HAL
open science

Sign gradient descent method based bat searching algorithm with application to the economic load dispatch problem

Haopeng Zhang, Qing Hui, Emmanuel Moulay, Patrick Coirault

To cite this version:

Haopeng Zhang, Qing Hui, Emmanuel Moulay, Patrick Coirault. Sign gradient descent method based bat searching algorithm with application to the economic load dispatch problem. 2020 59th IEEE Conference on Decision and Control (CDC), Dec 2020, Jeju Island (virtual), South Korea. pp.1140-1145, 10.1109/CDC42340.2020.9304286 . hal-03112304

HAL Id: hal-03112304

https://hal.science/hal-03112304v1

Submitted on 16 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Sign gradient descent method based bat searching algorithm with application to the economic load dispatch problem

Haopeng Zhang, Qing Hui, Emmanuel Moulay, Patrick Coirault

***Abstract*— Inspired by the echolocation behaviors of the bats and the swarm intelligence optimization, bat searching algorithm (BA) was developed to solve unconstrained optimization problems efficiently. However, due to the lack of the gradient term, the accuracy of the BA is not superior, and the enhancement of the algorithm is still of vital importance. The sign gradient descent method (SGD) is a first-order optimization method involving only the sign of the gradient of the function to minimize. Most importantly, the convergence and optimality issues of the SGD have been rigorously studied, which guarantees the competitive performance of SGD method. Therefore, in this paper, a combination of the BA and SGD method is proposed by integrating the SGD term into the update equation of the bats during the searching process. With the social behavior among the bats and the sign gradient descent method, the proposed algorithm shows significant improvement comparing with the original algorithm. Moreover, the convergence issue of the proposed algorithm is studied from system dynamics perspective. The numerical evaluations are provided to demonstrate the improvement of the proposed sign gradient descent method based bat searching algorithm. In the end, the economic load dispatch problem for the power system is studied as an application of the proposed BA algorithms. Based on the numerical results, the proposed BA shows superior performance.**

I. Introduction

Inspired by the echolocation behavior of microbats, Bat Algorithm (BA) is a metaheuristic for global optimization which was first proposed in [1] and widely applied to solve challenging optimization problems in various areas such as robotics[2], modern communication in [3], or energy systems [4], [5], [6]. Extensive work has been conducted to improve the performance of the original BA. In [7], a novel Bat searching algorithm (NBA) was proposed by mimicking bats habitat selection and their self-adaptive compensation for Doppler effect in echoes into the basic BA to improve the accuracy of the original BA. A bat algorithm based on iterative local search and stochastic inertial weight (ILSSI-WBA) was proposed in [8] to improve the algorithm to jump out of the local optima and to enhance the diversity and flexibility of bat population. In order to balance exploration behavior which expands the search area and exploitation behavior which favors a local minimum search, a new algorithm, known as directional Bat Algorithm (dBA) [9] or Enhanced Bat Algorithm (EBA) has been [10] proposed. Cooperative bat searching algorithm was proposed in [11] by combining the swarm intelligence and cooperative control to improve the original BA while a couple spring forced BA[12] was developed by guiding the bats with coupled springs' oscillation behavior and echolocation. Another main research direction is to extend the basic BA to address different types of optimization problems, such as, discrete optimization in [13], [14], multi-objective and many-objective optimization in [15], [16].

However, due to the lack of the gradient term, the accuracy of the BA is not superior, and the enhancement of the algorithm is still of vital importance. A first attempt to couple BA and SGD was proposed in [17] but both algorithms are totally decoupled. The sign gradient descent method (SGD) is a first-order optimization method involving only the gradient of the function to minimize. It has been first proposed in the Rprop algorithm for machine learning in [18] and its mathematical properties have been studied in [19]. One of the main interest of this method lies in the fact that the step size can be tuned precisely independently of the amplitude of the gradient avoiding the vanishing gradient problem explained in [20].

Therefore, in this paper, a combination of the BA and SGD methods is proposed by adding the SGD term into the update equations of the bats during the searching process. With the social behavior among the bats and the sign gradient descent method, the proposed algorithm shows significant improvement compared to the original algorithm when solving the benchmark functions. Moreover, the convergence issue of the proposed algorithm is studied from system dynamics perspective. The convergence condition of the proposed algorithm is provided. The Economic Load Dispatch (ELD) problem for power system is considered and numerically solved in this paper as an application. With the numerical comparison with other BA variants, the proposed algorithm shows superior performance.

