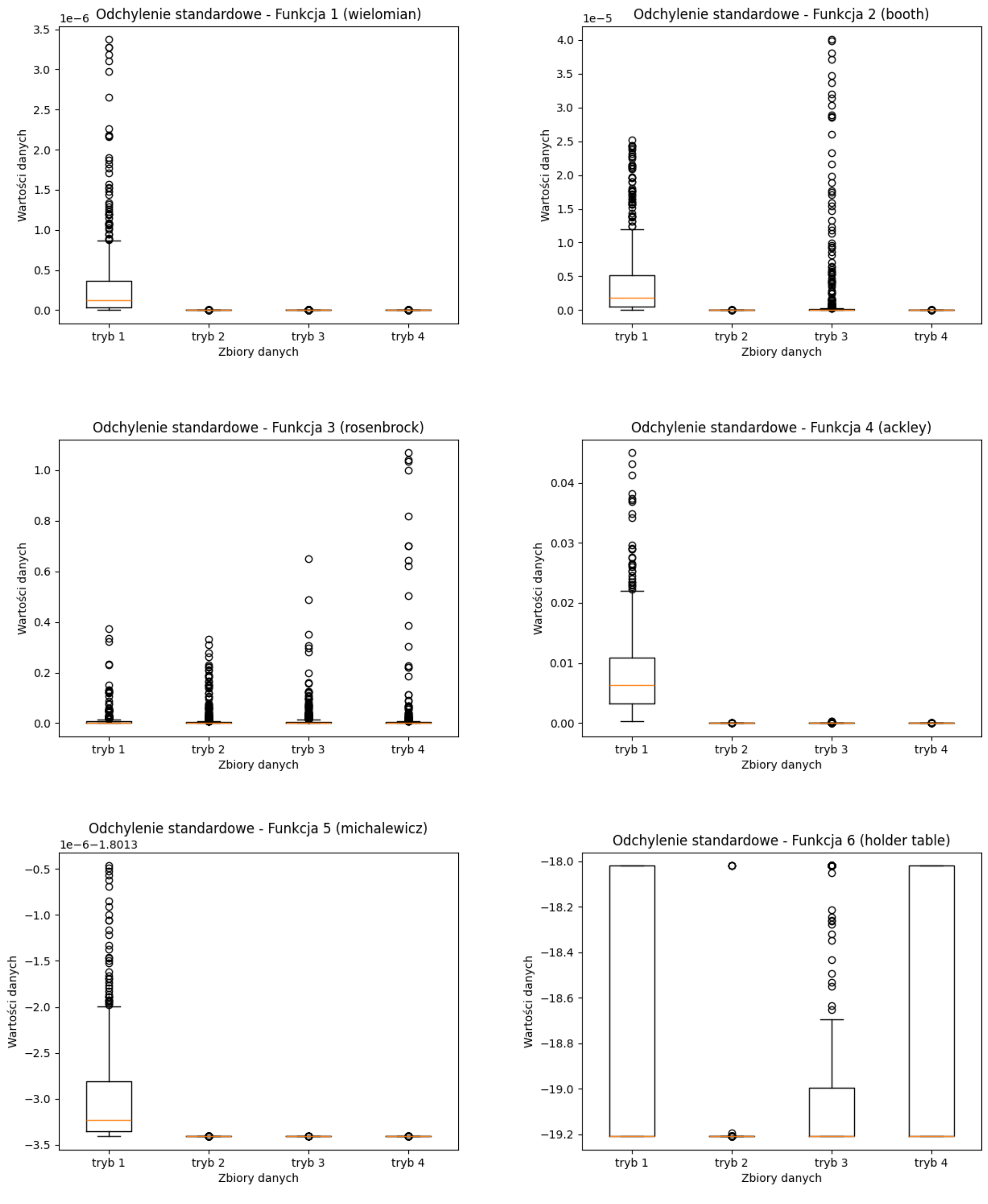




Rys. 4.3: Krzywe ECDF

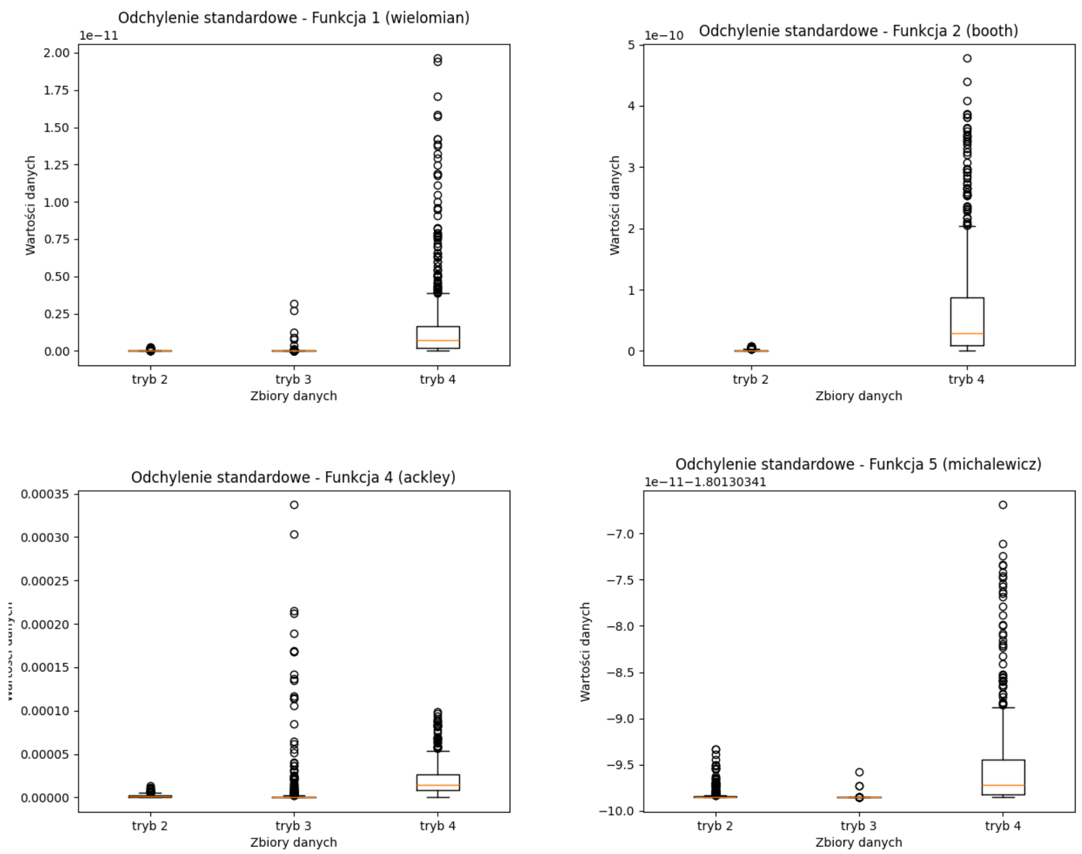
4.3. Wykresy pudeĆkowe

Wykresy pudeĆkowe graficznie przedstawiaj rozkĆad danych, prezentujc kwartyle, medianŹ i zakres interkwartylowy, co umoliwia analizŹ skupienia oraz rozproszenia wartoŻci w zbiorze.



Rys. 4.4: Wykresy pudeĆkowe

Wykresy pudeĆkowe dla funkcji, dla ktrych z powyszego wykresu nie da siŹ wybra najlepszego trybu.



Rys. 4.5: Wykresy pudeĆkowe

Na powyszych wykresach pudeĆkowych ponownie widzimy, e najlepiej radzi sobie algorytm w trybie 2, poniewa zwykle posiada najmniej wynikw odbiegajcych znaczco od optimum.

5. Wysze wymiary

Dla lepszej oceny jakoŻci modyfikacji algorytmu PSO przeprowadziliŻmy rwnie analizŹ dla funkcji o wyszych wymiarach. Przetestowane zostaĆo 5 oraz 10 wymiarowe funkcje: wielomianowa, Rosenbrocka, Ackley'a i Michalewicz'a.

5.1. 5 wymiarw

Przeprowadzono 300 testw z maksymaln liczb iteracji rwn 100.

Funkcja wielomianowa

Dziedzina przeszukiwa $x_i \in [-5, 5]$, dla i = 1, ę, 5.
