Metody Rozmyte i Algorytmy Ewolucyjne

mgr inż. Piotr Kaczyński

Wydział Matematyczno-Przyrodniczy
Szkoła Nauk Ścisłych
Uniwersytet Kardynała Stefana Wyszyńskiego

Algorytm symulowanego wyżarzania
Podstawowy schemat algorytmu ewolucyjnego

Plan prezentacji

Optymalizacja lokalna - przypomnienie

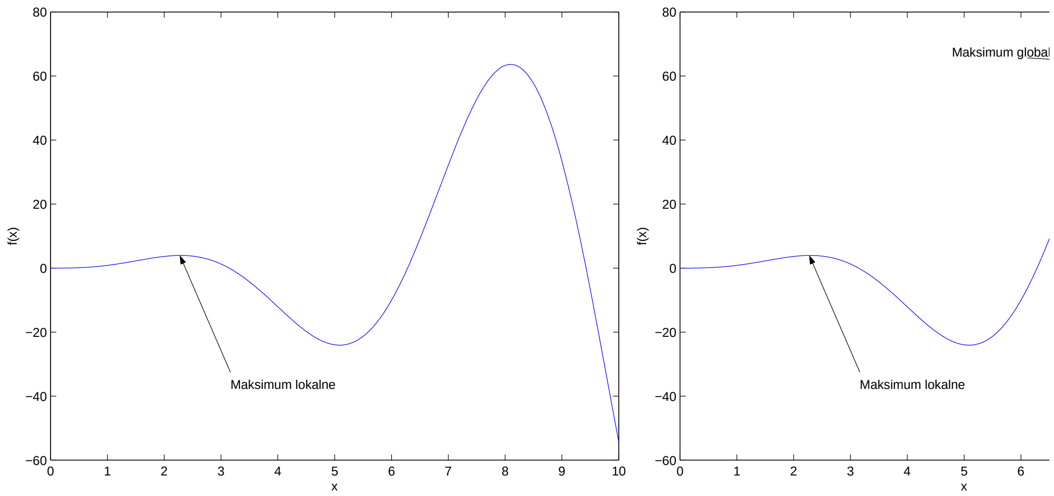
- Przedstawiane ostatnio algorytmy deterministyczne znajdują minimum lokalne,
- Najczęściej wykorzystywano gradient do określenia kierunku poprawy,
- Czy istnieją algorytmy dające rozwiązanie optymalne globalnie?
- Tak, ale kosztem tego, że są niedeterministyczne
- Takie algorytmy to m.in.
  - Symulowanego wyżarzania (jeden punkt),
  - Algorytmy ewolucyjne (wiele punktów),

Zmiana konwencji zapisu

- Dotychczas stosowana konwencja (standardowa dla teorii optymalizacji):
$$\min_x f(x)$$
- Konwencja stosowana w algorytmach genetycznych:
$$\max_x f(x)$$
- Konwencja ta będzie stosowana do końca wykładu,
- Przypomnienie: minimalizację zawsze można zamienić na maksymalizację
$$\min_x f(x) \equiv \max_x f(x)$$

Optymalizacja lokalna - przykład

Przykład



Optymalizacja lokalna - przykład II

- Algorytm startując z punktu w pobliżu maksimum lokalnego zbiegnie do tego właśnie punktu,
- Algorytm startując z punktu stacjonarnego nie wykona żadnej iteracji, bo
$$\nabla f = 0$$
- Algorytm optymalizacji lokalnej startując z punktu w pobliżu maksimum globalnego zbiegnie do tego maksimum,
- Algorytmy optymalizacji lokalnej najczęściej wykonują ruch zgodnie z gradientem funkcji celu (tylko polepszają),
- Co zrobić, aby algorytm startując z okolic optimum lokalnego zbiegł do optimum globalnego?
- Aby dotrzeć do optimum globalnego trzeba zdecydować się na ruch pogarszający aktualne rozwiązanie.

