IECO International Journal Of Industrial Electronics Control and Optimization
Print ISSN: 3517-2645
Online ISSN: 3568-2645
Homepage: https://ieco.usb.ac.ir/

Bio-Inspired Optimization: A New Meta-heuristic Based on Animal Adaptability

Vahid Shirzad-iraj[1] | Reza Keypour[2*]

Faculty of Electrical and Computer Engineering, Semnan University, Semnan, Iran[1,2]

rkeypour@semnan.ac.ir

Article Info

Article type:
Research Article

Article history:
Received: *************
Received in revised form: *************
Accepted: *************
Published online: ************

Keywords:
Animals' adaptability algorithm,
meta-heuristic algorithm,
optimization,
Decision-making.

ABSTRACT

Objective: Organisms continuously adapt to changing resources and environmental conditions to ensure survival. Such adaptations may involve shifts in diet, habitat, or hunting strategies. Species that fail to adapt often face extinction, as evidenced by many species that no longer exist. Adaptation can generally be understood as a two-stage process. The first stage encompasses phenotypic adjustments, which occur in response to factors such as climate change, food availability, or the development of adaptive behaviors. These short-term changes increase the chances of survival for individuals by favoring the fittest under immediate conditions. The second stage involves genetic changes, where offspring chromosomes are formed through crossover and mutation of parental chromosomes. These genetic variations may either enhance the survival of the next generation or reduce it, thereby influencing long-term evolutionary success. In this study, a novel metaheuristic algorithm is introduced, inspired by the adaptability of socially living animals. The proposed algorithm was evaluated on a set of standard benchmark functions and compared with several well-established optimization methods. The results indicate that the proposed algorithm outperforms others in terms of convergence speed, solution accuracy, and robustness. Finally, to demonstrate its practical applicability, the algorithm was applied to an engineering problem involving air-conditioner load scheduling, where it achieved superior performance.

NOMENCLATURE

T_{in} (°C)	Tripping time of the *i*-th relay.	P_{cool} (W)	electrical power consumption of the AC system
C (J/°C)	thermal capacity of the room, accounting for heat storage within the room's air and surfaces.	COP	Coefficient Of Performance of the AC, which measures its efficiency in converting electrical energy into cooling capacity.
Q_{cool} (w)	cooling power supplied by the AC system.	Cost	Total electricity cost over the optimization period.
Q_{loss} (w)	heat losses due to thermal conduction and convection through walls and windows.	N	Number of time intervals considered.
U (W/m²°C)	heat transfer coefficient of walls and windows, indicating their insulation efficiency.	$T_{min/max}$ (°C)	the acceptable range for indoor temperature, ensuring thermal comfort
A (m^2)	the total surface area of walls and windows exposed to external conditions.	$Q_{cool,min/max}$ (w)	specify the operational limits of the AC cooling power.
APP_i^{new}	outdoor temperature.	Price(t)	the electricity price at time t.

I. Introduction

Metaheuristic algorithms have become essential tools for solving complex optimization problems in engineering, computer science, and resource management. These algorithms provide effective strategies for exploring large and complex search spaces, offering solutions where conventional optimization methods often fail. Their flexibility and wide applicability have led to extensive adoption in real-world problems, including network design, scheduling, and industrial optimization. Although many metaheuristics are inspired by natural or social phenomena, their effectiveness ultimately depends on maintaining an appropriate trade-off between global exploration and local refinement within the search space.

Over the past decades, numerous metaheuristic algorithms have been proposed, each addressing specific optimization challenges. For example, Genetic Algorithms (GA) provide robust global search capabilities but may exhibit slow convergence [1-4]. Similarly, Ant Colony Optimization (ACO) effectively tackles routing and combinatorial problems, though at a high computational cost [5-6]. In contrast, Particle Swarm Optimization (PSO) and Differential Evolution (DE) perform well in continuous optimization; however, PSO can be trapped in local optima, while DE may converge slowly in highly multi-modal landscapes [7-10]. Swarm-based approaches, such as the Artificial Bee Colony (ABC) algorithm, are simple and require few parameters, yet they often struggle with more complex problems [11-12]. The Firefly Algorithm (FA) shows strong performance in multi-objective optimization but tends to converge slowly [13], In Contrast, Cuckoo Search (CS) can escape local optima but is highly sensitive to problem scaling [14]. Likewise, the Bat Algorithm (BA) offers a reasonable balance between diversification and intensification, though it may underperform in large-scale scenarios [15]. Teaching-Learning-Based Optimization (TLBO) is parameter-free and straightforward approach, but its convergence rate slows considerably in high-dimensional problems [16]. More recent algorithms, including the Grey Wolf Optimizer (GWO), Moth-Flame Optimization (MFO), and Whale Optimization Algorithm (WOA), aim to enhance the exploration–exploitation balance and improve global search capability. Nevertheless, issues such as premature convergence, parameter sensitivity, and inconsistent performance in complex or high-dimensional problems remain [17-21]. Collectively, these methods highlight substantial progress in metaheuristic optimization while revealing persistent limitations.

