

## Novel DV-hop Method Based on Krill Swarm Algorithm Used for Wireless Sensor Network Localization

Yang Sun<sup>1</sup>, Shoulin Yin<sup>\*2</sup>, Jie Liu<sup>3</sup>

Software College, Shenyang Normal University, Shenyang, China

No.253, HuangHe Bei Street, HuangGu District, Shenyang, P.C 110034 - China

\*Corresponding author, email: 17247613@qq.com<sup>1</sup>, 352720214@qq.com<sup>2</sup>, nan127@sohu.com<sup>3</sup>

### Abstract

*Wireless sensor network (WSN) is self-organizing network; it consists of a large number of sensor nodes with perception, calculation ability and communication ability. Generally, the floor, walls or people moving has an effect on indoor localization, so it will result in multi-path phenomena and decrease signal strength; also the received signal strength indicator (RSSI) is unable to gain higher accuracy of positioning. In order to improve the WSN positioning accuracy in indoor condition, in this paper, we firstly propose krill swarm algorithm used for WSN localization. We detailed analyze the multilateral measurement method in range-free distance vector-hop (DV-hop) localization algorithm. The position problem can be transformed into a global optimization problem. Then, we adequately utilize the advantage of calculating optimization problem and apply the krill swarm algorithm into the stage of estimating unknown node coordinates in DV-hop algorithm to realize localization. Finally, the simulation experience results show that the localization with krill swarm algorithm has an obviously higher positioning precision and accuracy stability with different anchor node proportion and nodes.*

**Keywords:** WSN, RSSI, DV-hop localization algorithm, krill swarm algorithm, range-free distance, multilateral measurement method

Copyright © 2016 Universitas Ahmad Dahlan. All rights reserved.

### 1. Introduction.

Recently, people have a growing demand for indoor location information services. The traditional GPS and cellular network positioning technology cannot meet the requirements of indoor positioning. Fortunately, wireless sensor network (WSN) [1] makes up for the deficiencies of the traditional positioning technology, which is widely used in exhibition center, intelligent building, sports venues and other fields. In indoor positioning technologies, the localization algorithm performance is directly related to the location accuracy of sensor network node. Currently, WSN localization technology has two major methods: range distance and range-free distance. It includes convex programming localization algorithm [2], centroid localization algorithm and DV-hop localization algorithm [3] based on the range-free distance in WSN node localization. These algorithms work without additional hardware support and high cost. The positioning precision, however, cannot meet the requirements of indoor localization accuracy. The range distance localization algorithms contains time of arrival (TOA) algorithm [4], receive signal strength indication (RSSI) algorithm and angle of arrival (AOA) algorithm. Moreover, some newest machine learning algorithms have been proposed such as [5-7]. In this paper, we design a new method based on Krill Swarm algorithm used for wireless sensor network localization. What's more, we also state its working principle and demonstrate the indoor localization performance through rigorous experiments. The rest of this paper is organized as follows: The next section of the paper introduces the general principles of the Krill Swarm algorithm and DV-hop algorithm. Section3 proposes the Krill Swarm algorithm used for the WSN localization. After that, experimental results with the Krill Swarm algorithm are presented in section 4. The final section of the paper includes the concluding remarks.

### 2. Novel DV-hop Method Based on Krill Swarm Algorithm

In this section, we detailed illustrate our new scheme. Anchor nodes (including unknown nodes and known nodes) are regarded as krill swarm. The gather behavior of krill is a

process of multi-objective, which is very closely to sensor nodes. The aim of our new method is to add sensor nodes density and approach to main node. Using krill swarm algorithm will make each node change their position frequently toward the optimal value direction moderately. Position changing and foraging behavior caused by other anchor nodes contain a local search strategy and a global search strategy. Two kinds of strategies work in parallel. After collecting the optimal position of nodes based on krill swarm algorithm, then we utilize DV-hop method to calculate the position of unknown nodes. Finally, we demonstrate its high performance and efficiency through many experiences. The anchor nodes model with krill swarm algorithm is as follows:

1. Determine the anchor nodes Lagrangian model based on krill swarm.

$$\frac{dX_i}{dt} = N_i + F_i + D_i \quad (1)$$

Where  $i$  is  $i$ -th anchor node.  $X_i$  is the state of anchor node.  $N_i$  denotes velocity vector of induced movement.  $F_i$  is velocity vector of finding key node.  $D_i$  is random diffusion velocity vector.

