Available online at http://scik.org

Commun. Math. Biol. Neurosci. 2021, 2021:47

https://doi.org/10.28919/cmbn/5767

ISSN: 2052-2541

DRAGONFLY ALGORITHM IN 2020: REVIEW

DELIAN HENDARDI[1], WILLIAM FELIX JOSEPHEN[1], HARCO LESLIE HENDRIC SPITS WARNARS[2,*], EDI ABDURRACHMAN[2], PRIATI ASSIROJ[2], ACHMAD IMAM KISTIJANTORO[3], ANTOINE DOUCET[4]

[1]Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia

[2]Computer Science Department, BINUS Graduate Program – Doctor of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia

[3]School of Electronics Engineering and Informatics, Institut Teknologi Bandung, Bandung 40132, Indonesia

[4]Laboratoire L3i - Université de La Rochelle, Avenue Michel Crépeau, F-17 042 La Rochelle Cedex 1, France

Abstract: Swarm Intelligence is the meta-heuristic algorithm that is inspired by the natural behavior of some groups of animals (like dragonfly, ants, ducks, etc.) striving for their life existence. One of them is Dragonfly Algorithm. Dragonfly Algorithm has been used to solve real-world nonlinear problems in engineering. In this paper, we reviewed Dragonfly Algorithm implementation in 2020. We collected papers that related to Dragonfly Algorithm in 2020, then we discussed it based on the subtopic that we stated. From our observation, we got the insight that this algorithm was excellent to implement optimization in some networks or electricals problems. Mostly, to get the best result, Dragonfly Algorithm must be hybridized with the other algorithm.

Keywords: swarm intelligence; optimization; meta-heuristic algorithm; nonlinear problem; computational intelligence.

2010 AMS Subject Classification: 65Y04.

---

*Corresponding author

E-mail address: spits.hendric@binus.ac.id

Received March 25, 2021

1. INTRODUCTION

Swarm Intelligence is the meta-heuristic algorithm that is inspired by the natural behavior of some groups of animals (like dragonfly, ants, ducks, etc.) striving for their life existence[1]. Swarm Intelligence is used on Computational Intelligence for solving non-linear model problems with real-world implementation which use almost all areas of science, engineering, and industry like data mining, business planning, and so on. This algorithm was invented for the first time by Gerardo Berni and Jing Wang in 1989 for cellular robotic system development. There are various kinds of Swarm Intelligence that are used for optimization. There are real-ant adoption Ant-Colony Optimization (ACO), fish-colony-based Artificial Fish Swarm Optimization (AFSO), flashing behavior fireflies Firefly Algorithm (FA), and so on. In 2015, Seyedali Mirjalili invented a Swarm Intelligence model that is inspired by static and dynamic swarming of dragonfly called Dragonfly Algorithm (DA)[3].

In the Dragonfly Algorithm, two important phases of optimization, exploration, and utilization, are designed by modeling social interactions in navigating, looking for food, and fighting compiled enemies dynamically or actively clustered by considering binary and multi-objective DA called binary DA (BDA) and multi-objective DA (MODA). In the real world, the Dragonfly Algorithm can be implemented to help doctors diagnose using thermographic images to get a very reliable clinical decision, which supports helping doctors make a diagnosis using thermographic images.

Dragonfly Algorithm is just the same as the other Swarm Intelligence algorithms. Dragonfly Algorithm has two swarming behaviors, i.e. static swarming, the dragonflies form a crowded formation and then move locally and abruptly change the steps, and dynamic swarming, a big group of dragonflies flying in one direction in the long-range distance[4]. These static and dynamic swarmings are served as exploration and exploitation behavior of the Dragonfly Algorithm. Swarm movements of the dragonfly are divided into five segments, namely separation, alignment, cohesion, attraction towards a food source, and distraction towards enemy sources[4]. Currently, Dragonfly Algorithm has been used to solve real-world nonlinear problems in engineering. Three well-known engineering problems such as welded beam design, pressure vessel, and motor design benchmark study are using Dragonfly Algorithm, accurately Memory based Hybrid Dragonfly Algorithm (MHDA), to get the best performance of its solution. We think the Dragonfly Algorithm has some prominent features that will help engineers to get the best optimal solution to their problems.

Therefore, in this paper, we will review several journals related to the Dragonfly Algorithm, especially the usage of the Dragonfly Algorithm in 2020. We choose this topic because we want to know what problems can be solved with the Dragonfly Algorithm, especially in 2020, and how efficient this algorithm works in modern problems. We also want to find the specific advantages and disadvantages when using this algorithm. We will start to find some papers that are published in 2020 and use Dragonfly Algorithm as their model. Then, we will read these papers and analyze their efficiency in solving the writers' case. Afterward, we will summarize the paper so it can be read by readers of this literature review. And the last, we hope, with this paper, the researcher could conduct some other research that can give impacts to the field of engineering because we think this algorithm gives a better efficiency for some real-world problems, especially we need more energy savings in this modern era.

2. Material and Methods

In order to create this literature review, we will do some steps. First, we will collect papers that are related to the implementation of the Dragonfly Algorithm in 2020. Then, we will form a subtopic to make a study of this paper clearer. Moreover, we will conduct a conclusion from what we've observed. These methods will be explained below.

I. Paper Collection and Filtering

First of all, we collect some papers, notably academic papers. We get these papers from Google Scholar. Because we want to review Dragonfly Algorithm implementations in 2020, we limit our papers' published year to only 2020. We use only the word 'Dragonfly Algorithm' and 'Dragonfly' to find the papers and then filter them by year of publication. Then, we filter the papers that are relevant to our topic. From this step, we get 40 papers from Google Scholar with Q1 journal is dominated in this paper.

