

m05

# Dynamic Niche Radiusに基づく 個体間距離を考慮したBat Algorithm

○岩瀬拓哉 高野諒 上野史 高玉圭樹 (電気通信大学)

# Dynamic Niche Radiusに基づく個体間距離を考慮した Bat Algorithm

○岩瀬拓哉 高野諒 上野史 高玉圭樹（電気通信大学）

Bat Algorithm [Yang. X.S, 2010] ... コウモリの発するラウドネス $A$ とその反射波 $r$ により大域探索と局所探索を自動で切り替えることが可能

## STEP1: 最良個体方向へ探索①

$$v_i^{t+1} = v_i^t + (x_*^t - x_i^t) * rand$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

## STEP2: グローバルベスト近辺を局所探索②

if  $rand > r_i$

$$x_{loc} = x_* + A_i^t * rand$$

endif

## STEP3: ランダムによる大域探索③

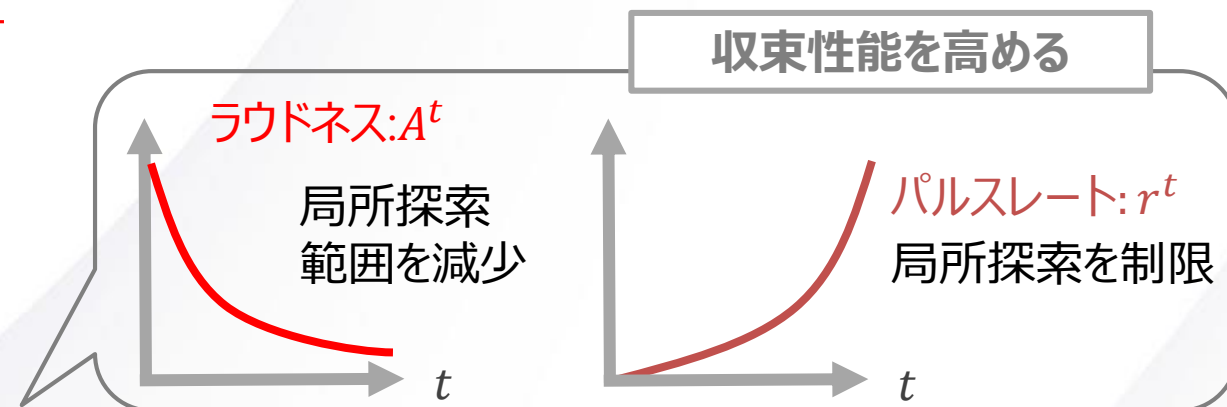
$$x_{rnd} = x_{lb} + (x_{ub} - x_{lb}) * rand$$

## STEP4: ①, ②, ③の解候補と現在の解の評価

$x_{i*}$ を更新

$$A_i^{t+1} = \alpha A_i^t$$

$$r_i^{t+1} = r_i^t [1 - \exp(-\gamma t)]$$



# Dynamic Niche Radiusに基づく個体間距離を考慮した Bat Algorithm

○岩瀬拓哉 高野諒 上野史 高玉圭樹 (電気通信大学)

## Dynamic Niche Radius [Miller,1996]

$$\lambda = \frac{1}{2} \sqrt{(x_{ub} - x_{lb})^2}$$

探索範囲の上限と下限:  $x_{ub}, x_{lb}$

$$\text{Niche Radius : } \sigma = \frac{\lambda}{\sqrt[D]{N}}$$

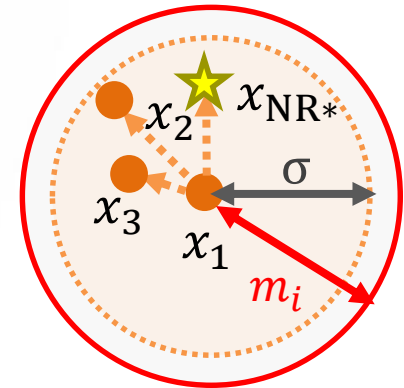
次元数:  $D$  個体数:  $N$

$$\text{Sharing function : } sh(d_{ij}) = \begin{cases} 1 - (\frac{d_{ij}}{\sigma}) & (\text{if } d_{ij} < \sigma) \\ 0 & (\text{otherwise}) \end{cases}$$

$$\text{Niche count : } m_i = \sum_{j=1}^N sh(d_{ij})$$

$$\text{Dynamic Niche Radius : } m_i^{dyn} = \begin{cases} \sigma & (\text{if } m_i < \sigma) \\ m_i & (\text{otherwise}) \end{cases}$$

値が大きいほど周辺に  
個体が密集



★ :  $x_{NR*}$

● :  $x_i$  ( $i = 1, 2, \dots, N$ )

# Dynamic Niche Radiusに基づく個体間距離を考慮した Bat Algorithm

○岩瀬拓哉 高野諒 上野史 高玉圭樹 (電気通信大学)

## Dynamic Niche Radius-based Bat Algorithm (DNRBA)

### STEP1: Dynamic Niche Radiusの算出

### STEP2: 最良個体から離れる方向へ探索①

$$v_i^{t+1} = v_i^t + (x_i^t - x_{NR*}) * rand$$

$$x_i^{t+1} = \begin{cases} x_i^t + v_i^{t+1} & (\text{if } d_i < m_i^{dyn}) \\ x_i^t & (\text{otherwise}) \end{cases} \quad d_i: \text{個体間距離}$$

### STEP3: Niche Radius内で局所探索②

if  $rand > r_i$

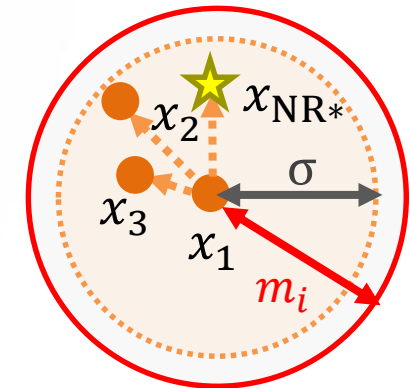
$$x_{loc} = x_{NR*} + A_i^t * rand$$

endif

### STEP4: Niche Radius内で大域探索③

$$x_{rnd} = x_{NR*} + rand(1, D, [-m, m])$$

### STEP5: ①, ②, ③の解候補と現在の解の評価



★ :  $x_{NR*}$

● :  $x_i$  ( $i = 1, 2, \dots, N$ )