Despite these advances, convergence speed and solution accuracy remain critical challenges. These issues are further exacerbated as the dimensionality of the problem increases, with many existing algorithms unable to maintain both high-quality solutions and rapid convergence in large-scale, complex scenarios. Therefore, there is a clear need for algorithms that can overcome these limitations while preserving an effective balance between exploration and refinement.

Thus, despite the significant progress in metaheuristic design, there is still a lack of algorithms that simultaneously ensure fast convergence and high-quality solutions in large-scale, complex problems. This gap motivates the development of the proposed Animal Adaptability Algorithm(AdA).

In this study, inspired by the adaptability of animals and their responses to changing environments, the Animal Adaptability Algorithm(AdA) is introduced. AdA leverages mechanisms inspired by territorial and social behaviors to achieve faster convergence, higher accuracy, and greater robustness in complex optimization problems. By simulating natural processes such as search space refinement, competition among individuals, and the generation of new offspring, AdA enables effective diversification and intensification of solutions simultaneously, thereby making it particularly suitable for a wide range of challenging optimization tasks in engineering, resource management, and network design.

II. Methodology

A. Fundamental Concept of the Animal Adaptability Algorithm

Some animal species, including cheetahs, tigers, wild boars, and songbirds, establish specific territories for survival and reproduction. This territorial behavior is a natural phenomenon from the perspective of behavioral biology. Moreover, it can serve as an inspiring model for designing search and optimization algorithms in the field of artificial intelligence. Animals with greater physical strength and health are capable of claiming richer territories. In algorithmic terms, this is analogous to stronger agents or better solutions occupying more promising regions in the search space. Consequently, access to abundant resources increases the likelihood of survival and, ultimately, the probability of transmitting favorable traits to the next generation.

The foraging behavior within a territory can also be compared to the search process in metaheuristic algorithms. Instead of conducting a random and dispersed search, animals utilize experience and knowledge acquired from their parents to narrow their search domain, thereby finding higher-quality resources. Similarly, in algorithms, the search usually begins in broad areas, but as the process progresses and memory and learning are incorporated, the focus gradually shifts toward more promising regions. In the AdA algorithm, this natural behavior is modeled using a distance-halving method, whereby the search space is progressively reduced to facilitate access to more promising points.

Furthermore, the competition among animals to maintain their territories and repel same-sex rivals is analogous to the selection operation in evolutionary algorithms, where only the strongest agents survive and contribute more to the formation of the next generation. Subsequently, the process of generating new individuals begins, which corresponds to the crossover and mutation operations in evolutionary algorithms. These processes enable the transmission of genetic information from parents to offspring and introduce diversity into the population.

It should be noted that the distance-halving method is not restricted to unimodal functions; it primarily accelerates convergence near promising solutions, while other operators—such as crossover, mutation and absorption/repulsion—continue to explore the broader search space for global optima.

B. Implementation of Adaptability Algorithms

The mathematical modeling of the proposed adaptability algorithm is based on the following hypotheses:

I. Animals that locate superior food sources are more likely to overcome their competitors.

II. Both the behavioral and genetic characteristics of animals evolve when they encounter novel or challenging circumstances.

III. It is feasible to assess the simultaneous effects of rivalry and individual performance in a given environment.

IV. The superior traits of earlier generations are maintained and will persist if no more adaptable individuals are produced in subsequent generations.

B.1 Initialization of the Population

The optimization process begins when species, such as birds, mammals, or aquatic animals, colonize a new environment. To model this scenario, an initial population of potential solutions is generated within a feasible search space. Each solution in the population represents a distinct characteristic of the animals. Formally, the initial population can be denoted as:

$$[APP_1, APP_2, \dots, APP_N], \quad NP=\text{Population size} \qquad (1)$$

, representative of distinctive feature of the animal i.

B.2 Modeling Animal Decision-Making

Animals often face numerous challenges and threats in their environment, prompting them to make decisions such as selecting food sources or relocating to new habitats. These decisions can either improve or impair their survival and well-being. To model this process, a unimodal function is used, as the goal is for the animal to find the optimal solution. If the animal's decision is correct, it progresses toward its goal, but an incorrect decision may lead to weakening or even death.

The probability of succeeding through random decision-making without prior knowledge is low, and the consequences may be irreversible. Therefore, animals tend to rely on the knowledge and experiences of other group members. This social learning behavior is common in species such as whales, dolphins, and others. In this study, the interval halving method is employed to model the decision-making process. As depicted in Figure 1(a), the method eliminates 50% of the information from group members whose results are weaker.

For example, if a disease spreads in the population, animals avoid high-risk areas and move toward safer regions [a, x0]. In the interval halving approach, information from the group is gathered, and decisions are made within the search space. Given that decision-making assumes the unimodal nature of the goal function, 50% of the search space is discarded. Ignoring the current situation or an animal's prior experience would be irrational and may lead to suboptimal decisions. As a result, the animal's own experiences are integrated into the search space, reducing the search space to the region [x1, x0], as shown in Figure 1(b).