Optimum = 0, dla punktu x = (0,...,0).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$5{,}46 \cdot 10^{-9}$	$\mathbf{4{,}57 \cdot 10^{-1}}$	$1{,}62 \cdot 10^{1}$	1,5651	1
$\mathbf{2{,}91 \cdot 10^{-17}}$	2,4692	$2{,}45 \cdot 10^{1}$	4,3228	2
$1{,}57 \cdot 10^{-8}$	$7{,}91 \cdot 10^{-1}$	$1{,}78 \cdot 10^{1}$	1,9681	3
$1{,}99 \cdot 10^{-16}$	$6{,}23 \cdot 10^{-1}$	$2{,}01 \cdot 10^{1}$	1,7849	4

Tab. 5.1: Porwnanie wynikw dla funkcji wielomianowej w 5 wymiarach w rnych trybach.

Funkcja Rosenbrock

Dziedzina przeszukiwa $x_i \in [-5, 10]$, dla i = 1, ę, 5.
Optimum = 0, dla punktu x = (1,...,1).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$\mathbf{1{,}57 \cdot 10^{-4}}$	$8{,}40 \cdot 10^{2}$	$3{,}87 \cdot 10^{4}$	$1{,}79 \cdot 10^{3}$	1
$9{,}78 \cdot 10^{-4}$	$3{,}46 \cdot 10^{3}$	$6{,}10 \cdot 10^{4}$	$6{,}18 \cdot 10^{3}$	2
$9{,}45 \cdot 10^{-2}$	$\mathbf{5{,}28 \cdot 10^{2}}$	$4{,}46 \cdot 10^{4}$	$1{,}25 \cdot 10^{3}$	3
$6{,}62 \cdot 10^{-2}$	$8{,}96 \cdot 10^{2}$	$5{,}63 \cdot 10^{4}$	$2{,}09 \cdot 10^{3}$	4

Tab. 5.2: Porwnanie wynikw dla funkcji Rosenbrock w 5 wymiarach w rnych trybach.

