Journal of Information Hiding and Multimedia Signal Processing ©2020 ISSN 2073-4212
Ubiquitous International Volume **11**, Number **2**, June **2020**

Applying Adaptive and Self Assessment Fish Migration Optimization on Localization of Wireless Sensor Network on 3-D Terrain

Qing-Wei Chai, Shu-Chuan Chu, Jeng-Shyang Pan* and Wei-Min Zheng
College of Computer Science and Engineering
Shandong University of Science and Technology
Qingdao 266590, China
jengshyangpan@gmail.com

Received April 2020; revised June 2020

Abstract. *This paper presents an improved fish migration optimization (FMO), which adopts novel update equations of individuals and energy. A chieftain concept is introduced and it can attract individuals to exploitation around it. Therefore, the novel algorithm reduced the randomness and improved the convergence ability of the original algorithm. A more flexible update equation of energy is introduced which adjusts the amplitude of energy increase of individuals according to its fitness quality. The performance of the new algorithm is verified by CEC 2013 benchmark function. Besides, the novel algorithm is applied in solving the localization problem of Wireless Sensor Network (WSN) on 3-D terrain.*

Keywords: 3-D localization; DV-Hop; FMO; intelligence computing

1. Introduction.

Intelligence computing is an important branch of artificial intelligence, which is inspired by the swarm intelligence movement that existed in nature. For intelligent computing, the optimal problem is regarded as a black box, that is, the details of the calculation are ignored, and only the input is adjusted according to the output of the system to maximize or minimize its output. It can effectively solve various optimization problems without mathematical formulas. Therefore, it has been paid more and more attention from scholars in recent years and has made great progress.

There has been many excellent algorithms, such as, Artificial Bee Colony (ABC) [1, 2, 3], Genetic Algorithm (GA) [4, 5], Differential Evolution (DE) [6, 7, 8], Particle Swarm Optimization (PSO) [9, 10, 11, 12], Ant Colony Optimization (ACO) [13, 14], Cat Swarm Optimization (CSO) [15, 16, 17], QUATRE [18, 19, 20, 21], Black Hole [22, 23]. In addition, some methods are proposed to enhance the performance of algorithms, in order to reduce the memory cost, the probability formula is introduced to replace the movement of population [24, 25, 26, 27, 28]. The surrogate-assisted method is proposed by Sun et al., which can efficiently reduce the running time of algorithm [29]. The parallel method can enhance the global search ability by dividing the population into several groups and exchanging the information in groups [30, 31].

The maturing of microelectronics and wireless communication technologies has aroused more and more scholars' attention to WSN. These technologies cause the advance of multi-functional, smart, extremely small, and inexpensive sensor nodes that have the capabilities to communicate with each other through a wireless medium [32]. These tiny SNs comprise of transceiver, dedicated memory, processor, actuator, sensors, and power module [33]. Due to the limited size and energy of sensor nodes, sensor nodes cannot handle complex calculations and require efficient use of energy. In the real world, hundreds or thousands of nodes are randomly deployed in the target area and use wireless media to communicate with each other to form a WSN.

The accurate location of the sensor node contributes to monitoring the target area. Global positioning system (GPS) is generally used to identify the location of sensor nodes, but due to its cost and energy consumption, it is not practical to equip each sensor node with a GPS module. Some localization methods have been proposed in recent years, which only need a few sensor nodes to equip with GPS and can calculate the position of other sensor nodes in various ways. The nodes configuring the GPS module are called anchor nodes, and the other nodes are called unknown nodes. The existed localization methods can be divided into range-based localization methods and range-free localization methods. Range-based localization methods utilize the sensing device equipped in sensor nodes to obtain the actual range and angle information and further determine the distance between sensor nodes, such as received signal strength indicator (RSSI), Angle of Arrival (AOA) and Time of Flight (TOF) methods, etc. These schemes usually estimate the position of target node by the distance from three or more anchor nodes. Then multiliteration or triangulation techniques are used to estimate the position of target node position. Although these methods show good performance, additional hardware is required to determine the actual distance.

In contrast, range-free localization methods only need the connectivity information of WSN and can estimate the localization of target node, such as distance vector hop (DV-Hop). But the defect of these methods is obviously, it has low localization accuracy. As there no spare devices consume energy, these methods can effectively extend the life of WSN. More and more researcher pay attention to increase the accuracy [34, 35].

2. **Related Work.**

2.1. **Fish Migration Optimization.** Pan et al. proposed Fish Migration Optimization (FMO) in 2010, which simulated the predation and growth process of fish [36]. In nature, animals are often in dangerous situations, such as being caught by natural enemies or suffering from natural disasters. It is impossible for all individuals to grow into adults safely. The algorithm divides the life of the fish into five stages, and each stage has a survival rate to control the diversity of the algorithm. FMO is consisted of swim process and migration process.

In order to find more food, schools of fish usually move quickly over a wide area, this process is presented by swim process in FMO. When algorithm in swim process, the individual move to any direction to look for optimal value and it is good for avoiding local optimal value. The population update their position according to Eq. 1

$$P_d^{t+1} = P_d^t + \frac{E_d^r \cdot U_{s,d}^t}{a + b \cdot (U_{s,d}^t)^x} \tag{1}$$

$$E_d^r = E \cdot rand \tag{2}$$

$$U_{s,d}^t = P_d^t - P_d^{t-1} \tag{3}$$

Where the $rand$ is a random number between -1 and 1, the E is the energy and the U_s is the velocity of the fish. The more energy a fish has, the faster its speed. The a, b and x are three constant and they are 2.25, 36.2 and 2.23 respectively. As E_r can be positive or negative value, the individual can move toward to any direction. The most excellent individual of population is called as P_{best}, if the P_d^{t+1} has better fitness value than P_{best}, the P_{best} would be replaced by P_d^{t+1} and the velocity of it will be updated according to Eq.4. After moving, the energy is need to update according to Eq.5 and if energy more than up limit of energy, the fish will grow to next stage.