The article is organized as follows. After some preliminaries given in Section II, the SGD-BA algorithm is developed in Section III. The convergence results are provided in Section IV. Numerical evaluations are provided in Section V to illustrate the efficiency of the proposed algorithms. An application problem is studied in Section VI. Finally, a conclusion is addressed in Section VII.

H. Zhang is with Department of Mechanical Engineering, University of Louisville, Louisville, KY 40292. haopeng.zhang@louisville.edu

Q. Hui is with Department of Electrical and Computer Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588-0511. qing.hui@unl.edu

E. Moulay is with DXLIM (UMR CNRS 7252), Université de Poitiers, 11 bd Marie et Pierre Curie, 86073 Poitiers Cedex 9, France. emmanuel.moulay@univ-poitiers.fr

P. Coirault is with LIAS (EA 6315), Université de Poitiers, 2 rue Pierre Brousse, 86073 Poitiers Cedex 9, France. patrick.coirault@univ-poitiers.fr

II. Preliminaries

Denoting $y = (y_1, \ldots, y_n)^T \in \mathbb{R}^n$, the gradient of a differentiable function $F : \mathbb{R}^n \to \mathbb{R}$ is the vector

$$\nabla F(y) = \left(\frac{\partial F(y)}{\partial y_1}, \ldots, \frac{\partial F(y)}{\partial y_n} \right)^T.$$

A point $y_* \in \mathbb{R}^n$ is a critical point of F if $\nabla F(y_*) = 0$. Denote

$$\texttt{sign}(\nabla F(y)) = \left(\texttt{sign}\left(\frac{\partial F(y)}{\partial y_1} \right), \ldots, \texttt{sign}\left(\frac{\partial F(y)}{\partial y_n} \right) \right)^T$$

where `sign` refers to the sign function defined by

$$\texttt{sign}(y) := \begin{cases} -1 & \text{if} \quad y < 0, \\ 0 & \text{if} \quad y = 0, \\ 1 & \text{if} \quad y > 0. \end{cases}$$

`rand` denotes a random number ranging from 0 to 1. Let $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ and $y = (y_1, \ldots, y_n) \in \mathbb{R}^n$, we denote $x \times y = (x_1\, y_1, \ldots, x_n\, y_n)$

A. BA algorithm

The BA was based on the echolocation or bio-sonar characteristics of microbats, see [1]. Let $x_i(k) \in \mathbb{R}^n$, $v_i(k) \in \mathbb{R}^n$, $f_i(k) \in \mathbb{R}$ be the position, velocity and frequency information of bat i at iteration k, respectively. We suppose that f_i belongs to set $[f_{min}, f_{max}]$. The update rules are

$$\begin{aligned} f_i(k) &= f_{min} + (f_{max} - f_{min})\beta \\ v_i(k) &= v_i(k-1) + (p(k-1) - x_i(k-1))\, f_i(k) \\ x_i(k) &= x_i(k-1) + v_i(k) \end{aligned} \tag{1}$$

where $k \in \mathbb{N}$, $1 \leq i \leq q$ where $q \in \mathbb{N}$ is the number of bats, $\beta \in [0, 1]$ is a random vector drawn from a uniform distribution and $p(k-1)$ is the best global solution of all bats at step $k-1$.

The local search is also embedded in the BA algorithm by using random walk around the current best solutions for each bat as follows

$$x_{\text{new}} \quad = \quad x_{\text{old}} + \epsilon A(k) \tag{2}$$

where $\epsilon \in [-1, 1]$ and $A(k) = \langle A_i(k) \rangle \in \mathbb{R}^n$ is the average loudness of all the bats i at time step k. The loudness A_i and pulse rate r_i for bat i are governed by the following two equations

$$\begin{aligned} A_i(k+1) &= \alpha\, A_i(k) \\ r_i(k+1) &= r_i(k)(1 - \exp(-\gamma k)) \end{aligned} \tag{3}$$

where α and γ are constants.

Let us now recall in Algorithm 1 the BA algorithm for a cost function $f : \mathbb{R}^n \to \mathbb{R}$ which must be minimized. And $\underline{x}, \overline{x}$, $\underline{v}, \overline{v}$ are the lower bounds and upper bounds for positions and velocities, respectively.