II. Classification

1) Subtopic Classification

Because there are so many papers that we got with the different subject material, we cannot review all of these papers. We will divide the papers into several topics to make them effective and easy to read. These topics are:

a. Dragonfly Algorithm

As a part of the Swarm Intelligence algorithm, the Dragonfly Algorithm has the same behavior as animals that always assemble when they have found some food and spread when they have found some threat. For this subtopic, we will briefly discuss Dragonfly Algorithm and how this algorithm works, including its pseudocode.

b. Application of Dragonfly Algorithm in the modern era

Dragonfly Algorithm was invented in 2015. As a result, the study of this algorithm is just a few. However, we want to know how this algorithm is applied in some study cases, especially in 2020.

c. Effectiveness of the Dragonfly Algorithm

Dragonfly Algorithm has better optimization than some swarm intelligence algorithms. Therefore, we want to explore what sectors can be optimized by the Dragonfly Algorithm. In this subtopic, we also discuss a study that gives Dragonfly Algorithm a not good result.

d. How Dragonfly Algorithm in the future

We want to analyze the impact of this algorithm in the future since it has some useful features that can make some cases more effective. In this subtopic, we analyze it based on what we have discussed in the prior section.

2) Paper Classification

To classify our paper, we will check categories for each paper. For categories for each paper, we will classify it based on international categories derived from Scimago. Papers that we will categorize are the papers that are published in 2020.

From what we've found, two papers cannot be classified in table 1. There are papers [35] and paper [41]. Paper[35] cannot be classified because its publisher is new in Scimago so there is no categorization yet. And for paper [41], it cannot be classified because its publisher does not exist in Scimago.

Paper type	References	Total
Q1	[17], [18], [22], [23], [25], [26], [27], [31], [34], [39], [42], [43], [44], [46], [47], [49], [50]	17
Q2	[11], [14], [20], [24], [30], [40], [45]	7
Q3	[12], [13], [15], [16], [19], [21], [29], [38], [48],	9
Q4	[28], [33]	2
Conference	[32], [36], [37],	3
Unknown	[35], [41]	2
Total		40

Table 1. Paper's classification

3. DISCUSSION

As the name stands, Dragonfly Algorithm is the algorithm that is inspired by the swarming behavior of dragonfly[5], one of the animals that are classified as an insect. Dragonfly Algorithm has two main activities, i.e. static swarming, analogous with dragonfly when they are hunting a portion of food and at the same time they form a small group, and dynamic swarming, dragonfly's behavior when they are going to migrate to another place by forming a large group[6]. Dragonfly Algorithm has N dragonflies that are always defined as search agents and do not be confused with dragonfly and search agent words. As the consequences of their natural activities, Dragonfly Algorithm has five mathematical equations that are influenced by them. These equations are[7]:

a) Separation

The purpose of separation is to avoid collision when they are in the same neighborhood. Given x_i as the current position of a particular object (in this case dragonfly) and x_j as other objects apart from x_i, the distance between them is formulated as:

$$S_i = -\sum_{j=1}^{N}(x_i - x_j) \qquad (1)$$

b) Alignment

Alignment is a habit from a search agent that adjusts its velocity to the other search agents in the same neighborhood. Given V_j as the velocity of its neighborhood and A_i as an alignment of a single search agent, Alignment can be formulated by:

$$A_i = \frac{1}{N} \cdot \sum_{j=1}^{N} V_j \qquad (2)$$

c) Cohesion

Cohesion is a habit of dragonfly that flies towards the center of search agents. Given x_i as the position of current individual and x_j as a position of its adjacent search agent, cohesion can be derived from the formula:

$$C_i = \frac{1}{N} \cdot \sum_{j=1}^{N} x_j - x_i \quad (3)$$

d) The attraction of a food source

The attraction of a food source is a response from a search agent when a dragonfly has found the food. Given x_i as the position of dragonfly and f as the location of the food. Tendency the search agent go towards the food as follows:

$$F_i = f - x_i \quad (4)$$

e) The distraction of an enemy

This response occurs when search agents have found some threatening object that surrounds them. Given x_i as the position of dragonfly and e as the location of the enemy. The calculation for this case as follows:

$$E_i = e + x_i \quad (5)$$

If the dragonfly has at least one neighbor, then its velocity and position must be updated [8]. The velocity can be updated using a formula that is similar to Particle Swarm Optimization (PSO) algorithm. For position, it can be updated using a formula:

$$x_{t+1} = x_t + dx_{t+1} \quad (6)$$

Where t noted as the current iteration, x_{t+1} noted as the next position, x_t as current position, and dx_{t+1} noted as a step vector that is formulated by:

$$s \cdot S_i + a \cdot A_i + c \cdot C_i + f \cdot F_i + e \cdot E_i + l \cdot dx_t \quad (7)$$

Where dx_t represents the current step vectors; s, a, c, f, and e represent the weight of separation, the weight of alignment, the weight of cohesion, food factor, and enemy factor; l noted as the inertia weight.