$$U_{s,d}^t = 2 \cdot U_{s,d}^t \tag{4}$$

$$eng = \sum_{d=1}^{D} E_d^r \tag{5}$$

During the migration process, some fish will be randomly selected and initialized. The fishes which belong to stage 1 and stage 2 have no probability to migrate as these fishes are immature. The probability of migration of other fish is different, and the more mature the fish, the greater the probability of migration. The migration probability of fishes in stage 3, stage 4 and stage 5 are 5%, 10% and 100% respectively. If the fish find better food source, in other words, the algorithm find better candidate, the velocity of this fish will be updated by the following equation, such that,

$$U_s = \pi \cdot U_s \tag{6}$$

2.2. **DV-Hop Localization Method of WSN.** DV-Hop is a range free localization method and it is proposed by Dragons Niculescu et al. [37, 38]. Many researchers have focused on it and improved its performance, but most articles have applied dv-hop to solve two-dimensional localization problems. This paper applies it to three-dimensional terrain.

In DV-Hop, the first work is each anchor node broadcasts a packet which consist of location of anchor node and hop count. The number of hops is initialized to zero and increased by one every transmit. For a receiving node, it records the minimal hop count between it and all anchor nodes. Then each anchor node obtains the location of other anchor nodes and the hop count between them. The distance of ever hop of an anchor node can be calculated by Eq. 7.

$$HopSize_i = \frac{\sum\limits_{j=1, j\neq i}^{n} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum\limits_{j=1, j\neq i}^{n} hop_{ij}} \tag{7}$$

Where the $HopSize_i$ is the average distance of each hop of i-th anchor node, the location of i-th anchor node is represented by (x_i, y_i) and the hop_{ij} is the hop count between i-th anchor node and j-th anchor node. The n is the number of anchor nodes, and the distance between any two nodes can be calculated by the corresponding hop size and hop count. After that, the distance information can be used to estimate the position of unknown node by trilateration survey.

In recent years, many scholars have proposed novel method to enhance the ability of DV-Hop, a novel method to calculate the hop size is presented by adopting the least square error criterion to revise the localization accuracy and the hop size can be calculated by Eq.8 [34].

$$HopSize_i = \frac{\sum\limits_{j=1, j\neq i}^{n} hop_{i,j} \cdot d_{ij}}{\sum\limits_{j=1, j\neq i}^{n} hop_{ij}^2} \tag{8}$$

Where the d_{ij} is the straight-line distance between i-th anchor node and j-th anchor node. The weighting hop size of unknown node to calculated the distance between anchor nodes and unknown nodes is proposed [35], which is presented in the following equations:

$$HopSize_{un} = \sum_{i=1}^{n} w_i \cdot HopSize_i \tag{9}$$

$$w_i = \frac{HopSize_i}{\sum_{m=1}^{n} HopSize_m} \tag{10}$$

3. **Adaptive and Self Assessment Migration Optimization.** From the above described about FMO, we can see the FMO algorithm spend the more time making explore, so it has a weaker exploitation ability. Although the original algorithm has strong performance in avoiding local optimal value, it has some difficulties in exploiting promising regions.

$$U_{s,d}^{t+1} = \frac{E_d^r \cdot U_{s,d}^t}{a + b \cdot (U_{s,d}^t)^x} \tag{11}$$

Besides, the Eq.11 can be obtained by combining the Eq.1 and Eq.3. As the E_d^r is a random number between -1 and 1, the value of a, b and x are 2.25, 36.2, and 2.23 respectively, so the denominator is much larger than the numerator in Eq.11. As the number of iterations increases, the speed of the fishes becomes slower and slower, the algorithm is at a standstill after dozens of iterations. This is bad for the algorithm solving optimization problems.

In the intelligent computing algorithm, the area around the optimal individual is considered to be the most promising area, which means that a better candidate solution can be found with a high probability. In order to enhance the exploitation ability of algorithms, the concept of best individual is introduced in the novel algorithm, which attracts other individuals exploiting around it. As "Ther is no free lunch" theorem saying [39], "any performance improvement for one type of problem will be offset by performance for another type of problem." If the algorithm obtain fast convergence rate, it would loss the stronger global search ability, and it is easily trapped in the local optimal value.

In order to overcome this problem, a self assessment method is proposed in this article. There are two case will be happen after a individual moving about fitness value, be better or worse. Therefore, a individual moves forward to a better individual or lefts a worse individual both can lead to the current individual become better. In the new algorithm, the population updates their position according to the following equations:

$$V_i^{t+1} = w \cdot (p_{best}^t - p_i^t) \cdot \frac{e_i^t}{Emax} + \frac{fv_i - fv_r}{|fv_i - fv_r|} \cdot c \cdot rand \cdot (p_r^t - p_i^t) \tag{12}$$

$$p_i^t + 1 = p_i^t + V_i^{t+1} \tag{13}$$

Where the w is a weight factor which decreases from 2 to 0.4 as the number of iterations increases. The algorithm with a lager w at an early stage is contributing to explore more areas and jump out local optimal value. On the contrary, the algorithm needs to exploit a promising area carefully. The e_i^t represents the energy of i-th individual at t iteration and the individual with more energy, the individual can move longer distance.$Emax$ is a constant value, if the energy of a individual exceed $Emax$, it would grow to the next stage and the energy initial to zero. The p_{best}^t, p_i^t, and p_r^t are positions of best individual, i-th individual and randomly selected individual at t iteration respectively. The $rand$ is a random number between 0 and 1, c is a constant value. In addition, the role of p_r^t can

be positive or negative. If the function value of p_r^t is greater than p_i^t, p_i^t will move to p_r^t, otherwise p_i^t will move away from p_r^t.
When an individual did not get a greater function value at the current iteration, the energy of this individual increases by the Eq.14, such that,

$$e_i^{t+1} = e_i^t + R_1 \cdot Emax \cdot \frac{fv_i - fv_{best}}{fv_{max} - fv_{best}} \tag{14}$$

To increase the random disturbance, a random number R_1 is introduced with a value between 2 and 12. The minimum function value is presented by fv_{best} and fv_{max} is the maximum function value. In the new algorithm, the worse the individual, the greater the energy. In the initialize of the algorithm, all fishes are set to stage 1, as the energy increase the fishes will grow to stage 2, stage 3, and stage 4. The fishes in stage 1 will not be initialized randomly, and the probability of the fish in stage 2, stage 3 and stage 4 is 0.15, 0.35, and 1 respectively.