Algorithm 1 Bat Searching Algorithm

Initialize the bat's position and velocity with two uniformly distributed random vectors $x_i \sim U(\underline{x}, \overline{x})$, $v_i \sim U(\underline{v}, \overline{v})$, respectively.
Initialize the global best position p, loudness $A_i(0)$, pulse rate $r_i(0)$, and frequency $f_i(0)$.
while $k <$ MaxIter; **do**
  while $i <= q$ **do**
    Update the bat's frequency:
    $f_i(k) = f_{\min} + (f_{\max} - f_{\min})\beta$;
    Update the bat's velocity:
    $v_i \leftarrow v_i + (x_i - p) f_i$;
    Update the bat's position:
    $x_i(k) = x_i(k-1) + v_i(k)$;

    if *rand* $> r_i$ **then**
      Select a solution among the best solutions
    else
      Generate a local position around the best solution.
    end
    if *rand* $< A_i$ *and* $f(x_i) < f(p)$ **then**
      Accept the new solutions
      Increase r_i and reduce A_i
    else
      Keep the same p
    end
  end
end

III. SGD-BA algorithm

In this section, the gradient descent method is added into the update equation (1) of the original BA to obtain SGD-BA to improve the performance of the original bat searching algorithm. More precisely, the local search given by Equations (2) and (3) is replaced by SGD local search directly in Equation (1). Mathematically, it reads as

$$\begin{aligned} f_i(k) &= f_{min} + (f_{max} - f_{min})\beta \\ v_i(k) &= \omega v_i(k-1) + (p(k-1) - x_i(k-1))\, f_i(k) \\ &\quad -SGD(k-1) \\ x_i(k) &= x_i(k-1) + v_i(k) \end{aligned}$$

where ω is the weight inertia, $SGD(k-1) = d(k-1) \times \texttt{sign}(\nabla(F(x_i(k-1))))$ and $d(k) \in \mathbb{R}^n$ is the step size of the searching. Overall, the SGD-BA is described in Algorithm 2. With the combination of the BA and SGD method, the proposed SGD-BA will search and find the optimal solution with high efficiency.

In the following, we propose two approximations of the partial derivative term.

Modification 1: for $1 \leq i \leq q$ and $1 \leq j \leq n$:

$$\frac{\partial F(x_i(k))}{\partial x_{ij}(k)} \approx \frac{F(x_{i1}(k), \ldots, x_{ij}(k+1), \ldots, x_{in}(k)) - F(x_i(k))}{x_{ij}(k+1) - x_{ij}(k)} \tag{4}$$

Modification 2: for $1 \leq i \leq q$ and $1 \leq j \leq n$:

$$\frac{\partial F(x_i(k))}{\partial x_{ij}(k)} \approx \frac{F\left(X(k+1)\right) - F(X(k))}{x_{ij}(k+1) - x_{ij}(k)} \tag{5}$$

with $x_i(k) = (x_{i1}(k), \ldots, x_{in}(k)) \in \mathbb{R}^n$. Moreover, for the step size, we choose

$$d(k) = \frac{p(k-1) - x_i(k-1)}{NofB} \tag{6}$$

where $NofB$ represents the number of bats used in the algorithm. We have that $\lim_{k\to\infty} d(k) = 0$ which is a necessary condition for the convergence of the algorithm as explained in [19].

Remark 3.1: Overall, **Modification 1** is a better approximation of the partial derivative than **Modification 1**. However, SGD-BA takes more computation time when using **Modification 2** than when using **Modification 1** and it can be useful in practice to use **Modification 1**.

Algorithm 2 Sign-gradient descent based bat searching algorithm

Initialize the bat's position and velocity with two uniformly distributed random vectors $x_i \sim (lbx, ubx)$, $v_i \sim U(\underline{v}, \overline{v})$, respectively.
Initialize the global best position p, loudness $A_i(0)$, pulse rate $r_i(0)$, and frequency $f_i(0)$.

```
while k < MaxIter; do
    while i <= q do
        Update the bat's frequency:
        f_i(k) = f_min + (f_max − f_min)β;
        Update the bat's velocity:
        v_i ← ωv_i + (x_i − p)f_i − SGD;
        Update the bat's position:
        x_i(k) = x_i(k − 1) + v_i(k);

        if rand > r_i then
            Select a solution among the best solutions
        else
            Generate a local position around the best solution.
        end
        if rand < A_i and f(x_i) < f(p) then
            Accept the new solutions
            Increase r_i and reduce A_i
        else
            Keep the same p
        end
    end
end
```

IV. Convergence study

In this section, the convergence issue of the proposed SGD-BA with deterministic parameters is studied.