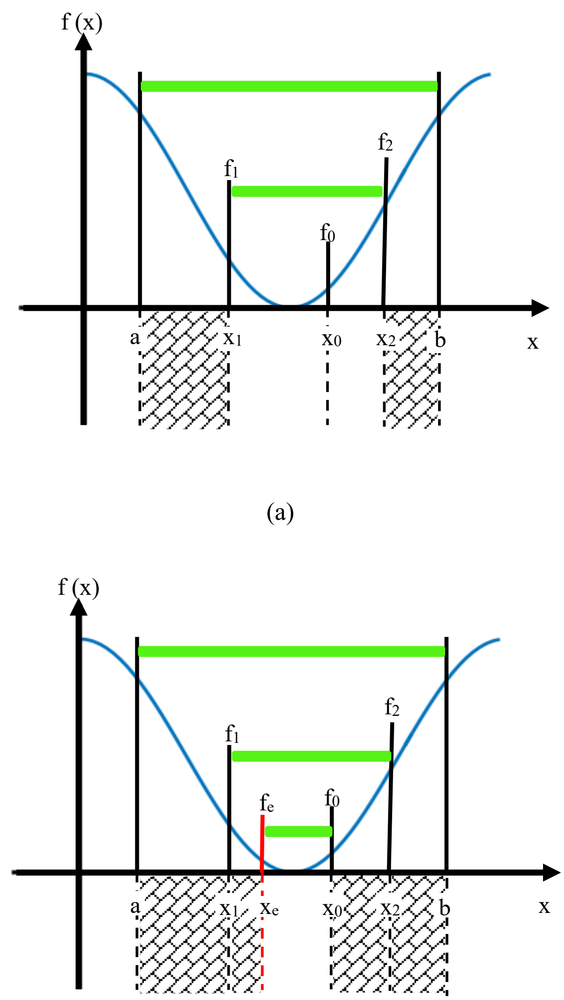

Figure 1: (a) Interval halving method, (b) Integration of the animal's experience with collective knowledge from the group

When the gathered information is accurate, new decisions lead to improved conditions. However, if the information is incorrect, the decision-maker may either recognize and discard it, or it may result in erroneous decisions. When the collected information contradicts the unimodal function, the information will be eliminated, and the decision-making area will be updated.

Once the new search space has been identified, the decision-maker proceeds with the search for an optimal solution. If the decision improves the animal’s condition, it will reach its goal. However, an incorrect decision will push the animal away from the optimal solution and create opportunity for competitors to capitalize on the mistake.

B.3 Repulsion and Attraction

Male animals, considering their physical condition, engage in competition to define and defend territories. The strongest animals control larger territories, and as a result, they attract more mates, thereby playing a significant role in the formation of the next generation. This competitive behavior is modeled by the following equation:

$$r_{i,j} = \left|APP_j^{old} - APP_i^{old}\right|$$
$$APP_i^{new} = APP_i^{old} + e^{\frac{-1}{r_{i,j}^2}}(APP_j^{old} - APP_i^{old}) \tag{2}$$
$$APP_k^{new} = APP_j^{old} - e^{\frac{-1}{r_{j,k}^2}}(APP_j^{old} - APP_k^{old})$$

In Equation (2), the variable APP_i^{old} on the right-hand side represents the previous position of agent i, while the variable APP_i^{new} on the left-hand side denotes the updated (new) position of agent and $r_{i,j}$ shows Euclidean distance of the two animals APP_i^{old} and APP_j^{old}.

B.4 Creating new generation and teaching them

After forming territories, parents meet each other and create new generations. And the new generation will inherit genetic characteristics from their parents. In order to model this section, mutation and crossover were used.

B.5 Crossover

As it was mentioned earlier, new generations inherit their parents characteristics. the simplest methods choose one or more points in the chromosome to mark as the crossover points. Then the variables between these points are merely swapped between two parents. Consider the two parents to be:

$$APP_i = [APP_{i,1}, APP_{i,2}, \ldots, APP_{i,h}, \ldots, APP_{i,N}]$$
$$APP_j = [APP_{j,1}, APP_{j,2}, \ldots, APP_{j,h}, \ldots, APP_{j,N}] \tag{3}$$

In Equation (3), $APP_{i,h}$ represents h-th dimension of the i-th agent position or equivalently h-th decision variable of the ith individual and N indicates number of decision variables of the optimization problem.

In this method, parents are selected according to roulette wheel mechanism.

Crossover points are randomly selected, and the variables in crossover point present:

$$APP_{i,h}^{new} = APP_{i,h} - \beta\left(APP_{i,h} - APP_{j,h}\right)$$
$$APP_{j,h}^{new} = APP_{j,h} + \beta\left(APP_{i,h} - APP_{j,h}\right) \tag{4}$$

β is a random number; It should be noted that the range of the β parameter in the intersection mechanism is limited to [0, 1], and a boundary correction strategy is applied to ensure that generated offspring remain within the feasible solution space.