If in some cases the dragonfly doesn't have a neighbor, then the dragonfly must do some random movement. As a result, the position must be updated using Levy's flight, i.e. a random walk [9] formula. The formula as follows:

$$x_{t+1} = x_t + Levy \cdot x_t \quad (8)$$

```
1. Initialize dragonflies population (1..N)
2. Initialize step vector dx (1..N)
3. while not meet criteria do:
      Calculate the strength of N dragonflies
      Update food and enemy source
      Update l, c, a, s, f, and e
      Calculate S, A, C, F, E based on equations 1 to 5
      Set neighbor radius
      if has neighbor then
           Update velocity using Eq.7
           Update position using Eq.6
      else
           Update position using Eq.8
      end if
       Check new position based on fixed boundaries
4.     end while
```

Algorithm 1: Dragonfly Algorithm

The algorithm starts with the initialization of dragonflies (N) and their step vector. Food and enemy sources are also initialized in this step. After that, do iteration until maximum iteration or the desired result has been reached. When iteration occurs, calculate the objective of all dragonflies. This objective refers to what kind of problem that we want to solve. Then, update food and enemy sources, separation, cohesion, alignment, and inertia weight using the equation that has been stated above (Eqs. 1 to Eqs. 5). After that, check whether the dragonfly has a neighbor. If it is, then update position and velocity using Eq. 6. Else, just update the position. Below is the pseudocode of the Dragonfly Algorithm that is used to implement Extreme Learning Machine (ELM) for prediction [10].

1) Application of Dragonfly Algorithm in the modern era

Since 2015, Dragonfly Algorithm has been used in some applications of science. From paper [11], the Dragonfly Algorithm was used to reduce the load of energy consumption in the household. The study revealed that the Dragonfly Algorithm could decrease a load of energy consumption significantly. Dragonfly Algorithm also can reduce the cost of energy consumption by 35.51%. From the paper [12], the Dragonfly Algorithm aims to show that the DA cannot take out the indicator of the Single Diode Model and produce undesirable results. Improvement of the DA exploration ability will boost the efficiency and accuracy of the algorithm and at the same time decrease the error. From paper [13], the Dragonfly Algorithm was enhanced, named Improved Dragonfly Algorithm (IDA), and used in Flexible Flow-Shop Scheduling optimization. The result showed IDA had a better performance compared to Partial Swarming Optimization (PSO) and the original Dragonfly Algorithm (DA). From paper [14], the Dragonfly algorithm has the advantage of optimizing task scheduling and resource allocation in a cloud computing environment, and to evaluate the performance of the proposed method three problem instances are analyzed. Thus, the results of the simulation of the proposed model can be compared with existing techniques with specified execution time and execution costs. From paper [15], the Dragonfly algorithm is used for model transformation.

This method is used by combining it with the ACO algorithm and increasing the adaptive dragonfly method for transformation. By doing a comparison technique by illustrating that the ACADF value is better than what has existed. For more improvements in the transformation of the model, other optimization algorithms can be used. From paper [16], the Dragonfly algorithm is used to optimize the performance and stability of power systems and to overcome power quality problems, optimal location, and sizing of UPQC in power systems. In this paper [17], Binary Dragonfly Algorithm (BDA) was used to be a comparison with the Late Acceptance Hill Climbing (LAHC) algorithm using 18 University of California Irvine (UCI) datasets. The observation yielded the fact that BDA has near the same accuracy metric as LAHC, i.e. more than 90 percent. Another implementation of the Dragonfly Algorithm is written in paper [18]. The author used the Dragonfly Algorithm as a comparison with the Artificial Bee Colony (ABC) algorithm in signal reconstruction optimization based on big data. Paper [19] presented Dragonfly Algorithm and Ad Hoc On-Demand Distance Vector (Dragon-AODV) that is applied to avoid the black hole attack, i.e. malicious security attack in ad hoc networks. From paper [20], Dragonfly Algorithm is applied to make placing Virtual Machines for customers more efficient in cloud computing implementation.

From their experiment, we get the result that Dragonfly Algorithm was less waste of resources compared to four models that related to the Swarm Intelligence algorithm. Ultra-Reliable Low Latency Communication and Dragonfly Algorithm (URLCC-DF) on paper[21] are applied to the industrial wireless network. This paper stated that URLCC-DF has a better way in the implementation of the industrial wireless network. This paper [22] Dragonfly Algorithm is used as a hyperparameter for Support Vector Machine for regression (SVMr) to estimate a notable amount of the Infinite Dilution Activity Coefficient (IDAC). This combination produced a better model compared to each algorithm if they work independently. SVMr with DA hyperparameter got 99.6% accuracy and 1% of Root Mean Squared Error (RMSE). In this paper [23], the Dragonfly Algorithm is improved using various transfer functions (TF) to increase vector steps from continuous to binary space to increase its effectiveness. And the Dragonfly algorithm in this paper is used to change variables that change the search space to find an optimization solution [24].