4. **Applying Novel Algorithm to Locate the Unknown Nodes on 3-D terrain.**

In this paper, the sensor nodes are deployed on the 3-D terrain as shown in Figure 1 and the terrain is drowned by the "peak" function of Matlab. Sensor nodes communicate with each other on 3-D terrain, and signals are usually blocked by terrain obstacles, so it is an important task to detect whether there are obstacles between communication nodes. Suppose node P sends a message to node S, and there is a point M between them. If the height of M is higher than the ray height between P and S at the same position, M is regarded as a terrain obstacle between them. To visually show this situation, Figure 1 was introduced.
Signal transmission is affected by many factors, such as terrain, distance, and vegetation. This article uses a binary sensor model because it can simply and efficiently calculate whether nodes can be connected to each other. It is described by the Eq 15, such that,

$$C(r,s) = \begin{cases} 1, distance(r,s) \le R \; and \; there \; no \; obstacle \\ 0, distance(r,s) > R \; or \; there \; are \; obstacles \end{cases} \tag{15}$$

The intelligence computing has the advantage of solving optimization problems effectively and some work has applied it to locate the unknown nodes of WSN on 2-D surface [30]. In this article, the novel algorithm is applied to reduce the locate error based DV-Hop and the error can be calculated as following:

$$error = (\sum_{i=1}^{n} (\sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} - d_{ui}))^2 \tag{16}$$

$$Accuracy = \frac{error}{radius} \tag{17}$$

Where the estimated position of i-th anchor node is (x_i, y_i, z_i) and the (x, y, z) is the position of unknown node. The d_{ui} is the estimated distance between unknown node and i-th anchor node through DV-Hop method. There are n anchor nodes, so the error is the sum of errors which between all anchor nodes and unknown node. As the purpose of this article is reduce the localization error, so the fitness function can be described as follows:

$$f(x,y) = min(\sum_{i=1}^{n} (\frac{1}{hop_{ui}})^2 (\sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} - d_{ui})^2) \tag{18}$$

The hop_{ui} is the hop count between unknown node and i-th anchor node. From this equation, we can see the novel algorithm is used to look for the best position of unknown node, which is the most suitable for all anchor nodes. The main work of this article can be devided into four steps:

- Randomly deploying the sensor nodes on 3-D terrain as shown in Figure 1.
- Forming the connective network by using Eq 15.
- Calculating the distance between unknown nodes and anchor nodes.
- Determind the position of unknown nodes by utilizing novel algorithm and the distance information.

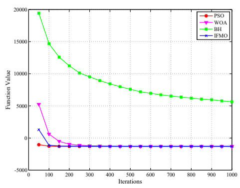
5. Simulation Results and Discussion.

5.1. **CEC 2013 Benchmark Function.** CEC 2013 benchmark function is a convincing test function for intelligence computing algorithm and it is proposed in 2013. This paper adopts CEC 2013 benchmark function to verify the performance of the novel algorithm and compares it with some existing algorithms, such as PSO, WOA and BH algorithms. There are 28 test functions and contains of unimodal functions, multimodal functions and composition functions. The simulations are implemented on the same notebook computer which equip with an i5-7300HQ CPU @2.5GHz and every result is means of 30 runs. Due to the poor performance of the original FMO under the benchmark function of CEC 2013, it did not participate in the results. The simulation results are presented in Table 1, the best results is marked by underline for every test function and the f_1 to f_5 are unimodal functions, they mainly checkout the convergence rate of optimization algorithm. In order to verify the performance of avoiding local optimal value of algorithms, the CEC 2013 introduced multimodal functions and denoted f6 to f_{20}. Besides, f_{21} to f_{28} are composition functions and their simulation results can reveal comprehensive performance of optimization algorithms.

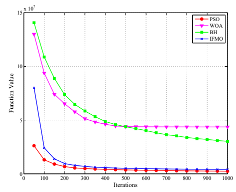
Table 1 lists the simulation results of the four algorithms under the CEC 2013 test suite. The search ability of the novel algorithm on 13 benchmark functions is better than other algorithms, and the same results are obtained on two functions. On the unimodal function, the new algorithm obtains the optimal values on f_3, f_5 and f_6, while the PSO algorithm has the best performance on other functions. The results about unimodal functions reveals the novel algorithm has excellent convergence rate as same as PSO. The multimodal function can verify the ability to avoid the local optimal value of the algorithm, and the performance of the proposed algorithm on five multi-peak functions is better than other algorithms. The results show that the algorithm has good global search capabilities. The algorithm obtains optimal values on five composition functions. In contrast, the PSO algorithm obtains three optimal results, while other algorithms cannot find optimal values on composition functions. The results of Table 1 indicates the novel algorithm has excellent convergence speed and is good at solving complex optimization problems. In addition, in the results of 30 experiments, the new algorithm achieved the smallest standard deviation in 21 test functions, so the stability of the algorithm is outstanding.