Ogólne własności algorytmu

- Algorytm symulowanego wyżarzania jest algorytmem iteracyjnym,
$$x_{k+1} = H(x_h)$$
gdzie H jest operatorem algorytmu,
- W algorytmie rozważany jest tylko jeden punkt w każdej iteracji,
- Wartość funkcji celu w punkcie wyznaczonym w kolejnej iteracji algorytmu może być większa, ale również może być mniejsza niż wartość funkcji celu w iteracji poprzedniej,
- Oczywiście rozwiązania polepszające powinny być w pewien sposób preferowane od pogarszających

Schemat algorytmu symulowanego wyżarzania

1. $k = 1$, $x_k = x^{start}$, $T_k = T^{start}$
2. `while not` warunek stopu `do`
3. $z = x_k + \xi_k$, gdzie ξ jest zmienną losową o zerowej wartości średniej
4. `if` $f(z) > f(x_k)$ `then`
   - Przyjmij $x_{k+1} = z$
5. `else`
   - Przyjmij $x_{k+1} = z$ z prawdopodobieństwem
   $$p_a = e^{-\frac{|\Delta f|}{T_k}}$$
   gdzie $|\Delta f| = |f(x_k) - f(z)|$
6. `endif`
7. Przyjmij $T_k = h(T_k)$, $k = k + 1$,

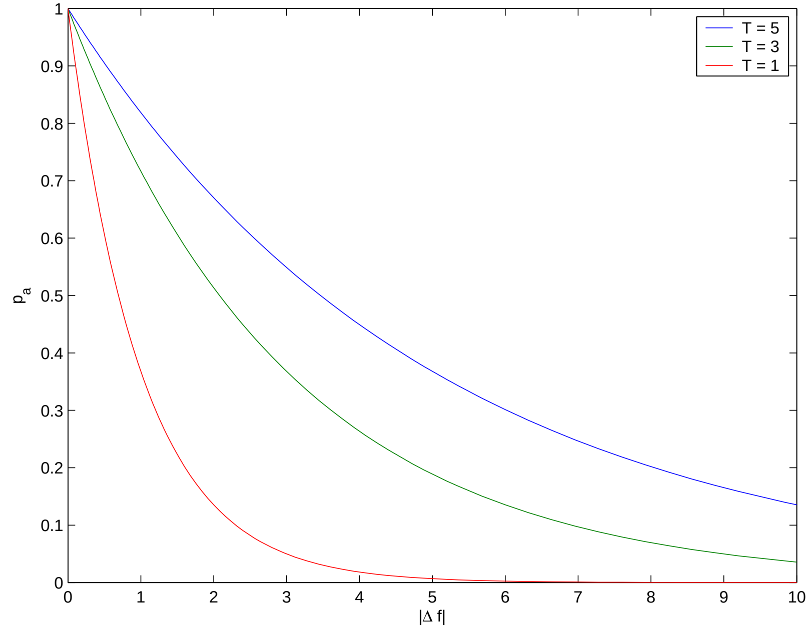
Algorytm symulowanego wyżarzania — obserwacje

- Parametr $T \geqslant 0$ nazywamy temperaturą,
- Dla szybkości działania algorytmu jest on bardzo istotnym parametrem,
- To, czy „gorszy” punkt zostanie wybrany jako kolejne rozwiązanie powinno być coraz mniej prawdopodobne

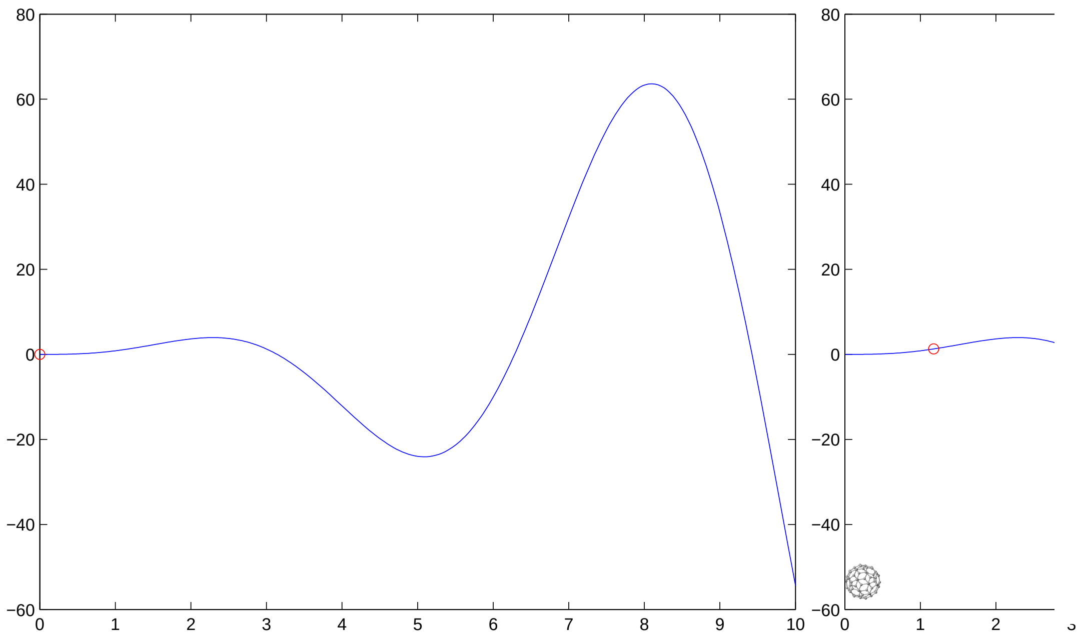
$$T_{k+1} < T_k$$

- Zbyt szybkie obniżanie osłabia dokładność algorytmu,
- Powolne obniżanie wydłuża czas obliczeń,
- Dodatkowo można zmieniać wariancję zmiennej ξ_k w kolejnych iteracjach (również zmniejszać)