In 2020, there are also new hybridizations of the Dragonfly Algorithm, namely the Non-Dominated Sorting Dragonfly Algorithm (NSDA)[25]. NSDA has a search mechanism that is very similar to Dragonfly Algorithm where the derived solutions are improved using step vectors. The result of this experiment showed that NSDA has its effectiveness both for qualitative and quantitative perspectives. NSDA algorithm got a good result both in convergence and coverage that the only Dragonfly Algorithm cannot reach it. One thing that must be concerned is NSDA can diminish its performance because of the complex task. Therefore, further research for the NSDA algorithm in complex tasks is needed. Another new hybridization DA is Dragonfly-based Joint Delay/Energy (DJDE)[26]. The purpose of DJDE usage is to attain the maximum spectral efficiency in managing radio resources in Long Term Evolution (LTE) networks. DJDE accomplished this task in three steps, there are record the scheduling parameters, optimize weighted effect regulator (α) that can be used to maximizes the fitness function so it can calculate the utility function to rank the User Equipment (UEs) in non-ascending order, and the scheduler allocates the availability of Resource Blocks (RBs) according to the ranked list. Although DJDE performance is not as good as Proportional Fair (PF), DJDE outperforms PF in terms of energy consumption. The Dragonfly algorithm in this paper [27] is used to optimize the exploitation of the dragonfly herd's basic behavior and is used to ensure the dragonfly's flying speed consistently or not. This algorithm is hybridized with Wavelet Packet Decomposition (WPD) and Nonlinear Autoregressive (NAR) become WPD-DA-NAR. Hybridization between Dragonfly Algorithm and Modified Conjugate Gradient (MCG) is presented to be compared with the original Dragonfly Algorithm [28]. This paper obtained the result that DA-MCG has better performance than its original DA. However, the author did not state what metric that was used to measure the performance between them. Dragonfly Algorithm in paper [29] is used to minimize energy consumption in implementing Vehicular Ad Hoc Network (VANET), the system that connects between road client and road system. In this observation, the author used Dragonfly Algorithm and Enhanced Dragonfly Algorithm (EDA) as its model for optimizing K-Medoid clustering. The result of this experiment is that K-Medoid clustering with EDA is more efficient than K-Medoid with the original Dragonfly Algorithm with some metrics that are used by the author, i.e. Throughput (accomplishment send-receive data), Packet to the Delivery Ratio (the number of the delivered package), and Energy Consumption (energy usage).

Dragonfly Algorithm for solving VANETs' problem was also used in the paper [30]. The problem in this paper is that sometimes VANETs is always weakened by Distributed Denial of Service (DDoS). This study was using Dragonfly Algorithm, hybridized with Particle Swarm Optimization plus enhanced with Chaos theory formed Hybrid Chaos Particle Dragonfly Swarm (HCPDS) optimization to minimize the occurrence of DDoS attack. The author used various indicators to demonstrate DDoS attacks, such as network accuracy, false alarm detection, and communication overhead, and processing delay. This experiment concluded that HCPDS is superior compared to four traditional approaches, i.e. TRI, T-CLAIDS, CFV, and GF. Hybridized Dragonfly Algorithm with Gaussian mixture is used in paper [31] to make intersection traffic control. With this combination, it had better optimization than the Webster model. Paper [32] studied about Dragonfly Algorithm as the utility with a combination of Quasi-Oppositional (QODA) to tune Proportional-Integral-Derivative Filter (PIDF) and Capacitive Energy Storage System (CESS) with the purpose to solve the instability problem of an isolated hybrid energy distributed power systems (IHEDPS). The conclusion of this study is the fact that QODA gave better convergence and minimum error compared to each of Quasi-

Oppositional and Dragonfly Algorithm when they work independently. Paper [33] also discussed Dragonfly Algorithm. This paper studied about Dragonfly Algorithm as the algorithm to optimize placement and sizing of multiple renewable distributed generation units. In this study, the author tested its efficiency using IEEE 33-bus and IEEE 69-bus radial distribution test networks. This observation gave the result of Dragonfly Algorithm performance which can find the optimal location and sizing of distributed generation properly. Dragonfly Algorithm was also excellent compared with some of the methods that were proposed in this paper.

2) Effectiveness of the Dragonfly Algorithm

As stated in Mirjalili's paper, Dragonfly Algorithm has a better optimization data exploration due to its swarming exploration ability. However, some papers proved that this algorithm is as effective as Mirjalili's paper stated. From paper [34] about wind power forecasting, Improved Dragonfly Algorithm (IDA), combined with Support Vector Machine (SVM), gave the smallest error for all tested models, i.e. Dragonfly Algorithm and Support Vector Machine (DA-SVM), Genetic Algorithm and Support Vector Machine (GA-SVM), Grid and Support Vector Machine (Grid-SVM), Backpropagation Neural Network (BPNN), and Gaussian Process Regression (GPR). This paper also stated that IDA had the highest R-Squared value for both winter and autumn seasons. This can happen because Dragonfly Algorithm has abilities to optimize itself using adaptive learning factors and differential evolution strategy. Although this paper talked about short-term wind power forecasting, it proved that Dragonfly Algorithm has better accuracy results than the proposed algorithms.

In paper [35], Dragonfly Algorithm, precisely Binary Dragonfly Algorithm (BDA), is used to optimize the location of Phasor Measurement Units (PMUs) installation. The author concluded that BDA can be used to solve the Optimal PMUs Placement (OPP) problem. The model results in better optimization compared to some models that are used by the author to compare its performance. This paper [36] also proved that Dragonfly Algorithm has better performance. In this journal, the author used the Multi-Objective Dragonfly Algorithm (MODA) to optimize the design of analog circuits. Compared to the Multi-Objective Particle Swarming Optimization (MPSO) algorithm, MODA has better results than MOPSO in terms of some analog circuits optimization. From conference [37], Dragonfly Algorithm is combined with Genetic Algorithm and Elman Neural Network to evaluate urban air quality. With this combination, the algorithm got the best performance compared to proposed models with 95% accuracy and optimal error-control. From paper [38], Dragonfly Algorithm is applied to optimizing the allocation of D-STATCOM, controller of energy consumption. This paper used Dragonfly Algorithm to find optimization of D-STATCOM allocation. From this experiment, Dragonfly Algorithm, Particle Swarm Optimization, and Genetic Algorithm are used to find the best allocation for D-STATCOM, and their performance is also measured using cost function and convergence. The experiment is conducted by quantifying the performance of three algorithms and using hours as their parameter. The result of this experiment is Dragonfly Algorithm's convergence is faster than these two algorithms. However, this algorithm has the lowest cost function compared to the other two algorithms.