TABLE 1. Simulation Results of CEC 2013 Benchmark Function

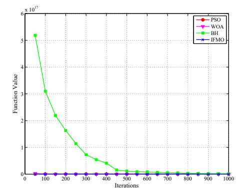
Functions	PSO		WOA		BH		IFMO	
Variable	Mean	Std	Mean	Std	Mean	Std	Mean	Std
f_1	-1.32×10^3	2.46×10^2	-1.29×10^3	6.59×10^1	5.64×10^3	8.53×10^2	-1.27×10^3	1.63×10^2
f_2	2.39×10^6	1.03×10^6	4.36×10^7	2.35×10^7	3.02×10^7	3.24×10^6	4.00×10^6	2.62×10^6
f_3	4.14×10^9	1.06×10^{10}	1.62×10^{11}	6.51×10^{11}	1.77×10^{15}	1.88×10^{15}	7.24×10^8	7.92×10^8
f_4	7.43×10^3	2.89×10^3	7.98×10^4	2.44×10^4	4.62×10^4	1.24×10^4	3.97×10^4	7.12×10^3
f_5	-9.74×10^2	4.61×10^1	-6.94×10^2	1.32×10^2	5.99×10^2	2.53×10^2	-9.95×10^2	2.44×10^1
f_6	-8.49×10^2	3.53×10^1	-7.58×10^2	5.69×10^1	3.77×10^2	2.15×10^2	-8.74×10^2	2.52×10^1
f_7	-7.26×10^2	3.38×10^1	2.67×10^4	9.89×10^4	2.99×10^4	2.16×10^4	-6.71×10^2	1.18×10^2
f_8	-6.79×10^2	5.96×10^{-2}	-6.79×10^2	7.43×10^{-2}	-6.79×10^2	7.20×10^{-2}	-6.79×10^2	6.58×10^{-2}
f_9	-5.82×10^2	2.96×10^0	-5.77×10^2	2.18×10^0	-5.77×10^2	2.49×10^0	-5.79×10^2	1.39×10^0
f_{10}	-4.48×10^2	7.98×10^1	-3.11×10^2	8.13×10^1	1.74×10^2	7.90×10^1	-4.80×10^2	1.47×10^1
f_{11}	-2.31×10^2	4.23×10^1	-1.01×10^2	6.77×10^1	-1.22×10^2	5.43×10^1	-2.53×10^2	1.17×10^1
f_{12}	-1.38×10^2	5.12×10^1	3.02×10^0	8.66×10^1	5.64×10^0	4.68×10^1	-1.43×10^2	1.62×10^1
f_{13}	-9.87×10^0	3.85×10^1	8.45×10^1	6.35×10^1	1.24×10^2	4.79×10^1	-3.36×10^1	1.97×10^1
f_{14}	2.36×10^3	5.23×10^2	3.25×10^3	4.97×10^2	3.92×10^3	4.36×10^2	4.37×10^3	2.00×10^2
f_{15}	2.69×10^3	4.23×10^2	3.51×10^3	7.19×10^2	3.76×10^3	5.84×10^2	4.33×10^3	2.59×10^2
f_{16}	2.02×10^2	3.69×10^{-1}	2.02×10^2	5.13×10^{-1}	2.01×10^2	4.88×10^{-1}	2.02×10^2	3.33×10^{-1}
f_{17}	4.33×10^2	2.20×10^1	6.44×10^2	7.55×10^1	5.86×10^2	6.15×10^1	5.17×10^2	1.48×10^1
f_{18}	5.47×10^2	1.51×10^1	7.40×10^2	7.28×10^1	6.81×10^2	5.09×10^1	6.16×10^2	1.25×10^1
f_{19}	5.08×10^2	2.36×10^0	5.54×10^2	2.06×10^1	1.90×10^3	5.27×10^2	5.06×10^2	1.92×10^0
f_{20}	6.10×10^2	8.30×10^{-2}	6.10×10^2	1.20×10^{-1}	6.10×10^2	1.62×10^{-1}	6.10×10^2	7.72×10^{-4}
f_{21}	1.01×10^3	7.42×10^1	1.29×10^3	2.12×10^2	1.80×10^3	4.71×10^1	1.07×10^3	3.75×10^1
f_{22}	4.28×10^3	5.37×10^2	4.85×10^3	5.38×10^2	5.52×10^3	5.70×10^2	5.82×10^3	3.18×10^2
f_{23}	4.53×10^3	7.03×10^2	5.37×10^3	4.89×10^2	5.83×10^3	5.22×10^2	6.04×10^3	2.32×10^2
f_{24}	1.27×10^3	1.05×10^1	1.28×10^3	8.46×10^0	1.29×10^3	9.77×10^0	1.26×10^3	6.85×10^0
f_{25}	1.38×10^3	1.14×10^1	1.38×10^3	7.67×10^0	1.41×10^3	1.27×10^1	1.37×10^3	6.00×10^0
f_{26}	1.48×10^3	7.30×10^1	1.48×10^3	8.11×10^1	1.41×10^3	2.13×10^0	1.42×10^3	5.09×10^1
f_{27}	2.14×10^3	8.79×10^1	2.27×10^3	6.64×10^1	2.34×10^3	8.10×10^1	2.10×10^3	4.06×10^1
f_{28}	4.38×10^3	4.50×10^2	5.53×10^3	8.05×10^2	5.34×10^3	5.01×10^2	3.57×10^3	7.20×10^2



