0.0.1 Re-framed Bat Inspired Algorithm (Re-framed BA)

- **Objectives**:

Objective Problem	
$f(\mathbf{x}_i)$	fitness of $\mathbf{x}_i$.
n	the dimension of the search space.
$[lb_{\mathbf{x}}, ub_{\mathbf{x}}]$	the interval of objective variable $\mathbf{x}$, in our cases, it is defined in the IOHprofiler, $[lb_{\mathbf{x}}, ub_{\mathbf{x}}] = [-5, +5]$.
Objective Solution	
$\mathbf{x}_i$	it can be imagined as one individual in Swarm-Intelligence Algorithms, $\mathbf{x}_i \in R^n$.

- **Parameters**:

T	maximum iteration, the budget in our cases, in our case, it is defined in IOHprofiler.
M	population size, $M = 20, M \in [20, 40]$.
$\mathbf{x}_g$	the best position that the whole population has found so far.
$\mathbf{y}_i$	the velocity for one individual $\mathbf{y}_i$.
z_1	a kind of probability, in this case, it is called the decreasing pulse rate.
z_2	the decreasing loudness.
z_1^0	the initial value of pulse rate z_1, $z_1^0 = 1$, $z_1^0 \in [0, 1]$.
z_2^0	the initial value of loudness z_2, $z_2^0 = 1$, $z_2^0 \in (0, +\infty)$.
w_1	used to decrease pulse rate z_1 , $w_1 = 0.1$, $w_1 \in [-1, 1]$.
w_2	used to decrease loudness z_2, $w_2 = 0.97$, $w_2 \in [-1, 1]$.
w_3	used to update local position, $w_3 = 0.1$, $w_3 \in [-1, 1]$.

$[lb_{w_4}, ub_{w_4}]$	the interval of frequency w_4, $[lb_{w_4}, ub_{w_4}] = [0,2]$. $[lb_{w_4}, ub_{w_4}] \subset [0,+\infty]$.

- **Components**:
  - Initialization Process:
    (1) Initialize $\mathbf{x}_i(t=0)$:

$$\mathbf{x}_i(t=0) = \boldsymbol{\mathcal{U}}(lb_{\mathbf{x}}, ub_{\mathbf{x}}), i = 1 \dots M \quad 1$$

    (2) Initialize $\mathbf{y}_i(t=0)$:

$$\mathbf{y}_i(t=0) = \boldsymbol{\mathcal{U}}(0,0), i = 1 \dots M \quad 2$$

    (3) Initialize $\mathbf{x}_g(t=0)$:

$$\mathbf{x}_g(t=0) = \mathbf{Min}(\{\mathbf{x}_i(t)\}), i = 1 \dots M \quad 3$$

    (4) Initialize $z_1(t=0)$:

$$z_1(t=0) = z_1^0 \times w_2 \quad 4$$

    (5) Initialize $z_2(t=0)$:

$$z_2(t=0) = z_2^0 \times (1 - e^{-w_1 \times t}) \quad 5$$

  - Optimization Process:
    (1) Update the velocity $\mathbf{y}_i(t)$ to generate $\mathbf{y}_i(t+1)$:

$$\mathbf{y}_i(t+1) = \mathbf{y}_i(t) + \mathcal{U}(lb_{w_4}, ub_{w_4}) \times (\mathbf{x}_i(t) - \mathbf{x}_g(t)) \quad 6$$

    (2) Update $\mathbf{x}_i(t+1)$ to generate $\hat{\mathbf{x}}_i(t+1)$:

$$\hat{\mathbf{x}}_i(t+1) = \begin{cases} \mathbf{x}_g(t) + w_3 \times \mathbf{rand} \times z_1(t) & , \quad rand < z_2(t) \\ \mathbf{x}_i(t) + \mathbf{y}_i(t+1) & , \quad \text{o.w} \end{cases} \quad 7$$

(3) Dealing with outliers C:

$$\mathbf{x}_{i,n}^{\text{fixed}}(t+1) = \begin{cases} ub_x & , \quad \mathbf{x}_{i,n}(t+1) > ub_{\mathbf{x}} \\ \mathbf{x}_{i,n}(t+1) & , \quad \text{o.w} \\ lb_x & , \quad \mathbf{x}_{i,n}(t+1) < lb_{\mathbf{x}} \end{cases} \tag{8}$$