Another advantageous example implementation of the Dragonfly Algorithm is in paper [39]. In this paper, the Dragonfly Algorithm is applied, hybridized with Quantum Computing Mechanism (QCM), turned into QCM-DA, to enrich the solution searching behaviors. From this research, the author hybridized again with Support Vector Regression (SVR) and then combined it with Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) to reduce noise and Intrinsic Mode Function (IMF) to optimize the forecasting model. In this observation, CEEMDA-QCM-DA has a better forecast compared to the proposed models. It is proven by the Mean Absolute Error (MAE), Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), and Root Mean Squared Error (RMSE) has the smallest value compared to proposed models. On another paper [40], Dragonfly Algorithm is blended with some Swarm Intelligence algorithms to measure performance enhancement among Swarm Intelligence algorithms and then be compared with Fuzzy C-means (FCM).

In paper [41], Dragonfly Algorithm is used to make the cluster energy in Radio Frequency Identification (RFID) more efficient than the current model. In this research, the author found that the Dragonfly Algorithm has better performance in energy consumption, packet loss, generated packet, received packet, packet delivery ratio, and throughput network. In this paper [42], Dragonfly Algorithm aims to promote computational time quickly in finding suitable high-quality solutions and selecting the most suitable set of parameters combined with the scheduling of creation in a task scheduling system for Unmanned Aerial Vehicle (UAV). This paper concluded that the proposed method which included Dragonfly Algorithm resulted in better performance than Particle Swarm Optimization (PSO). This paper [43] presented an Improved Dragonfly Algorithm (IDA) to be combined with Blowfish Algorithm to get better accuracy in the implementation of a cloud environment. From that paper, we get information, i.e. IDA-BA, in this paper named Enhanced Blowfish Algorithm, was more secure than the proposed algorithm in this paper although it has longer encryption and decryption. Another implementation of IDA is in paper [44]. This paper used an Improved Dragonfly Algorithm that is combined with higher exploitation (IEDA) to minimize Harmonic Pollution (HP) for find optimization of Hybrid Power Active Filter (HPAF), technology that is used to cut off the harmonic of a power system. In this experiment, the author gives an extension that makes IDA overcome its weakness, specifically always trapped in local optima, by extending its range of exploitation and exploration. This can be done by the author by divide the particles into some exploitation and exploration population. This paper also used Differential Evolution (DE) as a mechanism of exchange information to utilize the quality of exploitation. IEDA also was compared with the other algorithms such as Dragonfly Algorithm, Particle Swarm Optimization, Improved Dragonfly Algorithm, Differential Evolution, Moth-Flame Optimization (MFO), two-evolution learning MFO (DELMFO), an approach based on DE (L-SHADE), and IEDA. The dataset that was used in this experiment is the list of HPFA configurations. This study gave the result that IEDA has the lowest Harmonic Pollution and average occasion of HP on HPFA config 1. However, only Particle Swarm Optimization has the highest Harmonic Pollution.

Although there are some positive results about Dragonfly Algorithm, there is also a case that is not enough to use Dragonfly Algorithm. In another paper [45], Dragonfly Algorithm is used to find the near-to-optimal solution for dynamic scheduling of tasks in cloud computing. In this paper, the author used Dragonfly Algorithm combined with biogeography-based optimization and Mexican hat wavelet (BMDA) to find the optimal solution, near precisely, to get dynamic scheduling tasks. This was done because, in this research, Dragonfly Algorithm has a problem when dealing with high-dimensional data, i.e. bad exploitation to find global maximum/minimum. The author stated that the performance of the Dragonfly Algorithm for dynamic scheduling of tasks can be increased by adding a mutation. With this combination, it is found that BMDA gives an excellent result compared to baseline algorithms (Particle Swarm Optimization, Bat algorithm, etc.) Another example that Dragonfly Algorithm is not as superior as the previous examples is in paper [46]. In this research, the author used Dragonfly Algorithm to compare it with the other Swarm Intelligence algorithms, especially to compare them with Cat Swarm Optimization (CSO). In this research, the authors found that the Dragonfly Algorithm performance, overall, has a bad performance compared to CSO, Butterfly Optimization Algorithm (BOA), and Fitness Dependent Optimizer (FDO). The benchmark used 23 traditional and 10 modern methods for conducting this experiment. In this paper [47], Dragonfly Algorithm is used as a model to be compared with hybrid Harris Hawkins Optimization and Support Vector Machine (HHO-SVM). In this paper, the author experimented with HHO-SVM with another proposed model, one of which was the Dragonfly Algorithm. This experiment generated the results, one of them was Dragonfly Algorithm had the worst fitness data with MonoAmine Oxidase (MAO) dataset when iteration getting bigger. However, in the Biodegradation dataset, this algorithm was not as bad as another algorithm.

