

# Sprawozdanie

## Algorytm roju cząstek

Amadeusz Filipek

Laboratorium komputerowe WFiS AGH

### 1. Wstęp

Celem ćwiczenia jest wykorzystanie algorytmu roju cząstek do rozwiązania problemu optymalizacyjnego.

Algorytm roju cząstek (Particle Swarm Optimization – PSO) jest jedną z najważniejszych heurystycznych metod optymalizacyjnych. PSO jest inspirowany przyrodą mianowicie opiera się o naśladowanie stadnych zachowań wśród populacji rojowych zwierząt takich jak owady czy ryby. Algorytm wykorzystuje populację (rój) cząstek. Położenie każdej cząstki reprezentuje rozwiązanie w przestrzeni argumentów optymalizowanej funkcji. Każda cząstka porusza się w przestrzeni z pewną prędkością. W każdej iteracji algorytmu cząstka przemieszcza się o wartość swojej prędkości. Prędkość cząstki w każdej iteracji jest korygowana na podstawie uzyskanych przez rój informacji o globalnym najlepszym przystosowaniu  $g_{best}$  oraz najlepszym przystosowaniu znalezionym przez cząstkę  $p_{best}$ . Populacja zatem posiada pamięć o najlepszych położeniach i informacje te wykorzystuje.

Istnieje wiele technik bazujących na algorytmie PSO, które stanowią rozszerzenie tej idei. Przykładem może być rodzina algorytmów zwana algorytmami roju cząstek o pełnej informacji (FIPSO – Fully Informed PSO). Algorytmy wprowadzają modyfikację do wymiany informacji pomiędzy osobnikami w stosunku do podstawowego PSO. O ile w PSO wykorzystywane są tylko wartości  $g_{best}$  oraz  $p_{best}$  tak w algorytmie FIPSO wprowadzają przestrzenną wymianę informacji pomiędzy wszystkimi lub częścią z osobników. Wtedy prędkości każdej cząstki korygowane są na podstawie średniej ważonej z przystosowania pozostałych osobników lub w innym podejściu z osobników w otoczeniu o pewnym promieniu.

### 2. Realizacja ćwiczenia

W ramach ćwiczenia zaimplementowano algorytm roju cząstek do znalezienia wartości minimalnej dwuargumentowej funkcji o wzorze:

$$f(x, y) = x^2 + y^2 - 20[\cos\pi x + \cos\pi y - 2]$$

Badany obszar argumentów ma zakres  $x, y \in [-10, 10]$ . Do rozwiązania problemu wykorzystano  $P = 40$  osobników populacji. Położenia oraz prędkości początkowe cząstek inicjowane są losowo w sposób jednorodny. Aktualizacja prędkości każdej cząstki w kolejnych iteracjach wyrażona jest wzorem:

$$v_x^{i+1} = \omega[v_x^i + U(0, c_1)(p_{best} - x_i) + U(0, c_2)(g_{best} - x_i)]$$

gdzie  $U(a, b)$  stanowi liczbę losową wygenerowaną z rozkładu jednostajnego o parametrach  $a$  i  $b$ ,  $c_1, c_2$  są współczynnikami akceleracji oraz  $\omega$  stanowi współczynnik ścisku, który ma za zadanie zapobiegać niekontrolowanemu wzrostowi prędkości. Aktualizacja składowej  $v_y$  przebiega w sposób analogiczny. Parametr  $\omega$  wyrażony jest poprzez współczynniki akceleracji w sposób następujący:

$$\omega = \frac{2}{2 + \sqrt{(c_1 + c_2)^2 - 4(c_1 + c_2)}}$$

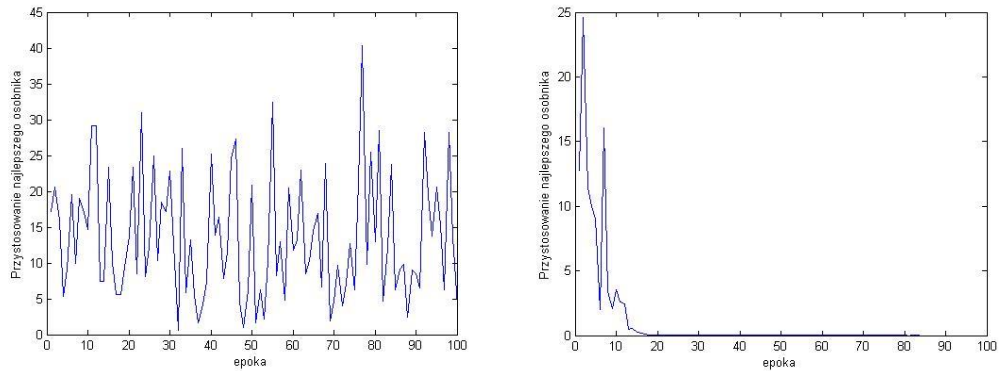
przy czym  $c_1 + c_2 > 4$ . Pierwsza wersja algorytmu nie wykorzystuje współczynnika ścisku, więc  $\omega = 1$  oraz współczynniki akceleracji nie muszą spełniać powyższej nierówności. Wówczas należy wprowadzić parametr prędkości maksymalnej, która ma za zadanie zapobiec nieskończonemu wzrostowi prędkości cząstek. Prędkość maksymalna oraz minimalna cząstek ustalone są jako  $v_{max} = 10, v_{min} = -10$ . W algorytmie uwzględniono sytuację wyjścia cząstki z analizowanego obszaru  $x, y \in [-10, 10]$ . Każda cząstka której położenie w następnym kroku iteracji wykraczałoby poza badanych obszar, uzyskuje to samo położenie (bez aktualizacji) natomiast jej składowa prędkości zmienia znak na przeciwny. Zbadano wyniki pracy tak zaimplementowanego algorytmu w zależności od wartości parametrów  $c_1$  i  $c_2$ . Jako warunek stopu algorytmu ustalono wartość 100 epok iteracji. Poniższa tabela przedstawia wartości średnie oraz odchylenia standardowe uzyskanych wartości najlepszego przystosowania przy  $N = 20$  testach dla każdej konfiguracji parametrów.

Tabela 1. Wartości średnie oraz odchylenia standardowe najlepszych przystosowań

		mean	std			mean	std
$c_1$	2	1.35	1.43	$c_1$	1	0.62	0.45
$c_2$	2			$c_2$	1		
$c_1$	1	1.01	0.84	$c_1$	2	0.89	0.79
$c_2$	2			$c_2$	1		
$c_1$	0	1.67	1.52	$c_1$	2	1.08	0.81
$c_2$	2			$c_2$	0		

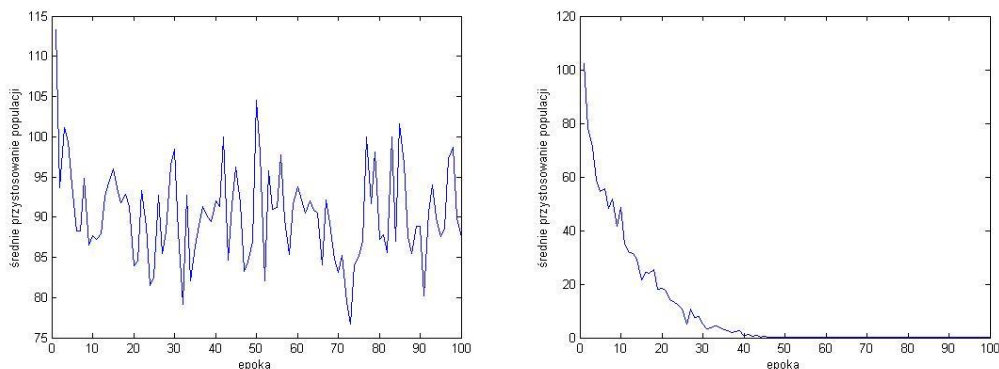
W tabeli kolorem szarym oznaczono konfigurację o najlepszych rezultatach. Najgorszy rezultat uzyskano gdy  $c_1 = 0$ , co oznacza brak pamięci każdego osobnika o jego własnym najlepszym uzyskanym położeniu. Najbardziej wyraźna jest różnica pomiędzy wynikami dla  $c_1 = c_2 = 2$  oraz  $c_1 = c_2 = 1$ . W obu przypadkach stosunki współczynników akceleracji są zachowane, jednakże mniejsze wartości wpływają na mniejszą zmianę prędkości cząstek co polepsza wyniki algorytmu. Wynik ten sugeruje, że przy mniejszych wartościach akceleracji cząstki poruszają się mniej chaotycznie i uzyskują lepszą zbieżność do minimum globalnego.

Następnie w algorytmie uwzględniono współczynnik ścisku  $\omega$  i ustalono  $c_1 = c_2 = 2,2$ . Zrezygowano także z warunku na prędkość maksymalną cząstek. Przy  $N = 20$  testach algorytmu uzyskano średnią  $\mu = 5 \cdot 10^{-18}$  oraz odchylenie standardowe o wartości  $\sigma = 9 \cdot 10^{-18}$ . Wprowadzenie współczynnika ścisku w sposób jednoznaczny poprawia pracę algorytmu. Dla obu wersji algorytmu zbadano przystosowanie najlepszego osobnika w kolejnych iteracjach.