Theorem 4.1: All the bats in the SGD-BA converge to the global best solution if and only if $-1 < \omega < 1$, $f > 0$ and $2\omega + 2 - f > 0$.

Proof: For bat i, we have

$$\begin{aligned} f_i(k) &= f_{min} + (f_{max} - f_{min})\beta \\ v_i(k) &= \omega v_i(k-1) + (p(k-1) - x_i(k-1))\, f_i(k-1) \\ &\quad -SGD(k-1) \\ x_i(k) &= x_i(k-1) + v_i(k) \end{aligned}$$

and then,

$$\begin{aligned} x_i(k) &= x_i(k-1) + \omega v_i(k-1) \\ &\quad + (p(k-1) - x_i(k-1))\, f_i(k) \\ &\quad -SGD(k-1) \end{aligned}$$

by replacing $v_i(k-1) = x_i(k-1) - x_i(k-2)$, we have:

$$\begin{aligned} x_i(k) &= (1+\omega-f)x_i(k-1) - \omega x_i(k-2) \\ &\quad +p(k-1)f - SGD(k-1) \end{aligned}$$

The Z-transformation is applied to the above equation:

$$\begin{aligned} \mu_i(z) &= (1+\omega-f)z\mu_i(z) - \omega z^2\mu_i(z) + Z\{p(k-1)f\} \\ &\quad -Z\{SGD(k-1)\} \end{aligned}$$

where $\mu_i(z)$ is the Z-transform of $x_i(k)$. Therefore,

$$\begin{aligned} &[1-(1+\omega-f)z^{-1} + \omega z^{-2}]\mu_i(z) \\ &= Z\{p(k-1)f\} - Z\{SGD(k-1)\} \end{aligned}$$

and

$$\begin{aligned} \mu_i(z) &= \frac{Z\{p(k-1)f\}}{[1-(1+\omega-f)z^{-1} + \omega z^{-2}]} \\ &\quad - \frac{Z\{SGD(k-1)\}}{[1-(1+\omega-f)z^{-1} + \omega z^{-2}]} \end{aligned}$$

$$\begin{aligned} \mu_i(z) &= \frac{\frac{1}{z^{-1}-1} fP^*}{[1-(1+\omega-f)z^{-1} + \omega z^{-2}]} \\ &\quad - \frac{Z\{SGD(k-1)\}}{[1-(1+\omega-f)z^{-1} + \omega z^{-2}]} \end{aligned}$$

Based on the final value theorem:

$$\begin{aligned} &\lim_{k\to\infty} x_i(k) = \lim_{z\to 1} \mu_i(z)(z-1) \\ &= \lim_{z\to 1} \frac{zfP^*}{[1-(1+\omega-f)z^{-1} + \omega z^{-2}]} \\ &- \lim_{z\to 1}(z-1)\frac{Z\{SGD(k-1)\}}{[1-(1+\omega-f)z^{-1} + \omega z^{-2}]} \end{aligned}$$

To grantee the stability of the system, the roots of $[1-(1+\omega-f)z^{-1} + \omega z^{-2}]$ should be inside the unit circle, and we have

- $-1 < \omega < 1$
- $f > 0$
- $2\omega + 2 - f > 0$

and $\lim_{k\to\infty} x_i(k) = p$. ■

A numerical example is provided here. Assume three bats are considered in the SGD-BA algorithm to solve the minimization problem of objective function $f(x) = x^2$. Let $\omega = 0.8$, and $f_i = 2$, then the positions and velocities of the bats during the searching process are shown in Fig. 1 and Fig. 2, which show the convergence of the algorithm. Moreover, the best solutions found in each iteration are shown in Fig. 3, and the fitness values during the searching process are provided in Fig. 4.

V. Numerical evaluations

In this section, the proposed SGD-BAs are numerically evaluated.