The new generation will be formed as follows:

$$APP_i^{new} = [APP_{i,1}, \ldots, APP_{i,h-1}, APP_{i,h}^{new}, APP_{j,h+1}, \ldots, APP_{j,N}]$$
$$APP_j^{new} = [APP_{j,1}, \ldots, APP_{j,h-1}, APP_{j,h}^{new}, APP_{i,h+1}, \ldots, APP_{i,N}] \tag{5}$$

B.6 Mutation

During the formation of a new generation, genetic mutations may introduce variations between offspring and their parents. These mutations not only account for random changes in inherited traits but also enable the acquisition of new skills. Unlike their parents, the new generation benefits both from inherited experiences and trial-and-error learning, allowing them to develop capabilities that were not present in previous generations. This process can be effectively modeled using the mutation operator.

$$APP_i^j = \left(var\,hi_{APP_i}^j - var\,lo_{APP_i}^j\right) * rand + var\,lo_{APP_i}^j \tag{6}$$

Flowchart of the algorithm is shown in Figure 2.

The following pseudo-code summarizes all main operations of the algorithm.

Table 1: pseudo-code for AdA Algorithm

```
Initialize agents randomly.
Evaluate the fitness
BestSolution = best agent.
for iter = 1 to MaxIter do
    Reduce the search space using
      distance-halving.
    Apply attraction-repulsion adjustment.
    Generate offspring via crossover.
    Apply mutation to offspring.
    Evaluate updated fitness.
    Update BestSolution if improved.
end for
Return BestSolution.
```

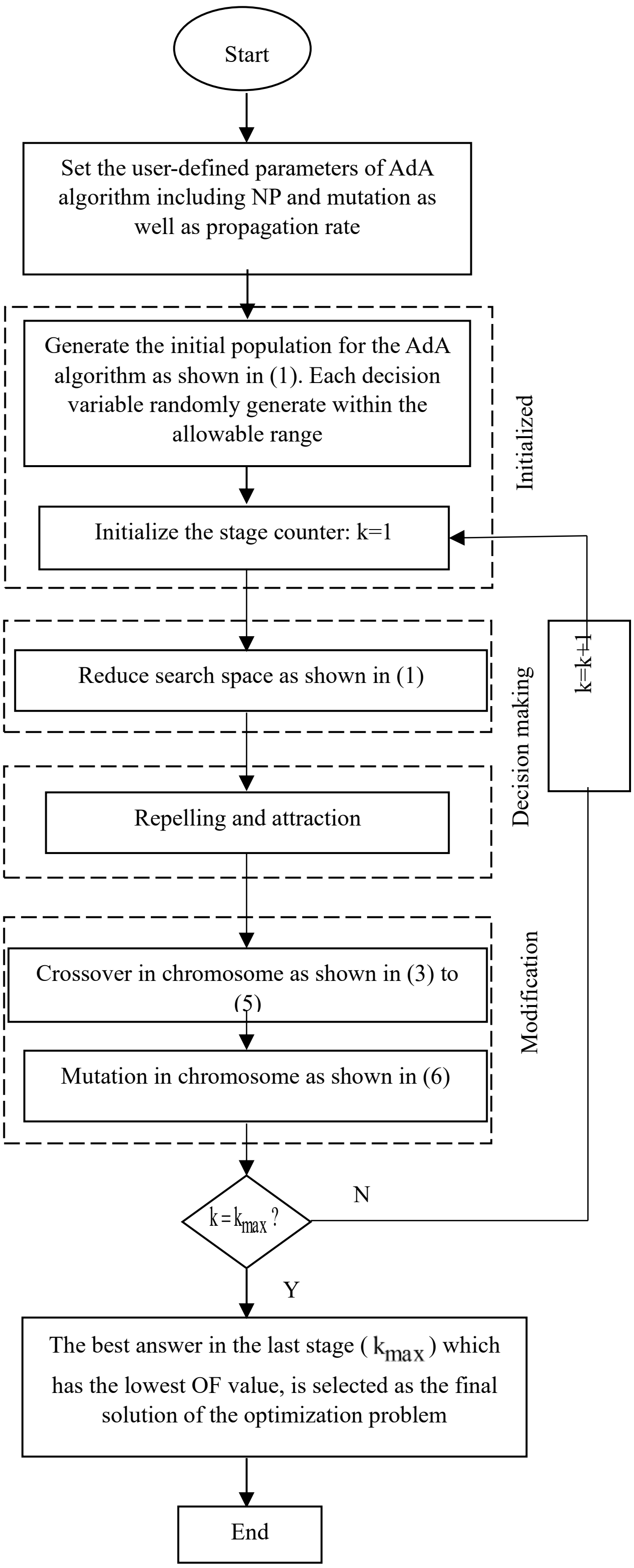

Figure 2: Flowchart of the algorithm

C. Test function

To evaluate the accuracy and efficiency of the proposed method, a comprehensive set of standard benchmark functions, as outlined in Table 2, has been employed. Furthermore, the proposed algorithm has been tested in a real-world scenario involving the optimal scheduling of air conditioning systems to minimize energy consumption and associated costs. The results of this practical evaluation highlight the algorithm's effectiveness in addressing complex real-world optimization problems, demonstrating its strong potential for deployment in practical applications.