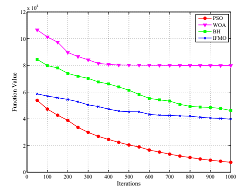
(a) $f1$



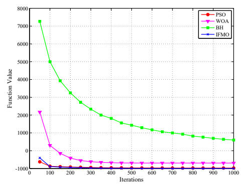
(b) $f2$



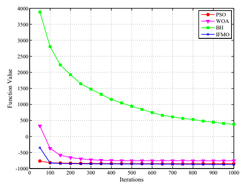
(c) $f3$



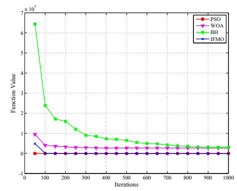
(d) $f4$



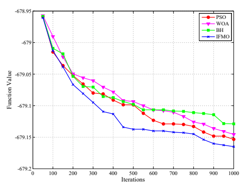
(e) $f5$



(f) $f6$



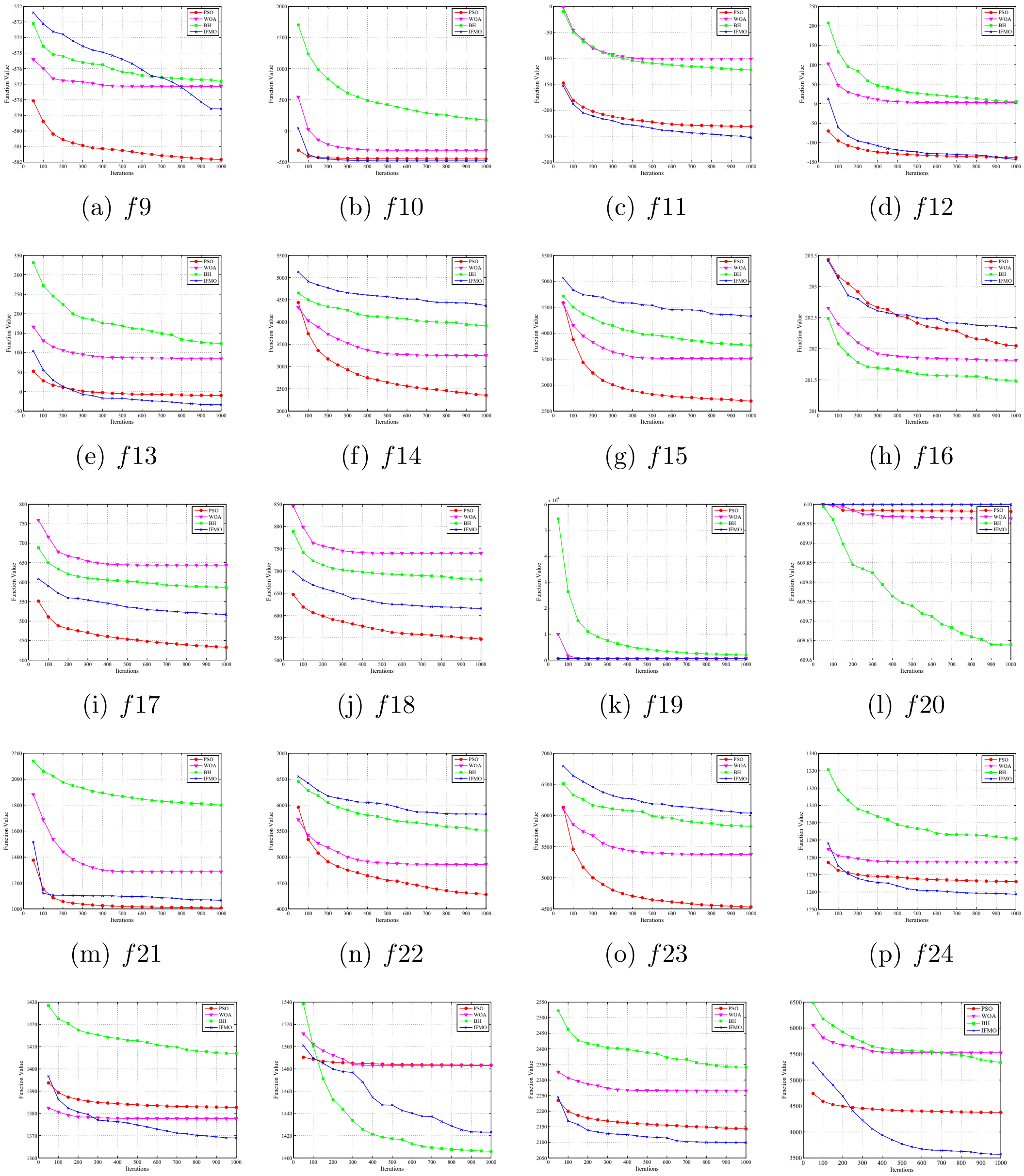
(g) $f7$



(h) $f8$

Figure 1. Simulation Results of CEC 2013 (1)

In order to intuitively present the performance of algorithm, the optimization process is showed in Figure 1 and Figure 2. On unimodal function, the proposed algorithm perform fast convergence rate, as adding the attract of best position of population to the novel algorithm, it can find the the value which close to the optimal value before 200 iterations on most unimodal functions. Benefit form the increase of diversity, the novel algorithm can jump out the local optimal value and further explore promising area. These figures obviously show the new algorithm continuously jumps out the local optimal value and finds the optimal value finally om f_8, f_{11}, f_{12} and f_{13}. About composition problems, the novel algorithm is outstanding from other algorithms, the concept of energy is introduced to this algorithm make the algorithm abandoning some valueless area timely. Therefor, the algorithm perform the excellent search ability on f_{21} to f_{28} functions.



(a) $f9$ (b) $f10$ (c) $f11$ (d) $f12$

(e) $f13$ (f) $f14$ (g) $f15$ (h) $f16$

(i) $f17$ (j) $f18$ (k) $f19$ (l) $f20$

(m) $f21$ (n) $f22$ (o) $f23$ (p) $f24$

(q) $f25$ (r) $f26$ (s) $f27$ (t) $f28$

FIGURE 2. Simulation Results of CEC 2013 (2)

5.2. **DV-Hop Localization Method of WSN on 3-D Terrain.** In this section, a new algorithm will be applied to reduce the positioning error of DV-Hop on 3-D terrain, and the simulation results are discussed. The 3D terrain used in this article is formed by Matlab, as shown in Figure 3. As can be seen from the figure, there are many "depression" and "mountains" that can effectively block the signal. Therefore, this 3-D terrain can y simulate a complex real-world environment.