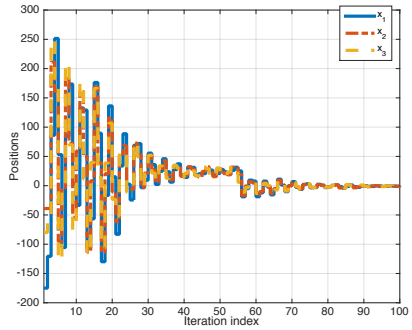

Fig. 1. Convergence of the SGD-BA: positions

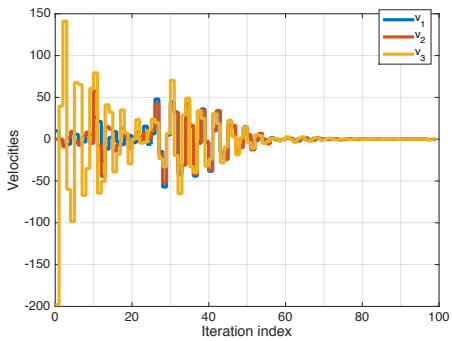

Fig. 2. Convergence of the SGD-BA: velocities

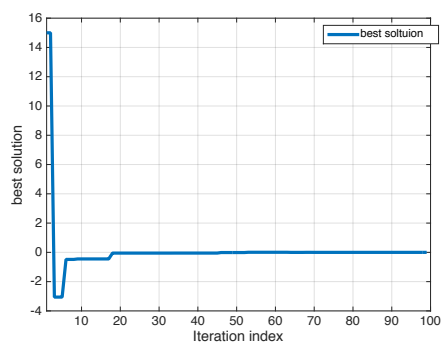

Fig. 3. Convergence of the SGD-BA: best solution

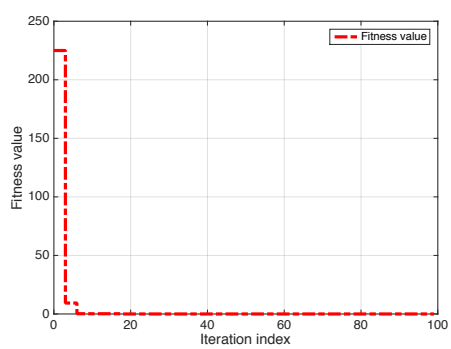

Fig. 4. Convergence of the SGD-BA: fitness value

A. Benchmark functions

The standard benchmark functions are chosen from the 2018 IEEE Congress on Evolutionary Computation (CEC). Due to the space limitation, all the functions are not fully reviewed, for more information, one can check [21].

1) Fully-separable Functions
   a) f_1: Elliptic Function
   b) f_2: Rastrigin Function
   c) f_3: Ackley Function
2) Partially Additively Separable Functions
   - Functions with a separable subcomponent:
     a) f_4: Elliptic Function
     b) f_5: Rastrigin Function
     c) f_6: Ackley Function
     d) f_7: Schwefels Problem 1.2
   - Functions with no separable subcomponents:
     a) f_8: Elliptic Function
     b) f_9: Rastrigin Function
     c) f_{10}: Ackley Function
     d) f_{11}: Schwefels Problem 1.2
3) Overlapping Functions
   a) f_{12}: Rosenbrock's Function
   b) f_{13}: Schwefels Function with Conforming Overlapping Subcomponents
   c) f_{14}: Schwefels Function with Conflicting Overlapping Subcomponents
4) . Non-separable Functions
   a) f_{15}: Schwefels Problem 1.2

B. Numerical results

All the algorithms run 20 times with random initial positions with 100 bats. The max, min, mean and the standard deviation values are calculated and are compared, which are given in Tables I–VI. In the last line of the tables, number of the best solutions are listed. Overall, equation (4) being a better approximation of the partial derivative than Equation (5), it explains why we obtain better results for SGD-BA with Equation (4) than with Equation (5) in Table IV. However, SGD-BA takes more computation time when using Equation (4) than when using Equation (5) and it can be useful in practice to use Equation (5).

VI. Application to Economic Load Dispatch Problem

To demonstrate the performance of the SGD method and the proposed BA algorithms, the Economic Load Dispatch (ELD) problem for power systems is considered and numerically solved in this section as an application. The ELD problem considers a series of power generation and power transmission facilities, each with associated operating limits, costs, and losses. Given a power demand for the entire grid, the optimal outputs for each generating station must be determined in order to minimize overall cost, see [22] for details. Due to the nonlinear cost functions and the presence of both inequality and equality constraints, the ELD

TABLE I

MAXIMUM COMPARISONS BETWEEN SGD-BA(4) USES EQUATION (4) AND SGD-BA(5) USES EQUATION (5)