Prawdopodobieństwo w zależności od T



Symulacja



Przykład — podsumowanie

- Algorytm nie w każdej iteracji wykonał ruch, prezentowane były tylko te iteracje, po których on nastąpił,
- Wykonano równo 100 iteracji, tylko w 15 zmienił się x_k,
- Przyjęto $T_{k+1} = 0.9T_k$, niezależnie od tego, czy wystąpiła zmiana, czy nie,
- Błądzenie losowe z rozkładem normalnym ξ o odchyleniu standardowym $\sigma = 2$,
- Algorytm znalazł optimum globalne pomimo startu z „niewygodnego” punktu.

Geneza

- Jak widać z algorytmu symulowanego wyżarzania – metody stochastyczne mogą dawać dobre rezultaty,
- Dodatkowo można rozważać więcej niż jedno rozwiązanie w każdej iteracji,
- Można uogólnić metodę „losowania” nowego punktu,
- Warto wykorzystać więcej niż jedno rozwiązanie do znalezienia kolejnych, lepszych punktów,
- Nadal utrzymujemy zasadę, że promujemy lepsze, ale „nie zapominamy” o gorszych

Oznaczenia i nazewnictwo

- Algorytm ewolucyjny przetwarza pewną populację osobników, z których każdy jest pewną propozycją rozwiązania problemu,
- Działa on w środowisku, które można zdefiniować na podstawie problemu,
- Każdemu osobnikowi można przyporządkować liczbę określającą jakość przechowywanego przez niego rozwiązania zwaną przystosowaniem,
- Środowisko można opisać funkcją przystosowania,
- Genotyp każdego osobnika składa się z chromosomów, które składają się z genów

Oznaczenia i nazewnictwo II

Przykład

Dla prostego zagadnienia optymalizacji

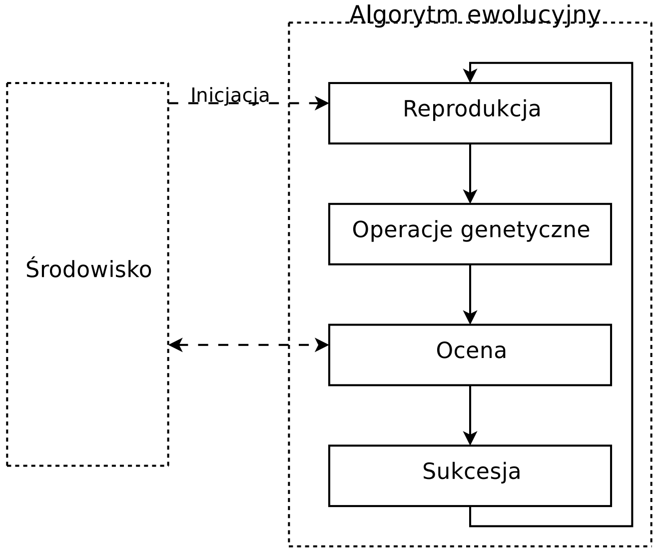
$$\max_{x \in \mathbb{R}^3} f(x)$$

osobniki mogą składać się z jednego chromosomu — wektora w $\mathbb{R}^3$, którego każdy z elementów jest pojedynczym genem. Środowisko opisuje funkcja przystosowania $f(x)$, która pozwala określić dopasowanie osobnika x.
Oczywiście im większa wartość przystosowania, tym lepszy osobnik.