2. Motion induced by other anchor node individuals.

$$N_i^{new} = N_{min} (\alpha_i^{local} + \alpha_i^{target}) + \omega_n N_i^{old} \quad (2)$$

Where  $N_{min}=0.1m/s$  is the minimum velocity of induced movement in WSN based on the real situation.  $\omega_n \in (0,1)$  is the inertia weight of induced movement.  $N_i^{old}$  is the last velocity vector of induced movement.  $\alpha_i^{local}$  is local influence of adjacent anchor node.

$$\alpha_i^{local} = \sum_{j=1}^{NN} \hat{K}_{i,j} \hat{X}_{i,j} \quad (3)$$

$$\hat{K}_{i,j} = \frac{K_i - K_j}{K_{worst} - K_{best}} \quad (4)$$

$$\hat{X}_{i,j} = \frac{X_j - X_i}{\|X_j - X_i\| + \varepsilon} \quad (5)$$

Where  $K_i$  denotes fitness of  $i$ -th anchor node.  $K_j$  is fitness of  $j$ -th neighborhood anchor node.  $K_{best}$  and  $K_{worst}$  are the best fitness and the worst fitness respectively.  $X_i$  is the state of  $i$ -th anchor node.  $X_j$  is the state of  $j$ -th anchor node.  $\varepsilon$  is a small positive number to avoid singularity.  $NN$  is the number of adjacent anchor node, which can be determined by perception distance of each anchor node  $d_i$ .  $d_i$  can be expressed as:

$$d_i = \frac{1}{5N} \sum_{j=1}^N \|X_i - X_j\| \quad (6)$$

Where  $N$  is the total number of anchor node.

$\alpha_i^{target}$  is influence of the optimal anchor node as (7)

$$\alpha_i^{target} = 2(rand + \frac{1}{I_{max}}) \hat{K}_{i,best} \hat{X}_{i,best} \quad (7)$$

Where  $rand$  is random number between 0 and 1.  $I$  is the current iterations number.  $I_{max}$  is the maximum iteration.

3. Finding key node.

$$F_i^{new} = V_f (\beta_i^{food} + \beta_i^{best}) + \omega_f F_i^{old} \tag{8}$$

Where  $V_f=0.2m/s$  is speed of finding key node.  $\omega_f \in (0,1)$  is the inertia weight of finding key node.  $\beta_i^{best}$  is the best previously visited position of the  $i$ -th anchor node individual.  $F_i^{old}$  is the last velocity vector of finding key node.  $\beta_i^{food}$  is influence of  $i$ -th anchor node, which can be represented by:

$$X_{food} = \sum_{i=1}^N X_i / K_i \div \sum_{j=1}^N 1 / K_j \tag{9}$$

$$\beta_i^{food} = 2(1 - \frac{I}{I_{max}}) \hat{K}_{i,food} \hat{X}_{i,food} \tag{10}$$

So the influence of current  $i$ -th node is:

$$\beta_i^{best} = \hat{K}_{i,best} \hat{X}_{i,best} \tag{11}$$

This is the finding key node stage.

4. Stochastic diffusion process.

$$D_i^{new} = D_{max} (1 - \frac{I}{I_{max}}) \delta \tag{12}$$

Where  $D_{max} \in (0.002, 0.01)m/s$  is the speed of random diffusion.  $\delta$  is the direction vector, which is subjected to  $(-1, 1)$  uniform distribution.

5. Updating anchor node positions.

$$X_i^I = X_i^{I-1} + \Delta t (N_i^{new} + F_i^{new} + D_i^{new}) \tag{13}$$

Where  $\Delta t$  is interval selected according to the real situation. In this paper,  $\Delta t = 1s$ . So every anchor node will get the optimal position in the WSN including the unknown nodes. We can use the following figure to show the anchor nodes movement track.

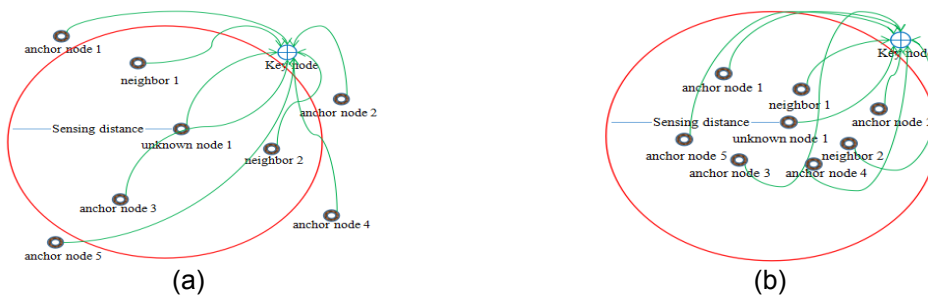


Figure 1. This figure shows the position of anchor nodes in WSN before and after using krill swarm algorithm. (a) Position of anchor nodes without krill swarm algorithm; (b) Position of anchor nodes with krill swarm algorithm

### 2.1. Process of novel DV-hop based on krill swarm.

We design the novel DV-hop algorithm as follows:

Anchor nodes average jump distance computation formula is:

$$S_{hop_i} = \sum_{j \neq i} \sqrt{(x_{ii} - x_{jj})^2 + (y_{ii} - y_{jj})^2} / \sum_{j \neq i} S_{hop_{ij}} \quad (14)$$

Where  $S_{hop_i}$  is corresponding average jump distance of anchor node  $i$ .  $j$  is other anchor node number in anchor node  $i$  data table.  $S_{hop_{ij}}$  is the hop count between  $i$  and  $j$ .  $(x_{ii}, y_{ii})$  and  $(x_{jj}, y_{jj})$  is the coordinate of  $X_i^l$ ,  $X_j^l$  (the obtained best position by krill swarm) respectively. Therefore, the  $S_{hop_i}$  is also the optimal corresponding average jump distance. According to the recorded hop information, unknown nodes will calculate the distance from itself to anchor node after receiving average jump distance by formula (15).

$$d_i = S_{hop} \cdot h_i \quad (15)$$

After calculating the distance, DV-hop algorithm solves the coordinate of unknown nodes by multilateral measurement method.