Functions	BA	SGD-BA(4)	SGD-BA(5)
f_1	2.2067e+11	2.2055e+11	2.2042e+11
f_2	4.9380e+04	5.1108e+04	5.1018e+04
f_3	2.1232e+01	2.1005e+01	2.1208e+01
f_4	2.3045e+13	2.0412e+13	2.0674e+13
f_5	4.6616e+07	4.0791e+07	5.0117e+07
f_6	1.0490e+06	1.0421e+06	1.0437e+06
f_7	4.4566e+14	2.5223e+14	2.7483e+14
f_8	7.0595e+17	8.6122e+17	9.1962e+17
f_9	3.9952e+09	3.6655e+09	3.8955e+09
f_{10}	9.5325e+07	9.5430e+07	9.5730e+07
f_{11}	1.8293e+16	1.7352e+16	1.6622e+16
f_{12}	2.3174e+12	2.9851e+12	2.4451e+12
f_{13}	6.5514e+12	5.2442e+12	5.7714e+12
f_{14}	3.1320e+16	3.7821e+16	3.6381e+16
f_{15}	3.8474e+14	1.0233e+14	1.1123e+14
# of the best	5 out of 15	8 out of 15	2 out of 15

TABLE II

MINIMUM COMPARISONS BETWEEN SGD-BA(4) USES EQUATION (4) AND SGD-BA(5) USES EQUATION (5)

Functions	BA	SGD-BA(4)	SGD-BA(5)
f_1	1.8409e+11	1.8328e+11	1.8248e+11
f_2	4.3991e+04	4.4003e+04	4.3773e+04
f_3	2.1129e+01	2.1003e+01	2.1113e+01
f_4	6.4297e+12	3.0125e+12	3.2055e+12
f_5	3.4032e+07	3.9709e+07	3.7870e+07
f_6	1.0211e+06	1.01071e+06	1.0203e+06
f_7	3.6324e+13	3.8211e+13	4.1261e+13
f_8	2.0124e+17	1.9841e+17	1.6494e+17
f_9	2.6673e+09	2.6099e+09	2.6569e+09
f_{10}	9.0103e+07	8.9881e+07	8.7831e+07
f_{11}	2.6236e+15	2.5429e+15	2.7939e+15
f_{12}	1.9961e+12	2.0186e+12	1.9886e+12
f_{13}	3.1547e+11	3.0626e+11	3.2266e+11
f_{14}	2.9560e+15	2.3127e+15	1.8397e+15
f_{15}	2.6404e+11	2.9621e+11	3.2036e+11
# of the best	3 out of 15	7 out of 15	7 out of 15

TABLE III

STANDARD DEVIATION COMPARISONS BETWEEN SGD-BA(4) USES EQUATION (4) AND SGD-BA(5) USES EQUATION (5)

Functions	BA	SGD-BA(4)	SGD-BA(5)
f_1	9.1698e+09	1.2123e+10	1.0144e+10
f_2	1.2191e+03	1.6361e+03	1.7831e+03
f_3	2.7249e-02	2.9941e-02	2.7894e-02
f_4	4.2594e+12	4.1901e+12	4.4449e+12
f_5	2.7758e+06	2.6825e+06	2.7652e+06
f_6	6.8454e+03	6.5851e+03	5.8550e+03
f_7	1.0716e+14	6.7881e+13	6.1388e+13
f_8	1.3153e+17	1.6193e+17	1.3103e+17
f_9	3.1330e+08	2.8234e+08	2.7825e+08
f_{10}	1.4175e+06	1.2972e+06	1.7479e+06
f_{11}	3.9061e+15	3.5565e+15	3.5565e+15
f_{12}	8.0107e+10	9.8551e+10	1.2875e+11
f_{13}	1.6392e+12	1.1441e+12	1.3254e+12
f_{14}	8.3149e+15	7.6873e+15	7.9783e+15
f_{15}	8.5946e+13	1.9634e+13	2.4633e+13
# of the best	4 out of 15	6 out of 15	5 out of 15

TABLE IV

MEAN VALUE COMPARISONS BETWEEN SGD-BA(4) USES EQUATION (4) AND SGD-BA(5) USES EQUATION (5)

Functions	BA	SGD-BA(4)	SGD-BA(5)
f_1	2.0517e+11	2.0321e+11	2.0084e+11
f_2	4.6593e+04	4.6823e+04	4.6483e+04
f_3	2.1177e+01	2.0614e+01	2.1164e+01
f_4	1.0324e+13	9.2241e+12	9.4264e+12
f_5	3.9574e+07	3.6561e+07	3.8886e+07
f_6	1.0368e+06	1.0011e+06	1.0363e+06
f_7	1.9538e+14	9.8141e+13	1.1431e+14
f_8	4.3026e+17	4.2181e+17	4.1478e+17
f_9	3.4215e+09	3.3323e+09	3.2352e+09
f_{10}	9.2536e+07	9.2750e+07	9.3008e+07
f_{11}	6.8157e+15	7.0711e+15	6.1711e+15
f_{12}	2.1201e+12	2.0853e+12	2.1583e+12
f_{13}	1.8864e+12	1.7783e+12	1.4487e+12
f_{14}	1.2638e+16	1.0051e+16	1.1155e+16
f_{15}	3.6229e+13	1.1414e+13	1.2447e+13
# of the best	1 out of 15	8 out of 15	6 out of 15