Dragonfly Algorithm also cannot surpass Enhanced Grasshopper Optimization Algorithm (EGOA) in paper [48]. Dragonfly Algorithm has the worst convergence curve compared to EGOA and four other algorithms in that paper. In paper [44], Dragonfly Algorithm was used as a combination with Multilayer Perceptron (MLP) to predict monthly natural gas consumption in the United States (US). In this paper, the author combined MLP with not only Dragonfly Algorithm (DA-MLP), but also with some Swarm Intelligence algorithms, namely Invasive Weed Optimization (IWO-MLP), Evolution Strategy (ES-MLP), Genetic Algorithm (GA-MLP), and Imperialist Competitive Algorithm (ICA-MLP). The author used a gas dataset that consisted of 8 independent variables. From this experiment, ICA-MLP has the best model to predict monthly natural gas with the highest total score, i.e. 25, and R-Squared value, i.e. 0,9999. However, in this example, Dragonfly Algorithm is not as bad as explained in some papers. In this paper, DA-MLP has the second-best metric after ICA-MLP with 20 total scores and has the smallest RMSE value with a score of 0.000168. In paper [50], Dragonfly Algorithm was used to be one of the proposed models to compare it with a new optimization algorithm, namely Improved Fitness-Dependent Algorithm (IFDA). From this study, Dragonfly Algorithm, with the other 18 proposed algorithms, lost to IFDA that has the best performance measured by a metric that was used by the author.

From these examples, we can conclude that most implementations of the Dragonfly Algorithm are to combine it with other Swarm Intelligence algorithms or some Artificial Intelligence related algorithms. The paper [45] stated that premature convergence in the Dragonfly Algorithm always occurs. This happened because of the weakness of exploration in the Dragonfly Algorithm where it cannot exploit global maximum/minimum. Another consequence for the premature convergence is Dragonfly Algorithm is not properly working for the complex or higher dataset.

3) How Dragonfly Algorithm in the future

So far, from what we've discovered, we analyze that the Dragonfly Algorithm is often used for optimizing real-world problems that are not related to business, mostly used for science-related cases. And also, the Dragonfly Algorithm cannot be used standalone. In other words, the Dragonfly Algorithm must be combined with another algorithm to get the best results. This happened because of the premature convergence where this algorithm always traps in local optima. However, the result from this algorithm combination always gives a better model than the proposed algorithms in some papers. Sometimes Dragonfly Algorithm also has a bad result in some cases, especially with the new optimization algorithm. The most implementation of the Dragonfly Algorithm is to optimize some network and electrical problems. Maybe in the future, the Dragonfly Algorithm could be used more frequently to solve some real problems, which might be business problems. Dragonfly Algorithm perhaps would be hybridized again with some Swarm Intelligence or other algorithms to get better result and optimization in the future.

4. Discussion

In this paper, we discuss briefly the Swarm Intelligence algorithm that was created based on natural animal behavior. After that, we discuss the Dragonfly Algorithm, an algorithm that is created based on static and dynamic swarming of dragonflies, and briefly we also discuss its implementation in real life. After that, we explain about advantages and disadvantages of the Dragonfly Algorithm. Then, we stated that we want to review this algorithm in 2020. To do this, our steps are collecting the papers, setting the subtopics for this paper, and classifying all the papers. Then, we started our literature review by discussing the implementation of the Dragonfly Algorithm in 2020. In this section, we got that the Dragonfly Algorithm has been implemented in many real-world problems. After that, we analyze the effectiveness of this algorithm. We found that this algorithm cannot be standalone because of its premature convergence that led the model to be trapped in local optima. After that, the last section was discussing the Dragonfly Algorithm in the future. We stated that this algorithm could be implemented standalone. However, it might be only in some cases the Dragonfly Algorithm could be implemented independently. We hope, Dragonfly Algorithm perhaps would be hybridized again with some Swarm Intelligence or another algorithm to get better results and optimization in the future.

Acknowledgements

This work is supported by Research and Technology Transfer Office, Bina Nusantara University as a part of Bina Nusantara University's International Research Grant entitled MEMETIC ALGORITHM IN HIGH-PERFORMANCE COMPUTATION with contract number: No.026/VR.RTT/IV/2020 and contract date: 6 April 2020.

Conflict of Interests

The author(s) declare that there is no conflict of interests.

REFERENCES

[1] A. Chakraborty, A.K. Kar, Swarm Intelligence: A Review of Algorithms, in: S. Patnaik, X.-S. Yang, K. Nakamatsu (Eds.), Nature-Inspired Computing and Optimization, Springer International Publishing, Cham, 2017: pp. 475–494.

[2] M. Mavrovouniotis, C. Li, S. Yang, A survey of swarm intelligence for dynamic optimization: Algorithms and applications, Swarm Evol. Comput. 33 (2017), 1–17.

[3] S. Mirjalili, Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems, Neural Comput. Appl. 27 (2016), 1053–1073.

[4] S.R. K.S., S. Murugan, Memory based Hybrid Dragonfly Algorithm for numerical optimization problems, Expert Syst. Appl. 83 (2017), 63–78.

[5] G.I. Sayed, A. Tharwat, A.E. Hassanien, Chaotic dragonfly algorithm: an improved metaheuristic algorithm for feature selection, Appl. Intell. 49 (2019), 188–205.

[6] P.T. Daely, S.Y. Shin, Range based wireless node localization using Dragonfly Algorithm, in: 2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN), IEEE, Vienna, Austria, 2016: pp. 1012–1015.

[7] M.-A. Díaz-Cortés, N. Ortega-Sánchez, S. Hinojosa, et al. A multi-level thresholding method for breast thermograms analysis using Dragonfly algorithm, Infrared Phys. Technol. 93 (2018), 346–361.

[8] Z. Zhang, W.-C. Hong, Electric load forecasting by complete ensemble empirical mode decomposition adaptive noise and support vector regression with quantum-based dragonfly algorithm, Nonlinear Dyn. 98 (2019), 1107–1136.

