

- d - długość chromosomu
- N - wielkość populacji
- θ_1 - parametr algorytmu, współczynnik uczenia
- θ_2 - parametr algorytmu, prawdopodobieństwo mutacji
- θ_3 - parametr algorytmu, współczynnik zaburzenia podczas mutacji

POPULATION-BASED-INCREMENTAL-LEARNING($F, N, \theta_1, \theta_2, \theta_3$)

```
1 p ← INITIAL-PROBABILITY-VECTOR();
2 P ← RANDOM-POPULATION(p,  $N$ );
3 POPULATION-EVALUATION(P,  $F$ );
4 while not TERMINATION-CONDITION(P)
5   do
6     x $i$  ← BEST-INDIVIDUAL(P);
7     for  $k$  ← 1 to  $d$ 
8       do
9          $p_k$  ←  $p_k \cdot (1 - \theta_1) + x_{ik} \cdot \theta_1$ ;
10    for  $k$  ← 1 to  $d$ 
11      do
12        if UNIFORM-RANDOM(0, 1) <  $\theta_2$ 
13          then  $p_k$  ←  $p_k \cdot (1 - \theta_3) + \text{BINARY-RANDOM}(0.5) \cdot \theta_3$ ;
14    P ← RANDOM-POPULATION(p,  $N$ );
15    POPULATION-EVALUATION(P,  $F$ );
```

BINARY-RANDOM(p)

```
1 if UNIFORM-RANDOM(0, 1) <  $p$ 
2   then  $z = 1$ ;
3   else  $z = 0$ ;
4 return  $z$ 
```

INITIAL-PROBABILITY-VECTOR()

```
1 p =  $\{p_1, p_2, \dots, p_d\}$ 
2 for  $k$  ← 1 to  $d$ 
3   do
4      $p_k$  ← 0.5;
5 return p
```

RANDOM-INDIVIDUAL(**p**)

```
1 x =  $\{x_1, x_2, \dots, x_d\}$ 
2 for  $k$  ← 1 to  $d$ 
3   do
4      $x_k$  ← BINARY-RANDOM( $p_k$ );
5 return x
```

RANDOM-POPULATION(**p**, N)

```
1 P =  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ 
2 for  $k$  ← 1 to  $N$ 
3   do
4      $\mathbf{x}_k$  ← RANDOM-INDIVIDUAL(p);
5 return P
```