In the following, a dynamic model[23] is presented for optimizing electricity costs and regulating indoor temperature (Tin), aiming to balance air conditioner (AC) energy consumption with thermal comfort. This model incorporates cooling power, heat losses, and internal heat sources, enabling intelligent analysis and management of HVAC systems. The indoor temperature dynamically evolves according to Equation (7):

$$C\frac{dT_{in}}{dt} = Q_{cool} - Q_{loss} \tag{7}$$

Heat losses through walls and windows are modeled using Equation (8):

$$Q_{loss} = U.A.(T_{in} - T_{out}) \tag{8}$$

The AC power consumption is determined based on its cooling capacity and coefficient of performance (COP), as expressed in Equation (9):

$$P_{cool} = \frac{Q_{cool}}{COP} \tag{9}$$

The objective function for minimizing electricity costs is formulated in Equation (10):

$$\begin{aligned} &\mathrm{Cost} = \sum_{t=1}^{N} P_{cool}(t).\,\Delta t.\,\mathrm{Price}(t) \\ &\text{s.t.} \\ &T_{min} \le T_{in} \le T_{max} \\ &Q_{cool,min} \le Q_{cool} \le Q_{cool,max} \end{aligned} \tag{10}$$

Table 2: test functions

No	Benchmark Function	Function	Absolute minimum
1	Rastrigin Benchmark Function	$f_1(x) = 20 + \sum_{k=1}^{2}\left(x_k^2 - 10.\cos(2.\pi.x_k)\right)$ $-20 \le x_k \le 20$	0
2	Griewangk Benchmark Function	$f_2(x) = \sum_{k=1}^{2}\frac{x_k^2}{4000} - \prod_{k=1}^{2}\cos\left(\frac{x_k}{\sqrt{k}}\right) + 1$ $-600 \le x_k \le 600$	0
3	Schaffer N.2 Benchmark Function	$f_3(x_1, x_2) = 0.5 + \frac{\sin^2\left(x_1^2 - x_2^2\right) - 0.5}{\left(1 + 0.001\left(x_1^2 - x_2^2\right)\right)^2}$ $-100 \le x_k \le 100, \quad k = 1,2$	0

4	Bohachevsky N.1 Benchmark Function	$f_4(x_1,x_2)=x_1^2+2x_2^2-0.3\cos(3.\pi.x_1)$ $-0.4\cos(4.\pi.x_2)+0.7$ $-100\le x_k\le 100,\quad k=1,2$	0
5	Ackley Benchmark Function	$f_5(x)=-\alpha\exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d}x_i^2}\right)$ $-\exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right)+\alpha+e$ $-32.768\le x_k\le 32.768$	0
6	Levy Benchmark Function	$f_6(x)=\sin^2(\pi\omega_1)$ $+\sum_{i=1}^{d-1}(\omega_i-1)\left[1+10\sin^2(\pi\omega_i+1)\right]$ $+(\omega_d-1)^2\left[1+\sin^2(2\pi\omega_d)\right]$ $\omega_i=1+\frac{x_i-1}{4}$ $-10\le x_k\le 10$	0

III. Test Results

The comparative performance of the proposed Animal Adaptability Algorithm (AdA) and ten well-established metaheuristics on four standard benchmark functions (Rastrigin, Griewangk, Schaffer N.2, and Bohachevsky N.1) is reported in Table 3. Each experiment was performed over 100 independent runs, and the minimum (Min), mean (Mean), and standard deviation (Std) of the obtained solutions were calculated to ensure statistical robustness and to assess the algorithm's accuracy and stability.

A detailed analysis of the results reveals that AdA consistently achieved the global optimum across all benchmark functions, with both the minimum and mean values approaching the optimal solution. Moreover, the standard deviation of AdA remained significantly lower than that of most competing algorithms, indicating not only high accuracy but also strong stability and repeatability of the results. For instance, in the Rastrigin and Griewangk test functions (f1 and f2), AdA outperformed algorithms such as BA, BFO, and HS by several orders of magnitude in terms of solution accuracy and robustness. Another notable feature of AdA is its computational efficiency. The average execution time of AdA over 100 runs ranged from 6.4 to 8.7 seconds, which is substantially faster than algorithms such as HS (over 300 seconds), SA (about 150–200 seconds), and even classical methods like PSO (20–40 seconds). This fast convergence highlights the algorithm's effective balance between exploration and exploitation. When compared to strong competitors such as CS, BBO, and GSA, AdA demonstrated comparable or superior accuracy but achieved convergence in fewer iterations and with much lower computational cost. While algorithms like CS and BBO also reached optimal solutions, they required significantly more time, which limits their practicality for real-world large-scale problems.

To further assess the scalability and robustness of the proposed AdA algorithm, additional experiments were performed on four benchmark functions: Rastrigin (f1), Griewangk (f2), Ackley (f5), and Levy (f6), with 10, 50, 100, and 500 decision variables. Each configuration was executed 100 times independently, and the minimum (Min), mean (Mean), standard deviation (Std), and execution time were recorded, as summarized in Table 4. The results clearly demonstrate that AdA consistently achieves near-optimal or optimal solutions across all tested functions and dimensions. Even as the dimensionality increases, the algorithm maintains low variation among runs, indicating strong stability and repeatability. For instance, in the 500-dimensional Rastrigin and Griewangk test function, AdA successfully converged to high-quality solutions without a significant increase in computational error, while maintaining reasonable execution times.