Schemat algorytmu ewolucyjnego

- Działanie algorytmu polega na iteracyjnym wykonywaniu
  - Reprodukcji,
  - Operacji genetycznych,
  - Oceny,
  - Sukcesji,
- Algorytm jest zapoczątkowywany przez inicjację osobników,
- Reprodukcja i sukcesja nazywane są łącznie selekcją,

Schemat algorytmu ewolucyjnego II



Inicjacja i Ocena

- Inicjacja
  - Służy znalezieniu początkowych rozwiązań (osobników),
  - Tworzy początkową populację bazową,
  - Proces ten jest najczęściej losowy, niekiedy wykorzystuje się informacje o funkcji przystosowania,
  - Konieczna jest ocena nowoutworzonych osobników,
- Ocena
  - Proces oceny nowych osobników powstałych po zastosowaniu operatorów genetycznych,
  - Obliczenie funkcji przystosowania,

Sukcesja i reprodukcja

- Łącznie nazywane selekcją,
- Modelują proces umierania starych osobników i walki o przewodnictwo w stadach,
- Sukcesja
  - Wybór spośród osobników z poprzedniej epoki algorytmu,
  - Podobna do reprodukcji, lecz mająca inny charakter,
  - Promuje osobniki o lepszym przystosowaniu,
  - Może nie występować w algorytmie,
- Reprodukcja
  - Tworzenie prostych kopii osobników,
  - Promowane są osobniki o lepszej funkcji przystosowania,

Operatory genetyczne

- Razem z reprodukcją modelują proces rozmnażania, podczas którego materiał genetyczny przodków przekazywany jest potomkom
- Krzyżowanie
  - Operator który dwóm lub więcej osobnikom przyporządkowuje jeden lub więcej nowych osobników,
  - Chromosomy osobników potomnych powstają w wyniku pewnego „wymieszania” chromosomów osobników rodzicielskich,
- Mutacja
  - Losowe zaburzenie chromosomu osobnika,
  - Niewielkie perturbacje powinny być bardziej prawdopodobne niż duże.

Schemat algorytmu

1. $k = 0$
2. inicjacja P^0
3. ocena P^0
4. `while not` warunek stopu `do`
   - T^k = reprodukcja P^k
   - O^k = krzyżowanie i mutacja T^k
   - ocena O^k
   - P^{k+1} = sukcesja O^k
   - $k = k + 1$

Opis zadania

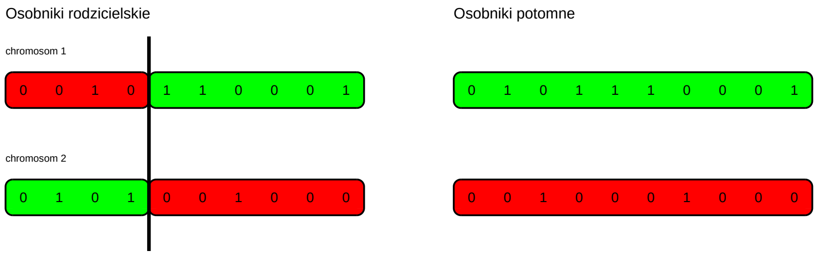
- Zadanie polega na znalezieniu wektora binarnego o jak największej ilości jedynek w przestrzeni 10 wymiarowej,

$$\max f(x) = \sum_{i=1}^{10} x_i, \quad x \in \{0,1\}^{10}$$

- Zadanie jest więc zadaniem programowania binarnego,
- Rozwiązanie jest oczywiste, klasyczne algorytmy mogą mieć problemy
- Przestrzeń rozwiązań do przeszukania: $2^{10} = 1024$ elementów
- Osobniki składają się z jednego chromosomu o 10 genach,

Zastosowane operatory genetyczne

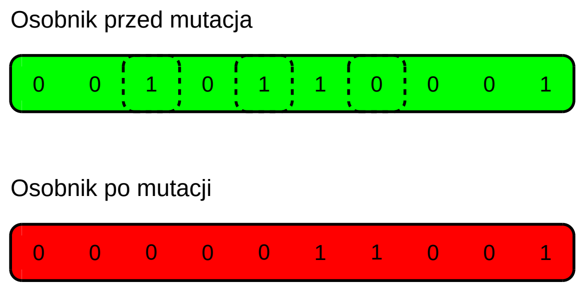
Przykład — krzyżowanie



- Wybór miejsca cięcia z rozkładem jednostajnym
- Prawdopodobieństwo krzyżowania p_c

Zastosowane operatory genetyczne II

Przykład — mutacja



- Ziana bitu na przeciwny,
- Prawdopodobieństwo zajścia zmiany p_m

Zastosowana reprodukcja

- Reprodukcja ruletkowa,
- Prawdopodobieństwo reprodukcji proporcjonalne do przystosowania osobnika,
- Jeśli $f(x)$ to przystosowanie osobnika x, to

$$p_r(x) = \frac{f(x)}{\sum_{y \in P^k} f(y)}$$

  oznacza prawdopodobieństwo reprodukcji zadanego osobnika x

Przykład

Na tablicy!