Funkcja Ackley'a

Dziedzina przeszukiwa $x_i \in [-32.768, 32.768]$, dla i = 1, ę, 5.
Optimum = 0, dla punktu x = (0,...,0).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$1{,}51 \cdot 10^{-3}$	1,6962	$1{,}98 \cdot 10^{1}$	4,8387	1
$\mathbf{7{,}31 \cdot 10^{-8}}$	3,4348	$2{,}00 \cdot 10^{1}$	5,9962	2
$9{,}48 \cdot 10^{-5}$	2,2039	$1{,}97 \cdot 10^{1}$	5,2112	3
$2{,}34 \cdot 10^{-7}$	**1,5420**	$2{,}00 \cdot 10^{1}$	4,5194	4

Tab. 5.3: Porwnanie wynikw dla funkcji Ackley w 5 wymiarach w rnych trybach.

Funkcja Michalewicz'a

Dziedzina przeszukiwa $x_i \in [0, \pi]$, dla i = 1, ę, 5.
Optimum = -4,687658

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$-4{,}687\,651$	**$-4{,}2958$**	$-2{,}8558$	$3{,}69 \cdot 10^{-1}$	1
$-4{,}687\,658$	$-4{,}2951$	$-2{,}9479$	$3{,}52 \cdot 10^{-1}$	2
$-4{,}687\,658$	$-4{,}2465$	$-2{,}5714$	$3{,}74 \cdot 10^{-1}$	3
$-4{,}687\,658$	$-4{,}1932$	$-2{,}6571$	$3{,}99 \cdot 10^{-1}$	4

Tab. 5.4: Porwnanie wynikw dla funkcji Michalewicz w 5 wymiarach w rnych trybach.

5.2. 10 wymiarw

Przeprowadzono 300 testw z maksymaln liczb iteracji rwn 200.

Funkcja wielomianowa

Dziedzina przeszukiwa $x_i \in [-5, 5]$, dla i = 1, ę, 5.
Optimum = 0, dla punktu x = (0,...,0).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$3{,}94 \cdot 10^{-10}$	$1{,}84 \cdot 10^{1}$	$6{,}55 \cdot 10^{1}$	$1{,}66 \cdot 10^{1}$	1
$7{,}97 \cdot 10^{-5}$	$2{,}63 \cdot 10^{1}$	$5{,}81 \cdot 10^{1}$	$1{,}73 \cdot 10^{1}$	2
$1{,}60 \cdot 10^{-3}$	**$1{,}57 \cdot 10^{1}$**	$5{,}75 \cdot 10^{1}$	$1{,}55 \cdot 10^{1}$	3
$3{,}25 \cdot 10^{-13}$	$1{,}75 \cdot 10^{1}$	$5{,}62 \cdot 10^{1}$	$1{,}55 \cdot 10^{1}$	4

Tab. 5.5: Porwnanie wynikw dla funkcji wielomianowej w 5 wymiarach w rnych trybach.

Funkcja Rosenbrock

Dziedzina przeszukiwa $x_i \in [-5, 10]$, dla i = 1, ę, 5.
Optimum = 0, dla punktu x = (1,...,1).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$3{,}56 \cdot 10^{-1}$	$6{,}17 \cdot 10^{4}$	$4{,}27 \cdot 10^{5}$	$6{,}99 \cdot 10^{4}$	1
$4{,}5778$	$8{,}78 \cdot 10^{4}$	$5{,}14 \cdot 10^{5}$	$7{,}80 \cdot 10^{4}$	2
$3{,}3961$	**$4{,}63 \cdot 10^{4}$**	$3{,}52 \cdot 10^{5}$	$5{,}52 \cdot 10^{4}$	3
$4{,}08 \cdot 10^{-1}$	$5{,}80 \cdot 10^{4}$	$5{,}46 \cdot 10^{5}$	$5{,}86 \cdot 10^{4}$	4

Tab. 5.6: Porwnanie wynikw dla funkcji Rosenbrock w 5 wymiarach w rnych trybach.

Funkcja Ackley'a

Dziedzina przeszukiwa $x_i \in [-32.768, 32.768]$, dla i = 1, ę, 5.
Optimum = 0, dla punktu x = (0,...,0).

Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$1{,}79 \cdot 10^{-4}$	**$5{,}9385$**	$2{,}06 \cdot 10^{1}$	$8{,}8042$	1
$1{,}6462$	$1{,}17 \cdot 10^{1}$	$2{,}08 \cdot 10^{1}$	$6{,}1319$	2
$1{,}09 \cdot 10^{-1}$	$9{,}4832$	$2{,}07 \cdot 10^{1}$	$7{,}9143$	3
$1{,}86 \cdot 10^{-6}$	$7{,}0258$	$2{,}05 \cdot 10^{1}$	$8{,}7486$	4

Tab. 5.7: Porwnanie wynikw dla funkcji Ackley w 5 wymiarach w rnych trybach.

Funkcja Michalewicz'a

Dziedzina przeszukiwa $x_i \in [0, \pi]$, dla i = 1, ę, 5.
Optimum = -9,66015.