(4) Select $\mathbf{x}_i(t+1)$ from $\hat{\mathbf{x}}_i(t+1)$:

$$\mathbf{x}_i(t+1) = \begin{cases} \hat{\mathbf{x}}_i(t+1) & , \quad f(\hat{\mathbf{x}}_i(t+1)) < f(\mathbf{x}_i(t)) \text{ or } rand > z_1(t) \\ \mathbf{x}_i(t) & , \quad \text{o.w} \end{cases} \tag{9}$$

(5) Update $\mathbf{x}_g(t)$ to generate $\mathbf{x}_g(t+1)$:

$$\mathbf{x}_g(t+1) = \mathbf{Min}(\mathbf{x}_g(t) \cup \{\mathbf{x}_i(t+1)\}), i = 1 \dots M \tag{10}$$

(6) Update the decreasing loudness $z_1(t)$ to generate $z_1(t+1)$:

$$z_1(t+1) = z_1(t) \times w_2 \tag{11}$$

(7) Update the pulse rate $z_2(t)$ to generate $z_2(t+1)$:

$$z_2(t+1) = z_2^0 \times (1 - e^{-w_1 \times (t+1)}) \tag{12}$$

Algorithm 1 Re-framed BA with population size M; search space $n, [lb_{\mathbf{x}}, ub_{\mathbf{x}}]$; stop condition T; initialization method $Init_{\mathbf{x}}$, optimization method $Opt_{\mathbf{x}}$, treatment C of outliers, and selection S to objective solutions; initialization method $Init_{\Delta}$ and optimization method Opt_{Δ} to step-size Δ.

$t \leftarrow 0$
$\mathbf{X}(t) \leftarrow Init_{\mathbf{x}}(n, M, [lb_{\mathbf{x}}, ub_{\mathbf{x}}])$ as Eq.1 ▷ initialize initial population
$F(t) \leftarrow f(\mathbf{X}(t))$ ▷ evaluate
$w, [lb_{\mathbf{y}}, ub_{\mathbf{y}}], z^0 \leftarrow Init_{\Delta:w}(w, [lb_{\mathbf{y}}, ub_{\mathbf{y}}], z^0)$ ▷ initialize w-relative step-size
$\mathbf{Y}(t) \leftarrow Init_{\Delta:\mathbf{y}}(n, M, [lb_{\mathbf{y}}, ub_{\mathbf{y}}])$ as Eq.2 ▷ initialize $\mathbf{y}$-relative step-size
$\mathbf{x}_g(t) \leftarrow Init_{\Delta:\mathbf{x}}(\mathbf{X}(t))$ as Eq.3 ▷ initialize $\mathbf{x}$-relative step-size
$z(t) \leftarrow Init_{\Delta:z}(t, z^0, w)$ as Eq.4, Eq.5 ▷ initialize z-relative step-size
while stop condition T **do**
    $\mathbf{Y}(t+1) \leftarrow Opt_{\Delta:\mathbf{y}}(\mathbf{Y}(t), \mathbf{x}_g, w)$ as Eq.6 ▷ update $\mathbf{y}$-relative step-size
    $\hat{\mathbf{X}}(t+1) \leftarrow Opt_{\mathbf{x}}(\mathbf{X}(t), \mathbf{Y}(t), \mathbf{x}_g(t), z(t), w)$ as Eq.7 ▷ generate temporarily updated population
    $\hat{\mathbf{X}}(t+1) \leftarrow C(\hat{\mathbf{X}}(t+1))$ as Eq.8 ▷ treatment to outliers
    $F(t+1) \leftarrow f(\hat{\mathbf{X}}(t+1))$ ▷ evaluate
    $\mathbf{X}(t+1) \leftarrow S(\mathbf{X}(t), \hat{\mathbf{X}}(t+1), z(t))$ as Eq.9 ▷ select and generate finally updated population
    $\mathbf{x}_g(t+1) \leftarrow Init_{\Delta:\mathbf{x}}(\mathbf{X}(t), \mathbf{X}(t+1))$ as Eq.**??** ▷ update $\mathbf{x}$-relative step-size
    $z(t+1) \leftarrow Opt_{\Delta:z}(t+1, z(t), w)$ as Eq.11, Eq.12 ▷ update z-relative step-size
    $t \leftarrow t+1$
end while