The sensor nodes are randomly deployed on 3-D terrain as above figure shows in initial step, and the plane area of simulation terrain is 50 $\times$ 50 meters. In order to fully test the performance of the new algorithm applied to DV-Hop, two simulations are provided in this section. The data used to draw the graph is the average of 30 runs, and the positions of the nodes in the two simulations are generated independently. In the first

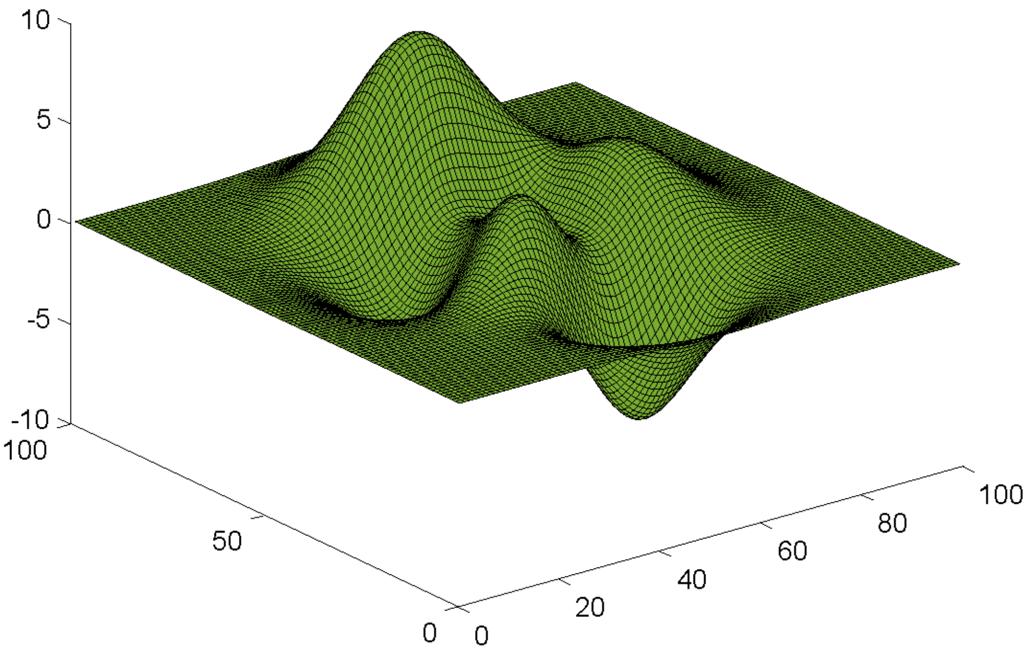


FIGURE 3. The 3-D Terrain which Sensor Nodes Deployed

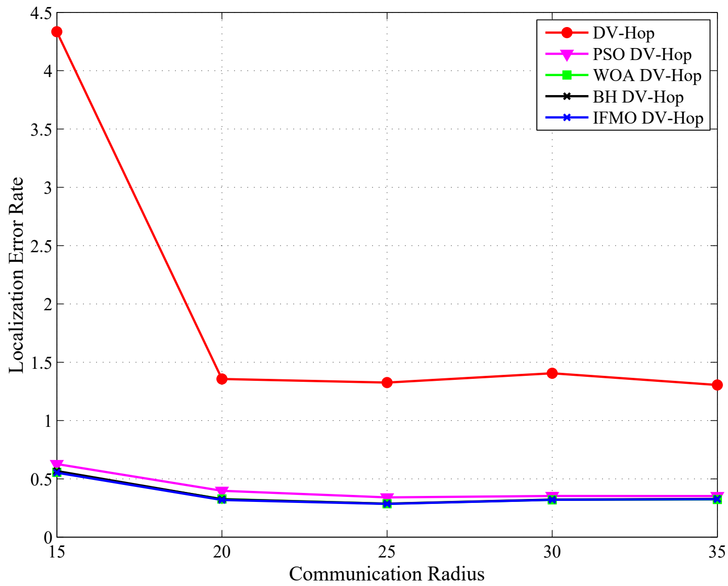


FIGURE 4. The Influence of Communication Radius on Results

simulation, there are 30 anchor nodes and 170 unknown nodes. The position of 200 nodes is randomly generated and save it into a matrix. Every group experiment has different communication radius and the results of this simulation are showed in Figure 4. The abscissa is the communication radius of sensor node, the ordinate is localization error rate and it can be calculated by Eq.17. The figure shows that WSN usually has a longer communication radius, positioning is more accurate, and the DV-Hop combined with intelligent computing is significantly superior to the original DV-Hop. When the communication radius is set to 30, the simulation result is worse than the radius set to 25. As the communication radius increases, the number of hops between sensor nodes

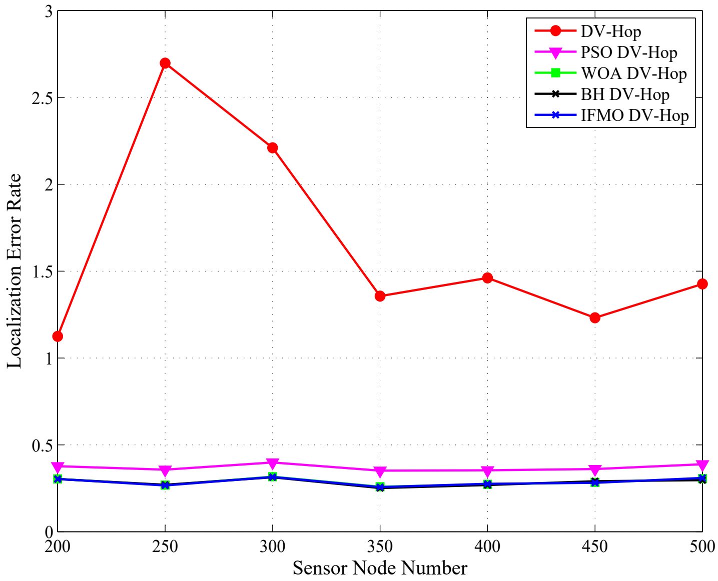


FIGURE 5. The Influence of Sensor Node Number on Results

becomes smaller, resulting in a larger positioning error. Therefore, if the communication radius is too long, the positioning accuracy cannot be improved. It is important to set an appropriate communication radius.