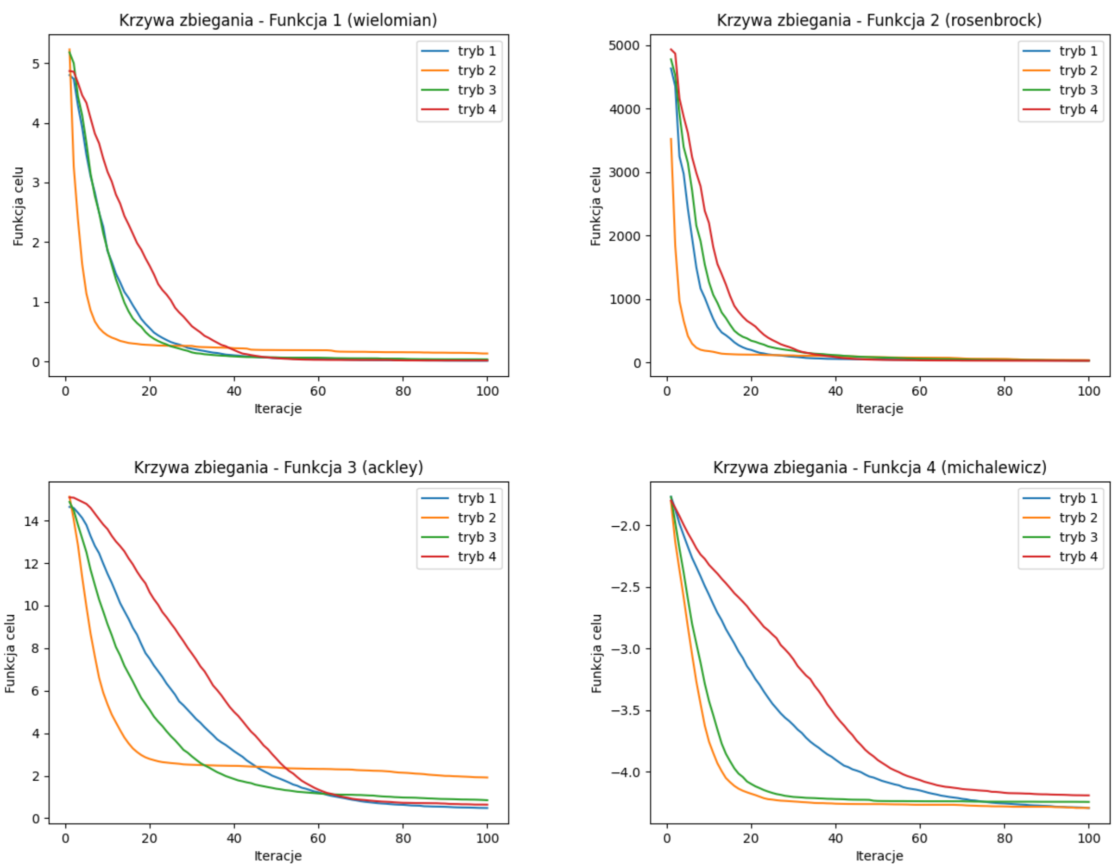
Najlepszy	redni	Najgorszy	Odchylenie standardowe	Tryb
$-9{,}386\,64$	$\mathbf{-7{,}5271}$	$-5{,}2036$	$7{,}69 \cdot 10^{-1}$	1
$-9{,}369\,41$	$-7{,}4807$	$-5{,}2040$	$7{,}82 \cdot 10^{-1}$	2
$\mathbf{-9{,}420\,70}$	$-7{,}3826$	$-5{,}2202$	$7{,}68 \cdot 10^{-1}$	3
$-9{,}269\,42$	$-7{,}0662$	$-4{,}8317$	$7{,}98 \cdot 10^{-1}$	4

Tab. 5.8: Porwnanie wynikw dla funkcji Michalewicz w 5 wymiarach w rnych trybach.

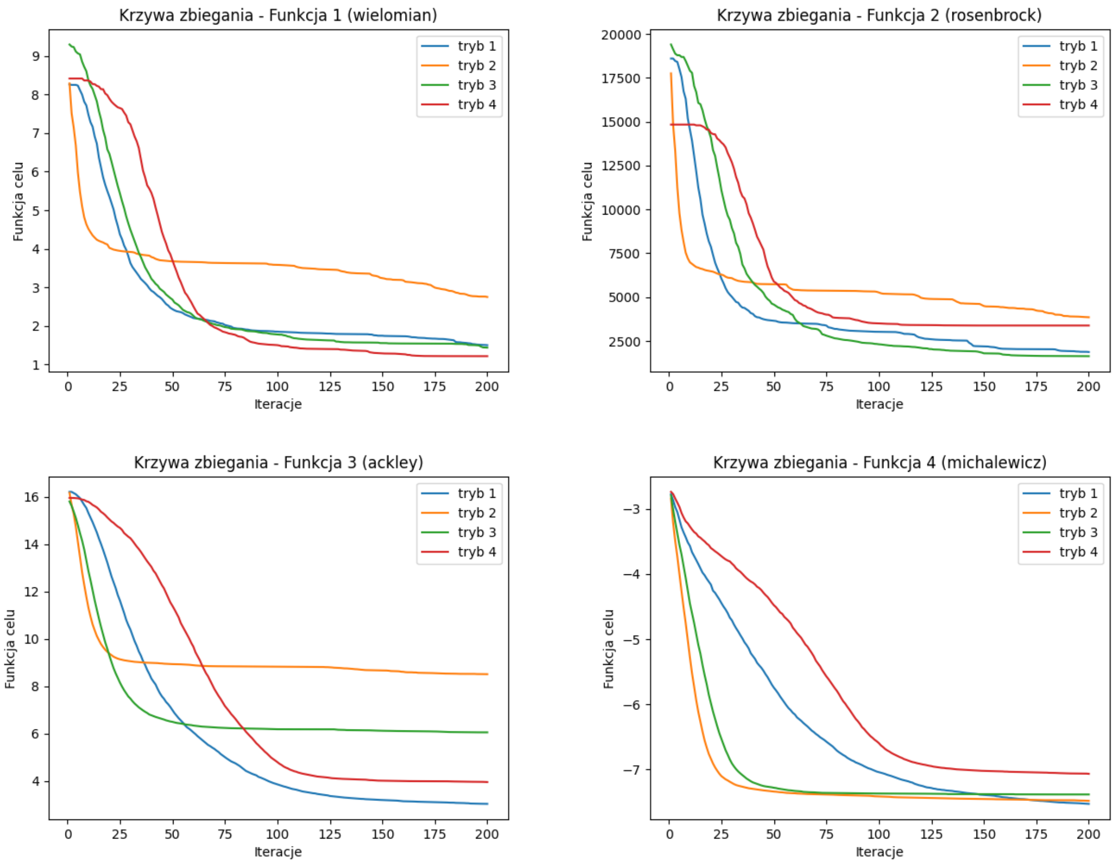
Widzimy, e dla wiŹkszej iloŻci wymiarw algorytm radzi sobie znacznie gorzej ni dla dwuwymiarowych funkcji. Nawet najlepsze wyniki dla rnych trybw odbiegaj potrafi znaczco od wartoŻci minimalnej funkcji. żadna z modyfikacji nie wysuwa siŹ na prowadzenie biorc pod uwagŹ dokĆadnoŻ wyniku optymalizacji oraz Żrednie wynikw.

