

Matematyka dla ciekawych świata.

Swarm optimization

Sławomir Walkowiak

16 maja 2014



- ▶ algorytm wyszukiwania lokalnego - przykład podstawowy
- ▶ algorytm genetyczny
- ▶ rój cząstek - inspiracja zachowaniem stada ptaków, opis algorytmu i implementacja dla wybranych problemów

Znajdowanie minimum funkcji

- ▶ funkcja/powierzchnia n-wymiarowa
- ▶ algorytm wyszukiwania lokalnego - przykład podstawowy
- ▶ symulowane wyżarzanie - różne schematy
- ▶ zawody dla dwuwymiarowej funkcji
- ▶ funkcje trudniejsze - Rosenbrock, Rastrigin, dziedzina $[-5.12, 5.12]$ dla każdego z wymiarów

Algorytm genetyczny - schemat ogólny

- ▶ piszemy wektorowo
- ▶ inicjalizacja populacji
- ▶ pętla aż do osiągnięcia optimum
 1. rekombinacja
 2. mutacje - 2 schematy
 3. selekcja - stała liczba osobników

Algorytm genetyczny - schemat ogólny

Mutacja pierwszego typu - izotropowa

- ▶ $\sigma' = \sigma \exp(\tau N(0, 1))$
- ▶ $x_{t+1} = x_t + N(0, \sigma')$
- ▶ $\tau = \frac{1}{\sqrt{n}}$
- ▶ więzy - $\sigma' < \epsilon_0 \implies \sigma' = \epsilon_0$

Algorytm genetyczny - schemat ogólny

Mutacja drugiego typu - anizotropowa

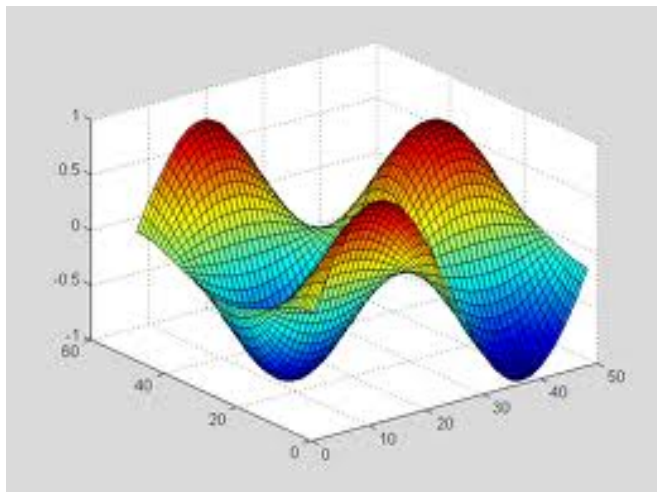
- ▶ $\sigma = \sigma_i \exp(\tau' N(0, 1) + \tau N_i(0, 1))$
- ▶ $x_{t+1} = x_t + N(0, \sigma')$
- ▶ $\tau' = \frac{1}{\sqrt{2n}}$
- ▶ $\tau = \frac{1}{\sqrt{2\sqrt{n}}}$
- ▶ więzy - $\sigma'_i < \epsilon_0 \implies \sigma'_i = \epsilon_0$

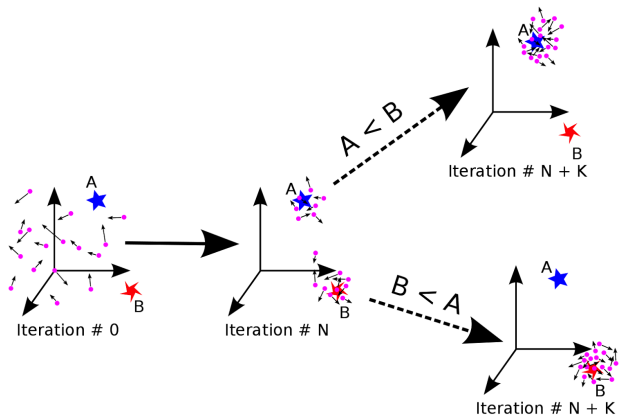
Swarm optimization - idea

- ▶ inicjalizacja populacji - położeń i prędkości, i historii położeń
- ▶ pętla aż do osiągnięcia optimum
 1. oblicz globalne optimum i lokalne optima
 2. oblicz prędkości
 3. ustal nowe położenia

Swarm optimization - algorytm, szczegóły

- ▶ ustalić prędkość maksymalną w każdym z wymiarów na np 5.12
- ▶ ustalić parametry c_1, c_2 - z reguły są równe 2 - odpowiadają sile przyciągania globalnego minimum i najlepszego jak dotąd znalezionego rozwiązania
- ▶ **pbest** - położenie z najlepszym jak dotąd wynikiem dla danej cząsteczki
- ▶ **gbest** - najlepsze położenie w danej iteracji
- ▶ liczenie prędkości -
$$\mathbf{v} = \mathbf{v} + c_1 \mathbf{N}(0, 1)(\mathbf{pbest} - \mathbf{x}) + c_2 \mathbf{N}(0, 1)(\mathbf{gbest} - \mathbf{x})$$
- ▶ aktualizacja położeń $\mathbf{x}_{t+1} = \mathbf{x} + \mathbf{v}$





podsumowanie

- ▶ samodzielna praca i poszukiwanie wiedzy
- ▶ dowolny język programowania
- ▶ podstawy optymalizacji

Particle Swarm
optimisation