In the continuation, the performance of the proposed Adaptive Algorithm (AdA) in optimizing energy consumption in air conditioning systems is evaluated. To assess the effectiveness of this algorithm, a comparison is made with other metaheuristic algorithms, including PSO, GA, and BBO. The simulation results, presented in Figure 3, demonstrate the superiority of the AdA algorithm in terms of convergence speed and accuracy in reaching the optimal solution.

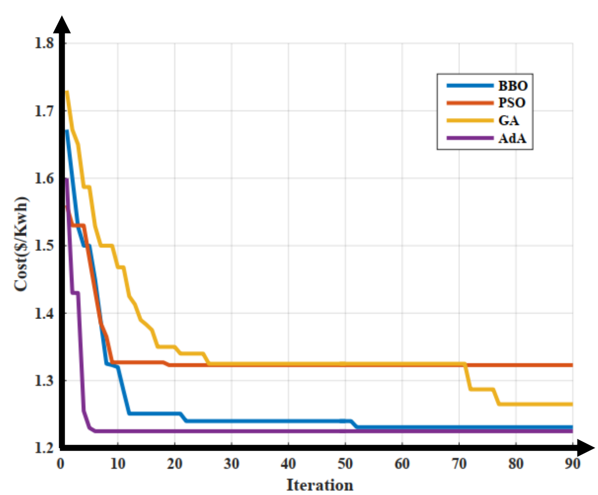

Figure3: Rate of convergence of algorithms

Table 3: Simulation results

Function	Statistic	Algorithms					
		ABC	CS	BA	BBO	CA	GA
f1	Min	3.43E-08	0	3.99E-09	0	0	9.63E-05
	Mean	6.19E-03	5.54E-15	1.47E-05	4.57E-06	1.79E-01	7.39E-01
	Std	1.53E-02	1.21E-14	1.77E-05	2.29E-05	4.09E-01	7.32E-01
	Time(s)	108.3	104.8	123	41	36.8	15.9
f2	Min	1.37E-05	3.33E-15	1.38E-10	0	0	1.18E-05
	Mean	3.63E-03	2.24E-04	5.61E-08	3.25E-03	1.31E-02	5.10E-03
	Std	3.10E-03	7.69E-04	5.90E-08	3.69E-03	1.15E-02	4.10E-03
	Time(s)	111.9	140.6	130.8	44.8	39.3	21
f3	Min	2.62E-08	9.33E-15	1.40E-14	0	0	1.56E-07
	Mean	7.22E-04	1.81E-05	2.12E-07	4.01E-04	3.35E-06	4.70E-03
	Std	1.30E-03	4.48E-05	1.57E-06	8.79E-04	3.35E-05	4.80E-03
	Time(s)	102.8	119.3	97.5	38.3	35.5	13.3
f4	Min	0	0	3.59E-09	0	0	1.32E-06
	Mean	1.77E-07	5.03E-15	1.91E-06	5.42E-08	1.31E-14	8.58E-02
	Std	2.28E-06	2.99E-15	2.25E-06	2.91E-07	8.65E-14	0.1436
	Time(s)	97.35	64.8	99.9	38.1	35.6	13.3

Table 3 (continued): Simulation results

Function	Statistic	Algorithms				
		GSA	HS	PSO	SA	AdA
f1	Min	0	8.42E-08	0	4.19E-05	0
	Mean	1.41E-01	4.62E-05	4.30E-11	5.32E-03	0
	Std	3.17E-01	7.37E-05	4.30E-10	7.65E-03	0
	Time(s)	102.4	345.8	39.6	208	6.4

f2	Min	0	4.44E-10	0	1.12E-08	0
	Mean	9.04E-03	4.22E-03	4.01E-03	1.25E-05	0
	Std	6.22E-03	3.68E-03	3.85E-03	1.82E-05	0
	Time(s)	107.4	360.8	41.7	192.6	8.7
f3	Min	0	1.11E-12	0	2.96E-10	0
	Mean	4.61E-05	1.02E-03	5.97E-08	7.71E-08	0
	Std	2.89E-04	1.71E-03	5.69E-07	1.65E-07	0
	Time(s)	73.7	341.2	20.2	101.9	6.7
f4	Min	0	4.53E-08	0	1.71E-06	0
	Mean	2.89E-03	5.14E-06	4.16E-07	9.50E-04	0
	Std	2.89E-02	1.12E-05	4.16E-06	2.49E-03	0
	Time(s)	73.7	337.3	23.4	159.8	6.7

Table 4: AdA performance for 10–500 variables (100 runs)

Function	Statistic	Dimension (D)			
		10	50	100	500
F1	Min	0	0	0	0
	Mean	0	0	0	0
	Std	0	0	0	0
	Time(s)	11.1	14.5	17.58	112.4
F2	Min	0	0	0	0
	Mean	0	0	0	0
	Std	0	0	0	0
	Time(s)	12.6	19.1	30	199.5
F5	Min	8.8E-16	8.8E-16	8.8E-16	8.8E-16
	Mean	8.8E-16	8.8E-16	8.8E-16	8.8E-16
	Std	8.8E-16	8.8E-16	8.8E-16	8.8E-16
	Time(s)	12.9	17.1	23.1	145
F6	Min	1.9E-4	1.8E-3	2.1E-3	2.1E-2
	Mean	8.3E-3	2.7	5.46	33.9
	Std	7E-3	1.6	3.36	14.2
	Time(s)	27.2	50.3	104.1	618

IV. Limitations and Future Work

AdA demonstrates robust and efficient performance across the tested benchmark functions. However, certain aspects present limitations, which also highlight some of the algorithm's strengths.