problem is a rigorous test that offers insight into how different configurations affect the performance of an algorithm. Mathematically, the ELD problem can be expressed as:

$$\min : \sum_{i=1}^{n} F_i(P_i) \tag{7}$$

$$\text{subject to} : P_i^{min} \leq P_i \leq P_i^{max}, \tag{8}$$

$$\sum_{i=1}^{n} P_i = D + P_l, \tag{9}$$

$$P_l = \sum_{i=1}^{n} \sum_{i=j}^{n} B_{i,j} P_i P_j, \tag{10}$$

$$F_{i,j}\left(P\left(i\right)\right) = a_i P_i^2 + b_i P_i + c_i \tag{11}$$

where the system consists of n generating stations each with a load of P_i. The fuel cost for each plant is given by the quadratic function $F_i(P_i)$. The load for each generating station is limited by the lower and upper bounds P_i^{min} and P_i^{max}, respectively, while the total load is P_l. The transmission losses are described by the B matrix.

In this section, the BA, SGD-BA(4) and SGD-BA(5) are applied to solve the ELD problem (7)-(11). 30 bats are considered in all the BA algorithms with 500 iterations. 20 executions have been conducted, and the max, min, standard derivation (SD), mean, and median values are compared and provided in Table. V. Based on the simulation results, the proposed two sign-gradient based BA significantly improve the original BA. And the second modification showed the best results.

VII. CONCLUSION

In this paper, two sign-gradient descent modifications are applied to the bat searching algorithm in order to enhance the accuracy of the BA. Due to the gradient term in the proposed algorithms, the SGD-BAs show competitive improvement compared with the original BA by solving the 2018 IEEE Congress on Evolutionary Computation's standard benchmark functions. Moreover, the convergence analysis of the proposed algorithm has been conducted, and the require-

TABLE V

COMPARISON BETWEEN DIFFERENT BA ALGORITHMS FOR ELD USING 30 BATS

BA	Max	Min	SD	Mean	Median
BA	5.9738e+05	5.9465e+05	8.3622e+01	5.9684e+05	5.9725e+05
SGD-BA(4)	5.2540e+05	4.7899e+05	1.5552e+04	4.9852e+05	4.9472e+05
SGD-BA(5)	4.5768e+05	4.5695e+05	2.1379e+02	4.5727e+05	4.5729e+05

TABLE VI

MEDIAN VALUE COMPARISONS BETWEEN SGD-BA(4) USES EQUATION (4) AND SGD-BA(5) USES EQUATION (5)

Functions	BA	SGD-BA(4)	SGD-BA(5)
f_1	2.0723e+11	2.1201e+11	2.0070e+11
f_2	4.6308e+04	4.6276e+04	4.6605e+04
f_3	2.1180e+01	2.0987e+01	2.1165e+01
f_4	8.4026e+12	8.9559e+12	8.9925e+12
f_5	4.0037e+07	4.2661e+07	4.1156e+07
f_6	1.1351e+06	1.0394e+06	1.0371e+06
f_7	1.8825e+14	1.7797e+14	1.0791e+14
f_8	4.1820e+17	3.7858e+17	3.9188e+17
f_9	3.3901e+09	3.3451e+09	3.2382e+09
f_{10}	9.2416e+07	9.1981e+07	9.2466e+07
f_{11}	5.5110e+15	6.6475e+15	6.7615e+15
f_{12}	2.1136e+12	2.4681e+12	2.1063e+12
f_{13}	1.3413e+12	1.7013e+12	1.0074e+12
f_{14}	9.8651e+15	9.9818e+16	1.0428e+16
f_{15}	9.1951e+12	4.818569e+12	4.8185e+12
# of the best	4 out of 15	4 out of 15	6 out of 15

ments of the parameters are derived for convergence. In the end, the ELD problem is considered and numerically solved by the three bat algorithms. Based on the numerical results, the proposed sign-gradient BAs show better performance than the original BA and two other BA variations in the literature.

