

Differential evolution

Algorytm

Opis parametrów

- Stała mutacji F
- Crossover constant C_r
- Liczebność populacji NP
- Ilość wymiarów naszej funkcji (odpowiednik ilości cech) D
- Ilość generacji GEN
- Granice działania L, H

Strategie ewolucji:

- $\omega = x1 + F \cdot (x2 - x3)$
- $\omega = x5 + F \cdot (x1 + x2 - x3 - x4)$
- $\omega = xbest + F \cdot (x1 - x2)$
- $\omega = xbest + F \cdot (x1 + x2 - x3 - x4)$
- $\omega = xind + \lambda \cdot (xbest - x1) + F \cdot (x2 - x3)$

Differential evolution

1. Deklaracja początkowej populacji [Pop]

- Generujemy dla populacji zestaw potencjalnych rozwiązań
- Dla każdego elementu losujemy zestaw cech.
- Wyznaczamy dla nich wartość naszej funkcji, wartość tą oznaczamy jako Fit (fitness)
- Oznaczamy najlepszego osobnika z populacji

2. Ewolucja przebiega określoną ilość iteracji(**GEN**). W każdej iteracji przeprowadzamy następujące kroki dla każdego osobnika:

- Losujemy 3 innych osobników z populacji
- Losujemy parametr mutacji (co najmniej jedna cecha musi ulec zmianie)
- Aktualizujemy cechę osobnika na podstawie wylosowanych 3 osobników(**F**)
- Sprawdzamy czy cechy nie wyszły za zakres(**L,H**), jeżeli tak losujemy mu nową cechę
- sprawdzamy jaki wynik uzyskuje nowy osobnik (**Fit**)
- Jeżeli Fit nowego osobnika jest lepszy od rodzica wchodzi on do populacji na jego miejsce.
- Jeżeli Fit jest lepszy od dotychczasowego, najlepszego wchodzi on na jego miejsce

Differential evolution

Tworzenie tymczasowego osobnika potomnego:

1. Losuj cechę która ma ulec zmianie.
2. Dla każdej cechy sprawdź czy jest ona wylosowana z pkt 1 lub czy może ulec ewolucji wykorzystując C_r
 - Użyj wybranej strategii by zmodyfikować cechę lub przepisuj ją od rodzica

```
Rnd = (int)(rand()*D) ;
```

```
for (i=0; i<D; i++) {
```

```
    if ( (rand()<CR) || (Rnd == i) )
```

```
        X[i] = Pop[i][r[2]] + F * (Pop[i][r[0]] - Pop[i][r[1]]) ;
```

```
    else
```

```
        X[i] = Pop[i][j] ;
```

```
}
```

Differential evolution

Wymagane:

D – ilość wymiarów

NP,F,Cr – parametry kontrolne

GEN – ilość interacji

L, H – przedział w którym szukamy najlepszego rozwiązania

Inicjalizacja populacji $Pop_j \leftarrow \text{randij}[L,H]$ and wyznaczenie $Fit_j \leftarrow f(Pop_j)$

for g = 1 **to** GEN **do**

for j = 1 **to** NP **do**

 Wybierz losowe $r_1, r_2, r_3 \in [1, \dots, NP], r_1 \neq r_2 \neq r_3 \neq j$

 Stwórz tymczasowego osobnika $X \leftarrow S(r, F, Cr, Pop)$

 Sprawdź czy mieści się w przedziale ($x_i \in [L, H]$) $x_i \leftarrow \text{randi}[L, H]$

 Wyierz lepsze rozwiązanie (X, Pop_j), zaktualizuj najlepszego osobnika

end for

end for