5.3. Analiza wynikw

5.3.1. Krzywe zbienoŻci



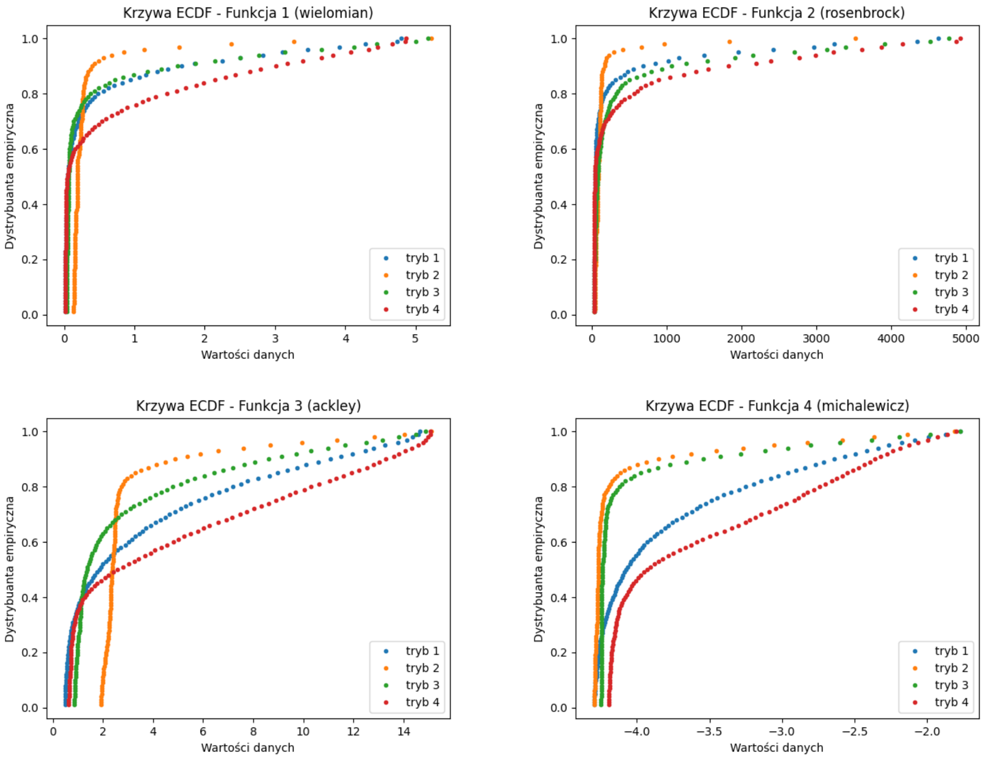
Rys. 5.1: Krzywe zbienoŻci - 5 wymiarowe