REFERENCES

[1] X.-S. Yang, "A new metaheuristic BAT-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*. Springer, 2010, pp. 65–74.

[2] P. Suárez, A. Iglesias, and A. Gálvez, "Make robots be bats: specializing robotic swarms to the bat algorithm," *Swarm and Evolutionary Computation*, vol. 44, pp. 113–129, 2019.

[3] A. Das, D. Mandal, S. Ghoshal, and R. Kar, "An efficient side lobe reduction technique considering mutual coupling effect in linear array antenna using bat algorithm," *Swarm and Evolutionary Computation*, vol. 35, pp. 26–40, 2017.

[4] B. Bahmani-Firouzi and R. Azizipanah-Abarghooee, "Optimal sizing of battery energy storage for micro-grid operation management using a new improved bat algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 56, pp. 42–54, 2014.

[5] E. Ali, "Optimization of power system stabilizers using bat search algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 61, pp. 683–690, 2014.

[6] B. Adarsh, T. Raghunathan, T. Jayabarathi, and X.-S. Yang, "Economic dispatch using chaotic bat algorithm," *Energy*, vol. 96, pp. 666–675, 2016.

[7] X.-B. Meng, X. Z. Gao, Y. Liu, and H. Zhang, "A novel bat algorithm with habitat selection and doppler effect in echoes for optimization," *Expert Systems with Applications*, vol. 42, no. 17-18, pp. 6350–6364, 2015.

[8] C. Gan, W. Cao, M. Wu, and X. Chen, "A new bat algorithm based on iterative local search and stochastic inertia weight," *Expert Systems with Applications*, vol. 104, pp. 202–212, 2018.

[9] A. Chakri, R. Khelif, M. Benouaret, and X.-S. Yang, "New directional bat algorithm for continuous optimization problems," *Expert Systems with Applications*, vol. 69, pp. 159–175, 2017.

[10] S. Yılmaz and E. U. Küçüksille, "A new modification approach on bat algorithm for solving optimization problems," *Appl. Soft Comput.*, vol. 28, pp. 259–275, 2015.

[11] H. Zhang and Q. Hui, "Cooperative bat searching algorithm: A combined perspective from multiagent coordination and swarm intelligence," in *Proc. IEEE Autom. Sci. Eng.*, 2017, pp. 1362–1367.

[12] ——, "A coupled spring forced bat searching algorithm: Design, analysis and evaluation," in *Proc. Amer. Control Conf.*, 2020, pp. 5016–5021.

[13] S. Mirjalili, S. M. Mirjalili, and X.-S. Yang, "Binary bat algorithm," *Neural Computing and Applications*, vol. 25, no. 3-4, pp. 663–681, 2014.

[14] H. Zhang, "A binary cooperative bat algorithm based optimal topology design of leader-follower consensus," *ISA Trans.*, vol. 96, pp. 51–59, 2020.

[15] M. Tuba and N. Bacanin, "Hybridized bat algorithm for multi-objective radio frequency identification (rfid) network planning," in *2015 IEEE congress on evolutionary computation (CEC)*, 2015, pp. 499–506.

[16] H. Zhang and Q. Hui, "Many objective cooperative bat searching algorithm," *Applied Soft Computing*, vol. 77, pp. 412–437, 2019.

[17] M. P. Reddy and R. Ganguli, "An automated gradient enhanced bat algorithm," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2018, pp. 2353–2360.

[18] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: the RPROP algorithm," in *IEEE International Conference on Neural Networks*, 1993, pp. 586–591.

[19] E. Moulay, V. Léchappé, and F. Plestan, "Properties of the sign gradient descent algorithms," *Information Sciences*, vol. 492, pp. 29–39, 2019.

[20] K.-L. Du and M. N. Swamy, *Neural networks and statistical learning*, ser. Computational Intelligence and Complexity. Springer, 2013.

[21] X. Li, K. Tang, M. N. Omidvar, Z. Yang, K. Qin, and H. China, "Benchmark functions for the CEC'2013 special session and competition on large-scale global optimization," *Technical Report, Evolutionary Computation and Machine Learning Group, RMIT University, Australia*, 2013.

[22] J.-B. Park, K.-S. Lee, J.-R. Shin, and K. Y. Lee, "A particle swarm optimization for economic dispatch with nonsmooth cost functions," *IEEE Transactions on Power Systems*, vol. 20, no. 1, pp. 34–42, 2005.