[9] L. Xu, H. Jia, C. Lang, X. Peng, K. Sun, A novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution, IEEE Access. 7 (2019), 19502–19538.

[10] M.A. Salam, H.M. Zawbaa, E. Emary, K.K.A. Ghany, B. Parv, A hybrid dragonfly algorithm with extreme learning machine for prediction, in: 2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), IEEE, Sinaia, Romania, 2016: pp. 1–6.

[11] I. Hussain, M. Ullah, I. Ullah, A. Bibi, M. Naeem, M. Singh, D. Singh, Optimizing energy consumption in the home energy management system via a bio-inspired dragonfly algorithm and the genetic algorithm, Electronics. 9 (2020), 406.

[12] Z. Mat Isa, N. Mohd Nayan, N. Ashbahani Mohamad Kajaan, M. Hafiz Arshad, A Dragonfly Algorithm

Application: Optimizing Solar Cell Single Diode Model Parameters, J. Phys.: Conf. Ser. 1432 (2020), 012041.

[13] Z. Han, J. Zhang, S. Lin, C. Liu, Research on the Improved Dragonfly Algorithm-Based Flexible Flow-Shop Scheduling, in: R. Wang, Z. Chen, W. Zhang, Q. Zhu (Eds.), Proceedings of the 11th International Conference on Modelling, Identification and Control (ICMIC2019), Springer Singapore, Singapore, 2020: pp. 205–214.

[14] P. Neelima, A.R.M. Reddy, An efficient load balancing system using adaptive dragonfly algorithm in cloud computing, Cluster Comput. 23 (2020), 2891–2899.

[15] P.P. Jadhav, S.D. Joshi, ACADF: Ant Colony Unified with Adaptive Dragonfly Algorithm Enabled with Fitness Function for Model Transformation, in: A. Kumar, S. Mozar (Eds.), ICCCE 2019, Springer Singapore, Singapore, 2020: pp. 101–109.

[16] K. Gaddala, P. Sangameswara Raju, Optimal UPQC location in power distribution network via merging genetic and dragonfly algorithm, Evol. Intel. (2020). https://doi.org/10.1007/s12065-020-00364-1.

[17] B. Chatterjee, T. Bhattacharyya, K.K. Ghosh, P.K. Singh, Z.W. Geem, R. Sarkar, Late Acceptance Hill Climbing Based Social Ski Driver Algorithm for Feature Selection, IEEE Access. 8 (2020), 75393–75408.

[18] S. Aslan, D. Karaboga, A genetic Artificial Bee Colony algorithm for signal reconstruction based big data optimization, Appl. Soft Comput. 88 (2020), 106053.

[19] M. Goyal, S. Kumar, V.K. Sharma, D. Goyal, Modified Dragon-Aodv for Efficient Secure Routing, in: H. Sharma, K. Govindan, R.C. Poonia, S. Kumar, W.M. El-Medany (Eds.), Advances in Computing and Intelligent Systems, Springer Singapore, Singapore, 2020: pp. 539–546.

[20] A. Tripathi, I. Pathak, D.P. Vidyarthi, Modified Dragonfly Algorithm for Optimal Virtual Machine Placement in Cloud Computing, J. Netw. Syst. Manage. 28 (2020), 1316–1342.

[21] S. Bhardwaj, M.R. Ramli, D.-S. Kim, Leveraging Biological Dragonfly Scheme for URLLC in Industrial Wireless Network, in: S. Choudhury, R. Mishra, R.G. Mishra, A. Kumar (Eds.), Intelligent Communication, Control and Devices, Springer Singapore, Singapore, 2020: pp. 377–383.

[22] H. Benimam, C.S. Moussa, M. Hentabli, S. Hanini, M. Laidi, Dragonfly-Support Vector Machine for Regression Modeling of the Activity Coefficient at Infinite Dilution of Solutes in Imidazolium Ionic Liquids Using σ-Profile Descriptors, J. Chem. Eng. Data. 65 (2020), 3161–3172.

[23] S.M. Seyedpoor, M.H. Nopour, A two-step method for damage identification in moment frame connections using support vector machine and differential evolution algorithm, Appl. Soft Comput. 88 (2020), 106008.

[24] A. Sánchez-Chica, E. Zulueta, D. Teso-Fz-Betoño, P. Martínez-Filgueira, U. Fernandez-Gamiz, ANN-Based

Stop Criteria for a Genetic Algorithm Applied to Air Impingement Design, Energies. 13 (2019), 16.

[25] P. Jangir, MONSDA:-A Novel Multi-objective Non-Dominated Sorting Dragonfly Algorithm. Glob. J. Res. Eng.: F, Electric. Electron. Eng. 20 (2020), 28-52.

[26] H. Nashaat, O. Refaat, F.W. Zaki, I.E. Shaalan, Dragonfly-Based Joint Delay/Energy LTE Downlink Scheduling Algorithm, IEEE Access. 8 (2020), 35392–35402.

[27] H. Liu, H. Wu, Y. Li, Multi-step wind speed forecasting model based on wavelet matching analysis and hybrid optimization framework, Sustain. Energy Technol. Assess. 40 (2020), 100745.

[28] L.R. Khaleel, B.A. Mitras, A Novel Hybrid Dragonfly Algorithm with Modified Conjugate Gradient Method. Int. J. Computer Netw. Commun. Secur. 8 (2020), 40-48.