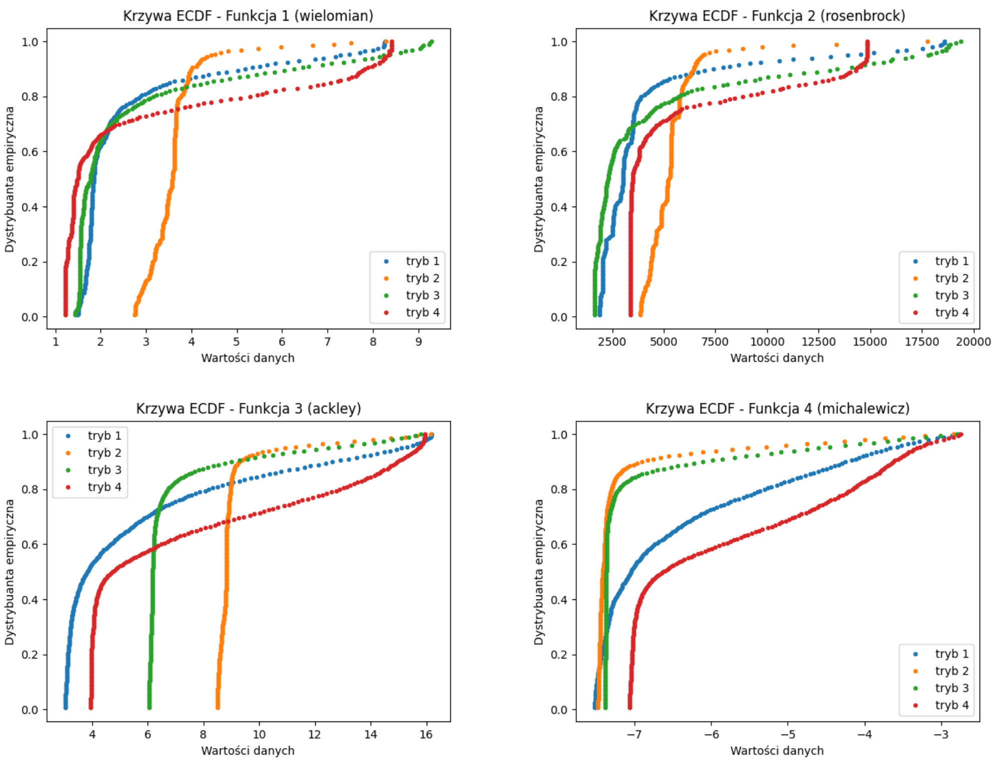
Rys. 5.2: Krzywe zbienoŻci - 10 wymiarowe

Powysze wyniki pokazuj, e dla wyszych wymiarw dobr modyfikacji nie jest ju tak oczywisty jak dla dwuwymiarowych funkcji. W wiŹkszoŻci przypadkw tryb nr 2 okazuje siŹ by pocztkowo najszybszym algorytmem, ale zwykle nie zbliajcym siŹ w dalszych iteracjach do optimum, zostajc daleko w tyle za pozostaĆymi modyfikacjami. Z tego powodu z powyszych krzywych zbienoŻci ciŹko jest wskaza najlepsz modyfikacjŹ.