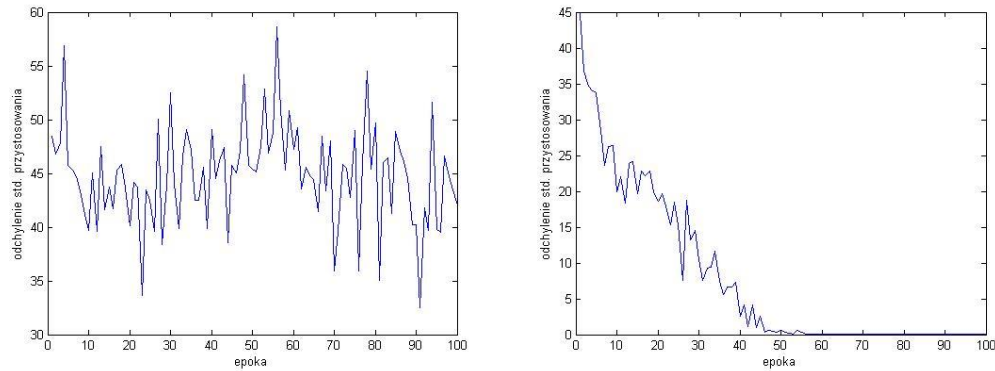
Wykres 1. Przystosowanie najlepszego osobnika w kolejnych iteracjach, z lewej bez współczynnika ścisku dla  $c_1 = c_2 = 1$ , z prawej przy uwzględnieniu współczynnika ścisku

Na wykresie lewym widać, że przystosowanie najlepszego osobnika w kolejnych iteracjach podlega silnym oscylacjom o charakterze losowym. Wprowadzenie współczynnika ścisku diametralnie zmienia charakter zbieżności przystosowania najlepszego osobnika, już po 20 iteracjach najlepszy osobnik osiąga minimum globalne. Następnie zbadano wartość średniego przystosowania populacji w kolejnych iteracjach działania algorytmu. Uzyskane wyniki przedstawione są na poniższych wykresach.



Wykres 2. Średnie przystosowanie populacji w kolejnych iteracjach, z lewej bez współczynnika ścisku dla  $c_1 = c_2 = 1$ , z prawej przy uwzględnieniu współczynnika ścisku

Średnie przystosowanie populacji na wykresie lewym w początkowym etapie algorytmu wykazuje tendencję spadkową jednakże w dalszym etapie obliczeń wchodzi w zakres silnych oscylacji, które utrzymują się do końca działania algorytmu. Po wprowadzeniu współczynnika ścisku średnie przystosowanie populacji wykazuje silny spadek o charakterze eksponencyjnym po czym osiąga zbieżność już przy 50 iteracjach. Zbadano także zależność odchylenia standardowego średniego przystosowania populacji w kolejnych iteracjach algorytmu i przedstawiono je na wykresach poniżej.



Wykres 3. Odchylenie standardowe średniego przystosowania w kolejnych iteracjach, z lewej bez współczynnika ścisku dla  $c_1 = c_2 = 1$ , z prawej przy uwzględnieniu współczynnika ścisku

Odchylenie standardowe średniego przystosowania populacji na wykresie lewym nie wykazuje trendu spadkowego, podlega natomiast silnym oscylacjom o charakterze losowym. Wprowadzenie współczynnika ścisku powoduje tłumienie oscylacji, w rezultacie odchylenie standardowe uzyskuje zbieżność po 60 iteracjach algorytmu. Na wykresie prawym wciąż obecne są oscylacje, jednakże nie zaburzają one zbieżności algorytmu.

### 3. Podsumowanie

W ramach ćwiczenia zaimplementowano algorytm roju cząstek i zastosowano go do optymalizacji dwuwymiarowej funkcji błędu. Zbadano dwie wersje algorytmu: pierwszą bez współczynnika ścisku (z prędkością maksymalną) oraz drugą uwzględniającą współczynnik ścisku. Dla wersji pierwszej zbadano wpływ parametrów akceleracji na wyniki działania algorytmu. Zaobserwowano, że wartości mniejsze ( $c_1 = c_2 = 1$ ) osłabiają chaotyczny ruch cząstek co prowadzi do lepszych rezultatów. Następnie porównano charakter zbieżności pierwszej wersji algorytmu z drugą. Zaobserwowano, że wprowadzenie współczynnika ścisku znacznie poprawia zbieżność algorytmu, tłumiąc statystyczne oscylacje i wynikający z nich chaotyczny charakter ruchu cząstek.