[29] M. Elhoseny, K. Shankar, Energy Efficient Optimal Routing for Communication in VANETs via Clustering Model, in: M. Elhoseny, A.E. Hassanien (Eds.), Emerging Technologies for Connected Internet of Vehicles and Intelligent Transportation System Networks, Springer International Publishing, Cham, 2020: pp. 1–14.

[30] S. Prabakeran, T. Sethukarasi, Optimal solution for malicious node detection and prevention using hybrid chaotic particle dragonfly swarm algorithm in VANETs, Wireless Netw. 26 (2020), 5897–5917.

[31] J. Mou, Intersection Traffic Control Based on Multi-Objective Optimization, IEEE Access. 8 (2020), 61615–61620.

[32] M. Agarwal, D. Guha, S. Purwar, Quasi-oppositional dragonfly algorithm: applied for frequency regulation of an isolated hybrid energy distributed power system, in: 2020 IEEE 9th Power India International Conference (PIICON), IEEE, SONEPAT, India, 2020: pp. 1–6.

[33] A. Boukaroura, L. Slimani, T. Bouktir, Optimal Placement and Sizing of Multiple Renewable Distributed Generation Units Considering Load Variations Via Dragonfly Optimization Algorithm. Iran. J. Electric. Electron. Eng. 16 (2020), 353-362.

[34] L.-L. Li, X. Zhao, M.-L. Tseng, R.R. Tan, Short-term wind power forecasting based on support vector machine with improved dragonfly algorithm, J. Clean. Product. 242 (2020), 118447.

[35] C.D. Patel, T.K. Tailor, S.S. Shah, S.H. Srivastava, Binary Dragonfly Algorithm Based Optimal PMU Placement Considering Contingency Constraints, Int. J. Adv. Res. Eng. Technol. 11 (2020), 34-43.

[36] M. Kotti, N. Dhahri, M. Fakhfakh, B. Benhala, MODA metaheuristic: evaluation and application to CCII sizing, in: 2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), IEEE, Meknes, Morocco, 2020: pp. 1–6.

[37] H. Liu, X. Geng, Application of GA-DA-Elman Neural Network Algorithm to Urban Air Quality Evaluation, IOP Conf. Ser.: Mater. Sci. Eng. 768 (2020), 052014.

[38] V. Tejaswini, D. Susitra, Dragonfly Algorithm for Optimal Allocation of D-STATCOM in Distribution Systems, in: S.S. Dash, C. Lakshmi, S. Das, B.K. Panigrahi (Eds.), Artificial Intelligence and Evolutionary Computations in Engineering Systems, Springer Singapore, Singapore, 2020: pp. 213–228.

[39] W.-C. Hong, Hybridizing QCM with Dragonfly Algorithm to Enrich the Solution Searching Behaviors, in: Hybrid Intelligent Technologies in Energy Demand Forecasting, Springer International Publishing, Cham, 2020: pp. 135–152.

[40] N. Bharanidharan, H. Rajaguru, Performance enhancement of swarm intelligence techniques in dementia classification using dragonfly - based hybrid algorithms, Int. J. Imaging Syst. Technol. 30 (2020), 57 - 74.

[41] P.S. Rathore, A. Kumar, V. García-Díaz, A Holistic Methodology for Improved RFID Network Lifetime by Advanced Cluster Head Selection using Dragonfly Algorithm, Int. J. Interact. Multimedia Artif. Intell. 6 (2020), 48-55.

[42] F. Sun, X. Wang, R. Zhang, Task scheduling system for UAV operations in agricultural plant protection environment, J. Ambient Intell. Human Comput. (2020). https://doi.org/10.1007/s12652-020-01969-1.

[43] V.K.R. Gangireddy, S. Kannan, K. Subburathinam, Implementation of enhanced blowfish algorithm in cloud environment, J. Ambient Intell. Human Comput. 12 (2021), 3999–4005.

[44] W. Dai, C. Li, Z. Cui, Y. Wu, L. Zhang, J. Huang, An improved dragonfly algorithm with higher exploitation capability to optimize the design of hybrid power active filter, IEEE Access. 8 (2020), 155020–155038.

[45] M.R. Shirani, F. Safi-Esfahani, Dynamic scheduling of tasks in cloud computing applying dragonfly algorithm, biogeography-based optimization algorithm and Mexican hat wavelet, J. Supercomput. 77 (2021), 1214–1272.

[46] A.M. Ahmed, T.A. Rashid, S.Ab.M. Saeed, Cat Swarm Optimization Algorithm: A Survey and Performance Evaluation, Comput. Intell. Neurosci. 2020 (2020), 4854895.

[47] E.H. Houssein, M.E. Hosney, D. Oliva, W.M. Mohamed, M. Hassaballah, A novel hybrid Harris hawks optimization and support vector machines for drug design and discovery, Computers Chem. Eng. 133 (2020), 106656.

[48] H. Feng, H. Ni, R. Zhao, X. Zhu, An Enhanced Grasshopper Optimization Algorithm to the Bin Packing Problem, J. Control Sci. Eng. 2020 (2020), 3894987.

[49] W. Qiao, H. Moayedi, L.K. Foong, Nature-inspired hybrid techniques of IWO, DA, ES, GA, and ICA, validated

through a k-fold validation process predicting monthly natural gas consumption, Energy Build. 217 (2020), 110023.

[50] D.A. Muhammed, S.A.M. Saeed, T.A. Rashid, Improved Fitness-Dependent Optimizer Algorithm, IEEE Access. 8 (2020), 19074–19088.