5.3.2. Krzywe ECDF

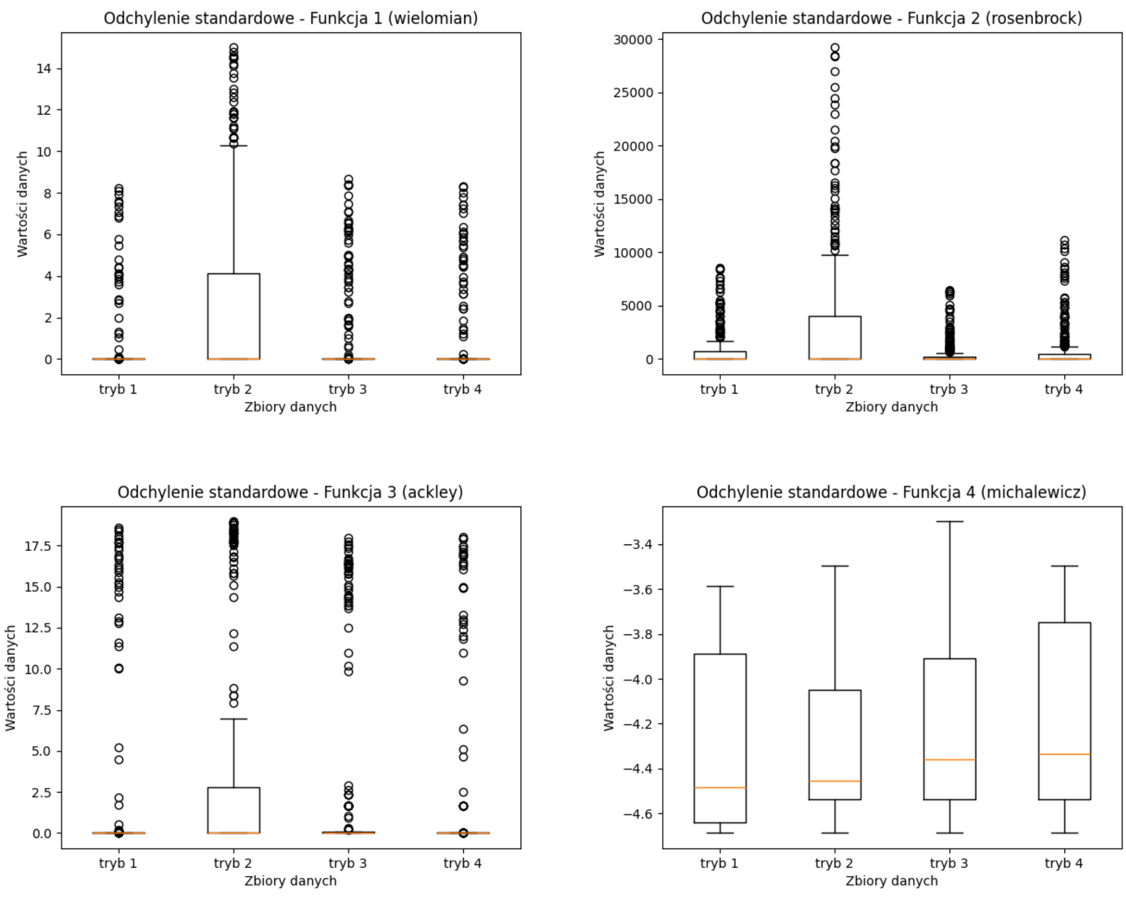


Rys. 5.3: Krzywe ECDF - 5 wymiarowe

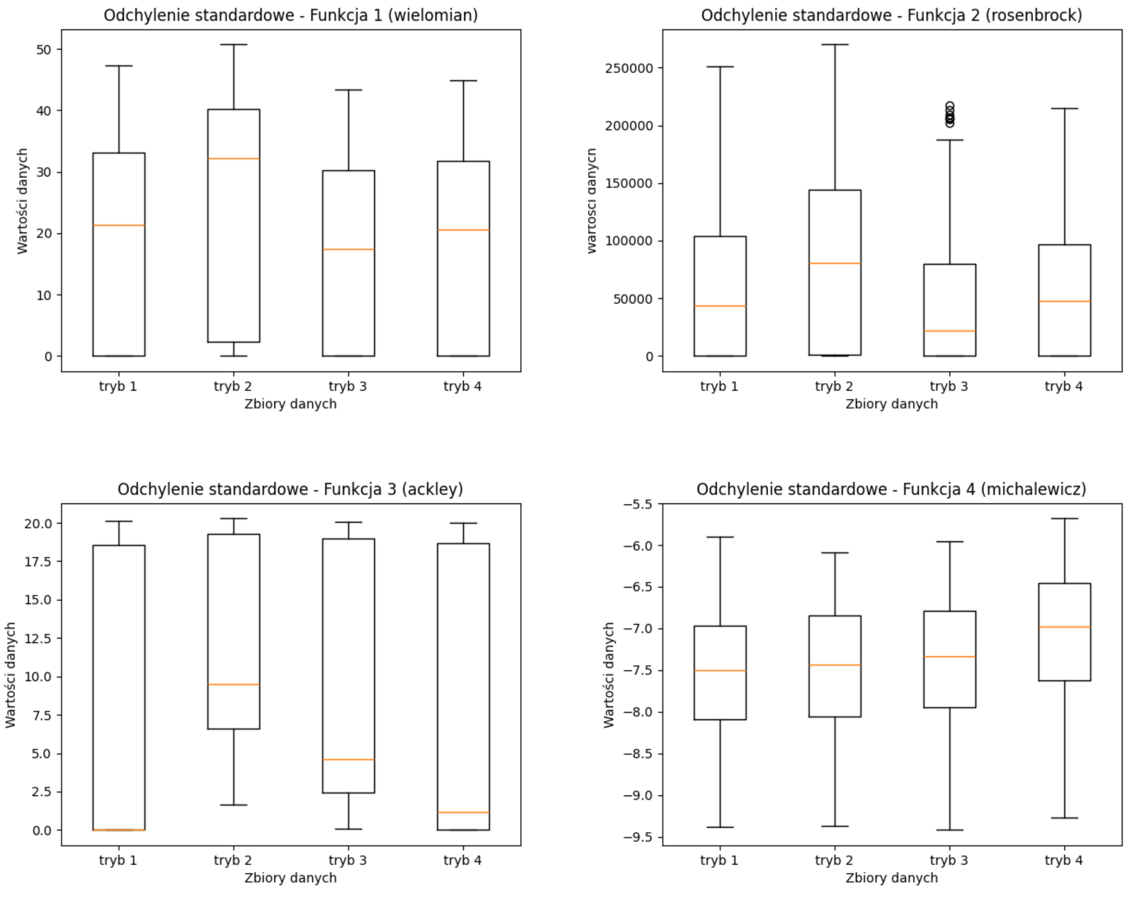


Rys. 5.4: Krzywe ECDF - 10 wymiarowe

5.3.3. Wykresy pudeĆkowe



Rys. 5.5: Wykresy pudeĆkowe - 5 wymiarowe



Rys. 5.6: Wykresy pudeĆkowe - 10 wymiarowe

Powyej widzimy, jak zwiŹkszenie wymiarw spowodowaĆo trudnoŻci ze znalezieniem optimum pomimo znaczego zwiŹkszenia czasu iteracji. Wykresy pudeĆkowe rwnie nie wskazuj, by ktraŻ z modyfikacji dawaĆa lepsze rezultaty ni pozostaĆe.

6. Podsumowanie

6.1. Porwnania tabelaryczne

Poniej oceniany jest kady tryb w rnych kategoriach w skali od 1 - najlepszy, do 4 - najgorszy

6.1.1. Najlepszy wynik

	2 wymiary						**5 wymiarw**				**10 wymiarw**				
	F1	**F2**	**F3**	**F4**	**F5**	**F6**	**F1**	**F2**	**F3**	**F4**	**F1**	**F2**	**F3**	**F4**	**Żrednia ranga**
tryb 1	3	4	4	4	2.5	2.5	3	1	4	4	2	1	2	2	2.786
tryb 2	2	2	2	2	2.5	2.5	1	2	1	2	3	4	4	3	2.357
tryb 3	1	1	3	1	2.5	2.5	4	4	3	2	4	3	3	1	2.5
tryb 4	4	3	1	3	2.5	2.5	2	3	2	2	1	2	1	4	2.357