The number of node also has important influence on localization accuracy, so this paper presents the simulation results of different account of sensor nodes. The communication radius is set 20 meter for all nodes and the position of sensor nodes used in this simulation is stored in matrix. The account of anchor nodes is 30 for every group experiment. The Figure 5 indicates the experimental results. The localization error rate increases with number of nodes commonly for most method instead of DV-Hop, and there is a abnormal point when sensor node number is 300. This is because the position of the nodes in each set of experiments is generated independently, so the position information is different. When the number of sensor nodes is 300, there may be more obstacles between nodes than other experiments, or fewer hops between nodes, which will lead to an increase in the positioning error rate. Based on these simulation data, the intelligent calculation algorithm combined with DV-Hop can effectively improve the accuracy and stability of the original DV-Hop. In addition, the new algorithm is competitive and can reduce the localization error rate more effectively than the PSO algorithm.

6. **Conclusion.** In this paper, the FMO algorithm is extended to ASAFMO. The adaptive strategy is applied on energy and position update, the more energy the fish has, the faster speed the fish has. When a fish with a worse function value, the energy of this fish increase substantially. The CEC 2013 benchmark function is used to verify the performance of the new algorithm and compare it with other famous algorithms. Besides, the novel algorithm is applied to solve the localization problem of WSN on 3-D terrain. The simulation results indicate the accuracy of DV-Hop is improved obviously by combining with the novel algorithm. The proposed scheme can be further improved by adopting some efficient approaches [40, 41, 42, 43, 44].

REFERENCES

[1] Dervis Karaboga. An idea based on honey bee swarm for numerical optimization. Technical report, Technical report-tr06, Erciyes university, engineering faculty, computer . . . , 2005.

[2] Pei-Wei TSai, Jeng-Shyang Pan, Bin-Yih Liao, and Shu-Chuan Chu. Enhanced artificial bee colony optimization. *International Journal of Innovative Computing, Information and Control*, 5(12):5081–5092, 2009.

[3] Hui Wang, Zhijian Wu, Shahryar Rahnamayan, Hui Sun, Yong Liu, and Jeng-shyang Pan. Multi-strategy ensemble artificial bee colony algorithm. *Information Sciences*, 279:587–603, 2014.

[4] Darrell Whitley. A genetic algorithm tutorial. *Statistics and computing*, 4(2):65–85, 1994.

[5] JS Pan, FR McInnes, and MA Jack. Application of parallel genetic algorithm and property of multiple global optima to vq codevector index assignment for noisy channels. *Electronics Letters*, 32(4):296–297, 1996.

[6] Rainer Storn and Kenneth Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.

[7] Jeng-Shyang Pan, Nengxian Liu, and Shu-Chuan Chu. A hybrid differential evolution algorithm and its application in unmanned combat aerial vehicle path planning. *IEEE Access*, 8:17691–17712, 2020.

[8] Zhenyu Meng, Jeng-Shyang Pan, and Kuo-Kun Tseng. Pade: An enhanced differential evolution algorithm with novel control parameter adaptation schemes for numerical optimization. *Knowledge-Based Systems*, 168:80–99, 2019.

[9] Russell Eberhart and James Kennedy. Particle swarm optimization. In *Proceedings of the IEEE international conference on neural networks*, volume 4, pages 1942–1948. Citeseer, 1995.

[10] Jing J Liang, A Kai Qin, Ponnuthurai N Suganthan, and S Baskar. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE transactions on evolutionary computation*, 10(3):281–295, 2006.

[11] Hui Wang, Wenjun Wang, and Zhijian Wu. Particle swarm optimization with adaptive mutation for multimodal optimization. *Applied Mathematics and Computation*, 221:296–305, 2013.

[12] Jeng-Shyang Pan, Thi-Kien Dao, Tien-Szu Pan, et al. Compact particle swarm optimization for optimal location of base station in wireless sensor network. In *International Conference on Genetic and Evolutionary Computing*, pages 54–62. Springer, 2016.

[13] Shu-Chuan Chu, John F Roddick, and Jeng-Shyang Pan. Ant colony system with communication strategies. *Information Sciences*, 167(1-4):63–76, 2004.

[14] Shu-Chuan Chu, John F Roddick, Che-Jen Su, and Jeng-Shyang Pan. Constrained ant colony optimization for data clustering. In *Pacific Rim International Conference on Artificial Intelligence*, pages 534–543. Springer, 2004.

[15] Shu-Chuan Chu, Pei-Wei Tsai, and Jeng-Shyang Pan. Cat swarm optimization. In *Pacific Rim international conference on artificial intelligence*, pages 854–858. Springer, 2006.

[16] Pei-Wei Tsai, Jeng-Shyang Pan, Shyi-Ming Chen, and Bin-Yih Liao. Enhanced parallel cat swarm optimization based on the taguchi method. *Expert Systems with Applications*, 39(7):6309–6319, 2012.

[17] Pei-Wei Tsai, Jeng-Shyang Pan, Shyi-Ming Chen, Bin-Yih Liao, and Szu-Ping Hao. Parallel cat swarm optimization. In *2008 international conference on machine learning and cybernetics*, volume 6, pages 3328–3333. IEEE, 2008.