I. Parameter Adaptability: AdA requires tuning of certain control parameters; however, this flexibility allows the algorithm to be effectively adapted to a wide range of optimization problems.

II. Computational Effort: The computational cost may increase for very large-scale problems, yet this is due to its thorough exploration and exploitation mechanisms, which enhance solution quality.

III. No Guarantee of Global Optimum: Like most metaheuristics, AdA does not guarantee finding the global optimum for all problem types. Nevertheless, it consistently achieves high-quality solutions across diverse benchmarks, demonstrating its robustness and reliability.

For future studies, extending AdA to handle multi-objective and dynamic optimization problems can broaden its applicability. Furthermore, developing hybrid versions that integrate AdA with classical optimization methods or machine learning techniques may further enhance its performance. In addition, applying AdA to real-world engineering problems such as energy systems, scheduling, and network design can provide deeper insights into its practical effectiveness.

V. Conclusions

In recent decades, optimization problems in various fields such as engineering have become increasingly complex. Meta-heuristic algorithms, particularly those inspired by nature and swarm intelligence, have emerged as effective approaches to tackle such problems. Their flexibility allows them to handle complex scenarios and provide high-quality solutions. In this study, a new meta-heuristic algorithm based on animals' adaptability was proposed. The algorithm consists of three stages: decision making, repelling or attraction, and reproduction. In the first stage, the animals' decision-making process in different situations was modeled using the interval halving method. In the second stage, competition among members to define territories was represented, followed by reproduction in the final stage.

The proposed algorithm was first tested on small-scale benchmark problems, where it outperformed other algorithms such as BBO, GA, PSO, and CA in terms of convergence speed and solution stability. To further evaluate its robustness, the number of decision variables was increased up to 500, and the algorithm continued to provide reliable and stable results. Furthermore, the algorithm was applied to a practical problem of air conditioner load scheduling, where it again demonstrated superior performance compared to the other methods in terms of solution quality and stability.

The algorithm was implemented in MATLAB and compared with other well-known methods. The results demonstrated that the proposed approach is capable of reaching the global optimum with faster convergence and reduced computational time compared to existing algorithms. Overall, these findings highlight the flexibility, robustness, and effectiveness of the proposed method for both benchmark and real-world optimization problems.

REFERENCES

[1] H. M. CheshmehBeigi and A. Mohamadi, "Torque ripple minimization in SRM based on advanced torque sharing function modified by genetic algorithm combined with fuzzy PSO," International Journal of Industrial Electronics Control and Optimization, vol. 1, no. 1, pp. 71–80, Jun. 2018, doi: 10.22111/ieco.2018.24302.1016.

[2] D. P. Toan and T. V. Van, "Improving the genetic algorithm in fuzzy cluster analysis for numerical data and its applications," Iranian Journal of Fuzzy Systems, vol. 20, no. 5, pp. 171–187, 2023, doi: 10.22111/ijfs.2023.7834.

[3] Y. Yu, J. Su, and B. Wu, "A hybrid Bayesian model updating and non-dominated sorting genetic algorithm framework for intelligent mix design of steel fiber reinforced concrete," Engineering Applications of Artificial Intelligence, vol. 161, Part A, Dec. 2025, Art. no. 112071, doi: 10.1016/j.engappai.2025.112071.

[4] X. Bai et al., "Efficient Hybrid Multi-Population Genetic Algorithm for Multi-UAV Task Assignment in Consumer Electronics Applications," IEEE Transactions on Consumer Electronics, vol. 71, no. 2, pp. 2395–2406, May 2025, doi: 10.1109/TCE.2025.3563339.

[5] M. Morin, I. Abi-Zeid, and C.-G. Quimper, "Ant colony optimization for path planning in search and rescue operations," European Journal of Operational Research, vol. 305, no. 1, pp. 53–63, Feb. 2023, doi: 10.1016/j.ejor.2022.06.019.

[6] Y. Hou, X. Guo, H. Han, J. Wang, and Y. Du, "Adaptive ant colony optimization algorithm based on real-time logistics features for instant delivery," IEEE Transactions on Cybernetics, vol. 54, no. 11, pp. 6358-6370, 2024. doi: 10.1109/TCYB.2024.3454346.

[7] J. Farzaneh, R. Keypour, and A. Karsaz, "A novel fast maximum power point tracking for a PV system using hybrid PSO-ANFIS algorithm under partial shading conditions," International Journal of Industrial Electronics Control and Optimization, vol. 2, no. 1, pp. 47–58, Jan. 2019, doi: 10.22111/ieco.2018.25721.1056.