6.1.2. Najlepsza Żrednia

	2 wymiary						**5 wymiarw**				**10 wymiarw**				
	F1	**F2**	**F3**	**F4**	**F5**	**F6**	**F1**	**F2**	**F3**	**F4**	**F1**	**F2**	**F3**	**F4**	**Żrednia ranga**
tryb 1	4	4	1	4	4	4	1	3	2	1	3	3	1	1	2,571
tryb 2	2	1	3	1	2	1	4	2	4	2	4	4	4	2	2,571
tryb 3	1	3	2	2	2	2	3	1	3	3	1	1	3	3	2,143
tryb 4	3	2	4	3	2	3	2	4	1	4	2	2	2	4	2,714

6.1.3. Krzywe zbienoŻci

	2 wymiary						**5 wymiarw**				**10 wymiarw**				
	F1	**F2**	**F3**	**F4**	**F5**	**F6**	**F1**	**F2**	**F3**	**F4**	**F1**	**F2**	**F3**	**F4**	**Żrednia ranga**
tryb 1	3	2.5	2	3	3	3	2.5	2	2.5	2.5	2.5	2.5	2.5	2	2.536
tryb 2	1	1	1	1	1.5	1	2.5	1	2.5	1	2.5	2.5	2.5	2	1.643
tryb 3	2	2.5	3	2	1.5	2	2.5	3	2.5	2.5	2.5	2.5	2.5	2	2.357
tryb 4	4	4	4	4	4	4	2.5	4	2.5	4	2.5	2.5	2.5	4	3.464

6.1.4. Wykresy pudeĆkowe

	2 wymiary						**5 wymiarw**				**10 wymiarw**				
	F1	**F2**	**F3**	**F4**	**F5**	**F6**	**F1**	**F2**	**F3**	**F4**	**F1**	**F2**	**F3**	**F4**	**Żrednia ranga**
tryb 1	4	4	2	4	4	3.5	2	3	1.5	1	3	2.5	1	1.5	2,643
tryb 2	1	1	1	1	2	1	4	4	4	2	4	4	4	1.5	2,464
tryb 3	2	3	3	2	1	2	2	1	3	4	1	1	3	3	2,214
tryb 4	3	2	4	3	3	3.5	2	2	1.5	3	2	2.5	2	4	2.679

6.1.5. Sumacyjne wyniki

modyfikacja	suma z Żrednich rang	ostateczna ranga
tryb 1	2.63	3
tryb 2	2.26	1
tryb 3	2.30	2
tryb 4	2.80	4

6.2. Podsumowanie

Wyniki przeprowadzonych przez nas testw wskazuj dwch faworytw. S to modyfikacja druga - wspĆczynnik zwiŹkszajcy swoj wartoŻ z czasem trwania programu oraz modyfikacja trzecia, czyli wspĆczynnik zaleny od jakoŻci rozwizania. Druga modyfikacja byĆa zdecydowanie najszybsza w wiŹkszoŻci testw, zazwyczaj najszybciej znajdowaĆa rozwizanie w ssiedztwie optimum, co szczeglnie wida na krzywych zbienoŻci. Modyfikacja trzecia za to najczŹŻciej zwracaĆa wynik w najmniejszym otoczeniu zaĆoonego optimum - rezultaty tej modyfikacji byĆy najdokĆadniejsze. DziaĆo siŹ to jednak kosztem liczby iteracji potrzebnych do znalezienia tych wartoŻci oraz potencjalnej wiŹkszej liczby przeprowadzonych podejŻ.

Analiza funkcji dwuwymiarowych niezaprzeczalnie wskazywaĆa na wyszoŻ modyfikacji numer 2 i 3 jednak dla wyszych wymiarw okazaĆo siŹ, e sposb modyfikacji a nawet jego brak nie wpĆywa jednoznacznie na jakoŻ wynikw optymalizacji. co prawda dalej algorytm z modyfikacj nr 2 najszybciej zmniejsza swoj funkcjŹ celu jednak do pewnego momentu oraz kosztem ostatecznego wyniku, ktry czŹsto znaczco odbiega od oczekiwanego optimum oraz wyniku innych, wolniejszych modyfikacji.

Dlatego w kwestii ostatecznego zwyciŹzcy warto rozway na czym nam zaley bardziej, ilu wymiarow funkcjŹ zamierzamy optymalizowa oraz czy mamy ograniczony czas wykonywania programu. Jeeli chodzi o szybkoŻ dziaĆania oraz przeciŹtnie otrzymany wynik to modyfikacja druga okae siŹ najskuteczniejsza. W przypadku, gdy moemy pozwoli sobie na wiŹcej testw, ktre potencjalnie mog zaj wiŹcej czasu, a zaley nam najbardziej na dokĆadnoŻci rozwizania, to modyfikacja trzecia okae siŹ odpowiedni opcj.

7. Bibliografia

1. https://en.wikipedia.org/wiki/Particle_swarm_optimization
2. https://www.sfu.ca/s̃surjano/
3. https://pure.port.ac.uk/ws/files/83267/geccoPP1.pdf
4. M. Clerc and J. Kennedy. The particle swarm - explosion, stability and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation, 6(1):5873, 2002