

1 Algorytm genetyczny

Start

Wczytaj parametry *population_size*, *iteration_count*, *mutation_probability* oraz *crossover_probability*

Wczytaj funkcję celu do minimalizacji *cost_function* oraz badany obszar *domain*

population \leftarrow pusta lista

Dla $i \leftarrow 0$ **do** *population_size* **wykonuj**

chromosome \leftarrow losowy punkt w badanym obszarze

 Dodaj *chromosome* do listy *population*

koniec

costs \leftarrow pusta lista

offspring \leftarrow pusta lista

best_chromosome \leftarrow *population*[0]

lowest_cost $\leftarrow \infty$

Dla $i \leftarrow 0$ **do** *iteration_count* **wykonuj**

 Wyczyść zawartość listy *costs*

 Wyczyść zawartość listy *offspring*

Dla $p \leftarrow 0$ **do** długość *population* **wykonuj**

chromosome \leftarrow *population*[*p*]

cost \leftarrow *cost_function*(*chromosome*)

Jeżeli *cost* < *lowest_cost* **to**

best_chromosome \leftarrow *chromosome*

lowest_cost \leftarrow *cost*

koniec

koniec

Gdy długość *offspring* < *population_size* **wykonuj**

a, b \leftarrow punkty z *population* wybrane sposobem turniejowym

Jeżeli losowa liczba z przedziału $[0, 1) \leq \textit{crossover_probability}$ **to**

 Przeprowadź krzyżowanie punktów *a* z *b* zgodnie ze wzorami (1) i (2)

koniec

Jeżeli losowa liczba z przedziału $[0, 1) \leq \textit{mutation_probability}$ **to**

 Przeprowadź mutację punktów *a* oraz *b* zgodnie ze wzorem (3)

koniec

 Ogranicz współrzędne *a* oraz *b* do badanego obszaru

 Dodaj *a* i *b* do listy *offspring*

koniec

population \leftarrow *offspring*

koniec

Wypisz *best_chromosome* oraz *lowest_cost*

Stop

2 Wzory

2.1 Krzyżowanie arytmetyczne (arithmetic crossover)

$$\bar{a}_i = a_i + \Phi * (b_i - a_i) \quad (1)$$

$$\bar{b}_i = b_i + \Phi * (a_i - b_i) \quad (2)$$

Gdzie Φ jest losową liczbą z przedziału $[0, 1]$.

2.2 Mutacja arytmetyczna (arithmetic mutation)

$$\bar{x}_i = x_i + \Phi_i \quad (3)$$

Gdzie Φ_i jest losową liczbą z przedziału $[-0.5, 0.5]$ dla każdego i .