[18] Zhenyu Meng and Jeng-Shyang Pan. Quasi-affine transformation evolutionary (quatre) algorithm: A parameter-reduced differential evolution algorithm for optimization problems. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 4082–4089. IEEE, 2016.

[19] Nengxian Liu, Jeng-Shyang Pan, and Jason Yang Xue. An orthogonal quasi-affine transformation evolution (o-quatre) algorithm for global optimization. In *Advances in Intelligent Information Hiding and Multimedia Signal Processing*, pages 57–66. Springer, 2020.

[20] Zhenyu Meng and Jeng-Shyang Pan. A competitive quasi-affine transformation evolutionary (c-quatre) algorithm for global optimization. In *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 001644–001649. IEEE, 2016.

[21] Zhi-Gang Du, Jeng-Shyang Pan, Shu-Chuan Chu, Han-Jiang Luo, and Pei Hu. Quasi-affine transformation evolutionary algorithm with communication schemes for application of rssi in wireless sensor networks. *IEEE Access*, 8:8583–8594, 2020.

[22] Abdolreza Hatamlou. Black hole: A new heuristic optimization approach for data clustering. *Information sciences*, 222:175–184, 2013.

[23] Jeng-Shyang Pan, Qing-Wei Chai, Shu-Chuan Chu, and Ning Wu. 3-d terrain node coverage of wireless sensor network using enhanced black hole algorithm. *Sensors*, 20(8):2411, 2020.

[24] Georges R Harik, Fernando G Lobo, and David E Goldberg. The compact genetic algorithm. *IEEE transactions on evolutionary computation*, 3(4):287–297, 1999.

[25] Thi-Kien Dao, Jeng-Shyang Pan, Trong-The Nguyen, Shu-Chuan Chu, and Chin-Shiuh Shieh. Compact bat algorithm. In *Intelligent Data analysis and its Applications, Volume II*, pages 57–68. Springer, 2014.

[26] Ai-Qing Tian, Shu-Chuan Chu, Jeng-Shyang Pan, Huanqing Cui, and Wei-Min Zheng. A compact pigeon-inspired optimization for maximum short-term generation mode in cascade hydroelectric power station. *Sustainability*, 12(3):767, 2020.

[27] Xingsi Xue and Jeng-Shyang Pan. A compact co-evolutionary algorithm for sensor ontology meta-matching. *Knowledge and Information Systems*, 56(2):335–353, 2018.

[28] Xingsi Xue and Junfeng Chen. A compact co-firefly algorithm for matching ontologies. In *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 2629–2632. IEEE, 2019.

[29] Chaoli Sun, Yaochu Jin, Ran Cheng, Jinliang Ding, and Jianchao Zeng. Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems. *IEEE Transactions on Evolutionary Computation*, 21(4):644–660, 2017.

[30] Qing-wei Chai, Shu-Chuan Chu, Jeng-Shyang Pan, Pei Hu, and Wei-min Zheng. A parallel woa with two communication strategies applied in dv-hop localization method. *EURASIP Journal on Wireless Communications and Networking*, 2020(1):1–10, 2020.

[31] Pei Hu, Jeng-Shyang Pan, Shu-Chuan Chu, Qing-Wei Chai, Tao Liu, and Zhong-Cui Li. New hybrid algorithms for prediction of daily load of power network. *Applied Sciences*, 9(21):4514, 2019.

[32] Ian F Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A survey on sensor networks. *IEEE Communications magazine*, 40(8):102–114, 2002.

[33] Azzedine Boukerche, Horacio ABF Oliveira, Eduardo F Nakamura, and Antonio AF Loureiro. Localization systems for wireless sensor networks. *IEEE wireless Communications*, 14(6):6–12, 2007.

[34] Jin-Zhao Lin, Xiao-Bing Chen, and Hai-Bo Liu. Iterative algorithm for locating nodes in wsn based on modifying average hopping distances. *Journal on Communications*, 30(10):107–113, 2009.

[35] Xiao Chen and Benliang Zhang. Improved dv-hop node localization algorithm in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 8(8):213980, 2012.

[36] Jeng-Shyang Pan, Pei-Wei Tsai, and Yu-Bin Liao. Fish migration optimization based on the fishy biology. In *2010 Fourth International Conference on Genetic and Evolutionary Computing*, pages 783–786. IEEE, 2010.

[37] Dragoş Niculescu and Badri Nath. Dv based positioning in ad hoc networks. *Telecommunication Systems*, 22(1-4):267–280, 2003.

[38] Nengxian Liu, Jeng-Shyang Pan, et al. A bi-population quasi-affine transformation evolution algorithm for global optimization and its application to dynamic deployment in wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2019(1):175, 2019.

[39] David H Wolpert and William G Macready. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.

[40] Xiao-Xue Sun, Jeng-Shyang Pan, Shu-Chuan Chu, Pei Hu, and Ai-Qing Tian. A novel pigeon-inspired optimization with quasi-affine transformation evolutionary algorithm for dv-hop in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 16(6):1550147720932749, 2020.

[41] Pei-Cheng Song, Jeng-Shyang Pan, and Shu-Chuan Chu. A parallel compact cuckoo search algorithm for three-dimensional path planning. *Applied Soft Computing*, page 106443, 2020.

[42] Chin-Shiuh Shieh, Van-Oanh Sai, Tsair-Fwu Lee, Quang-Duy Le, and Yuh-Chung Lin. Node localization in wsn using heuristic optimization approaches. *J Netw Intell*, 2:275–286, 2017.

[43] Zhengyi Tang, Xingsi Xue, Jinshui Wang, and Zhi Hang. The logic sense request of wsn and its analysis model.

[44] Zhenyu Meng, Jeng-Shyang Pan, and Wei-min Zheng. Differential evolution utilizing a handful top superior individuals with bionic bi-population structure for the enhancement of optimization performance. *Enterprise Information Systems*, 14(2):221–242, 2020.