[8] J. Águila-León et al., "Optimizing photovoltaic systems: A meta-optimization approach with GWO-Enhanced PSO algorithm for improving MPPT controllers," Renewable Energy, vol. 230, Sep. 2024, Art. no. 120892, doi: 10.1016/j.renene.2024.120892.

[9] T. Lan et al., "A robust method of dual adaptive prediction for ship fuel consumption based on polymorphic particle swarm algorithm driven," Applied Energy, vol. 379, Feb. 2025, Art. no. 124911, doi: 10.1016/j.apenergy.2024.124911.

[10] Y. Sun, Y. Wu, and Z. Liu, "An improved differential evolution with adaptive population allocation and mutation selection," Expert Systems with Applications, vol. 258, Dec. 2024, Art. no. 125130, doi: 10.1016/j.eswa.2024.125130.

[11] X. Zhou, G. Tan, H. Wang, Y. Ma, and S. Wu, "Artificial bee colony algorithm based on multi-neighbor guidance," Expert Systems with Applications, vol. 259, Jan. 2025, Art. no. 125283, doi: 10.1016/j.eswa.2024.125283.

[12] X. Kong et al., "Integrating cumulative binomial probability into artificial bee colony algorithm for global optimization in mechanical engineering design," Engineering Applications of Artificial Intelligence, vol. 151, Jul. 2025, Art. no. 110628, doi: 10.1016/j.engappai.2025.110628.

[13] T. Zhang et al., "A Firefly Algorithm-Based Spectral Fitting Technique for Wavelength Modulation Spectroscopy Systems," IEEE Sensors Journal, vol. 24, no. 1, pp. 478–485, Jan. 2024, doi: 10.1109/JSEN.2023.3331711.

[14] X. Yu and W. Luo, "Reinforcement learning-based multi-strategy cuckoo search algorithm for 3D UAV path planning," Expert Systems with Applications, vol. 223, Aug. 2023, Art. no. 119910, doi: 10.1016/j.eswa.2023.119910.

[15] H. Singh, D. Witarsyah, A. Misra, and R. G. Tiwari, "Bat Algorithm: A Review on Theory, Modifications and Applications," in 2023 3rd International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA), Denpasar, Bali, Indonesia, 2023, pp. 78–83, doi: 10.1109/ICICyTA60173.2023.10428705.

[16] A. S. K. Kannan, S. A. Balamurugan, and S. Sasikala, "A Novel Software Package Selection Method Using Teaching–Learning Based Optimization and Multiple Criteria Decision Making," IEEE Transactions on Engineering Management, vol. 68, no. 4, pp. 941–954, Aug. 2021, doi: 10.1109/TEM.2019.2918050.

[17] J. He et al., "Three-Dimensional Image Reconstruction of Breast Tumor by Electrical Impedance Tomography Based on Dimensional Grey Wolf Optimization Algorithm," IEEE Transactions on Instrumentation and Measurement,

vol. 74, pp. 1–10, 2025, Art. no. 4504310, doi: 10.1109/TIM.2025.3547095.

[18] R. Pereira, H. Morais, and R. Castro, "Optimising Portugal's 2050 energy system: Electric vehicles and hydrogen integration using Grey Wolf Optimiser and EnergyPLAN," Energy, vol. 312, Dec. 2024, Art. no. 133372, doi: 10.1016/j.energy.2024.133372.

[19] S. Huang, H. Yao, T. Mai, D. Wu, J. Xu, and F. R. Yu, "Multi-Agent Moth-Flame Reinforcement Learning Based Broadcast Beam Optimization," IEEE Transactions on Mobile Computing, vol. 24, no. 8, pp. 7223–7236, Aug. 2025, doi: 10.1109/TMC.2025.3547946.

[20] L. Deng and S. Liu, "Deficiencies of the whale optimization algorithm and its validation method," Expert Systems with Applications, vol. 237, Part B, Mar. 2024, Art. no. 121544, doi: 10.1016/j.eswa.2023.121544.

[21] J. Wang et al., "Method for designing A-weighting filter in sound level meter based on improved whale optimization algorithm," Measurement, vol. 257, Part B, Jan. 2026, Art. no. 118720, doi: 10.1016/j.measurement.2025.118720.

Vahid Shirzad Iraj received his B.Sc. degree in Electrical Power Engineering from Bojnourd Branch, Islamic Azad University, Bojnourd, Iran, in 2008, and the M.Sc. degree in Electrical Engineering from Semnan University, Semnan, Iran, in 2011. He is currently pursuing the Ph.D. degree in Electrical Power Engineering at Semnan University, Semnan, Iran. His research interests include smart grids, demand side management, stochastic optimization algorithms, and renewable energy.

Reza Keypour received his B.Sc. degree in electrical engineering from Shahid Beheshti University in 1997 and M.Sc. and Ph.D degrees from Tarbiat modares University, Tehran, Iran in 2000 and 2006, respectively. He is currently an Associate Professor with the Faculty of Electrical and Computer Engineering, Semnan University, Semnan, Iran. His research interests include power system operation and planning, smart grids, microgrids and renewable energy.