Known  $U_{11}(x_{11}, y_{11})$ ,  $U_{22}(x_{22}, y_{22})$ , ...,  $U_{nn}(x_{nn}, y_{nn})$  and the unknown node  $K(x, y)$  when there are  $n(n \geq 3)$  nodes.  $d_{11}$ ,  $d_{22}$ , ...,  $d_{nn}$  are the distance from  $U_{11}$ ,  $U_{22}$ , ...,  $U_{nn}$  to  $K$  respectively. So:

$$(x - x_{11})^2 + (y - y_{11})^2 = d_{11}^2, \dots, (x - x_{nn})^2 + (y - y_{nn})^2 = d_{nn}^2 \quad (16)$$

It can be expressed by system of linear equations  $AL=B$ ,  $L=(x, y)^T$ ,

$$A = [(2(x_{11} - x_{nn}) \ 2(y_{11} - y_{nn})), \dots, (2(x_{n-1|n-1} - x_{nn}) \ 2(y_{n-1|n-1} - y_{nn}))]^T \quad (17)$$

$$B = (x_{n-1|n-1}^2 - x_{nn}^2 + y_{n-1|n-1}^2 - y_{nn}^2 - d_{n-1|n-1}^2 + d_{nn}^2)^T \quad (18)$$

Because of the range error, environment factors and communication,  $AL=B$  can be rewritten as  $AL+\varepsilon=B$ .  $\varepsilon$  denotes the energy consumption parameter in WSN. We can get least-square solutions of equations by standard least squares method:  $L=(A^T A)^{-1} A^T B$ .  $d_n$  is measuring distance with error included in the  $B$ . So the calculation result is limited by  $d_n$ . If the error of  $d_n$  is small enough,  $L$  can meet the requirements. However, if the error of  $d_n$  is big enough, the error of result is big too. Although this method simplifies the process of solving nonlinear equations, it may reduce accuracy of solution. Aiming at this situation, this paper transforms the problem into global optimization.

Assuming that  $f_n$  is the measurement error between unknown node and anchor node, so  $f_n = \sqrt{(x - x_{nn})^2 + (y - y_{nn})^2} - d_n$ .

Then the unknown node  $K(x, y)$  can be solved.

$$f(x, y) = \sum_{i=1}^n (\sqrt{(x - x_{ii})^2 + (y - y_{ii})^2} - d_i) \quad (19)$$

When  $f(x, y)$  is the minimum value, the total error is very small and  $(x, y)$  is the most approach to the real value, so the  $(x, y)$  is optical value. Solution of  $f(x, y)$  is not limited to an equation, namely although range error of some anchor nodes is big, it has little effect on the solution of  $(x, y)$ .

As mentioned above, node localization problem is converted into global optimization problem. For formula (19), the traditional methods are difficult to solve it. But krill swarm

algorithm is a better and newest method to solve global optimization problem. Experiment results show that krill swarm algorithm is a better scheme used for WSN localization.

## 2.2. Detailed DV-hop Algorithm Based on Krill Swarm

Step 1: Initialization. Set the generation counter  $Q=1$ ; initialize the population  $P$  of  $NP$  anchor nodes randomly and each node corresponds to a potential solution to the given problem; set the finding key node speed  $V_i$ , the minimum diffusion speed  $D_{min}$ , and the maximum induced speed  $N_{max}$ .

Step 2: Fitness evaluation. Evaluate each anchor node according to its position.

Step 3: While the termination criteria is not satisfied or  $Q < MaxGeneration$  do.

Sort the population of nodes from best to worst.

for  $i=1: NP$  do

Perform the following motion calculation.

Motion induced by the presence of other nodes.

Finding key node.

Physical diffusion.

Implement the genetic operators.

Update the node individual position in the search space.

Evaluate each krill individual according to its position.

end for  $i$

Sort the population of node from best to worst and find the current best.

$Q=Q+1$ .

Step 4: End while.

Step 5: Get processed anchor node positions.

Step 6: DV-hop correction.

If  $RSSI \text{ value} > T$ , hops count=0.5. Otherwise, hops count=1.

Step 7: Hops between nodes acquisition.

Step 8: The average hop distance calculation.

Step 9: Position to calculate.

Step 10: Finish

## 3. Simulation Experiments and Analysis.

In this section, we make experiments to verify the performance our algorithm. The experiments of DV-hop algorithm based on krill swarm is made on MATLAB platform. In  $100m \times 100m$  region, we randomly scatter 200 nodes. The fixed node communication radius is 30m. Changing the number of anchor node from 20 to 55. We also make a comparison to DV-hop and the method (IDV-hop) in reference [8] with our new scheme (KSDV-hop). After experiments, we get the results and average value as Figure 2.

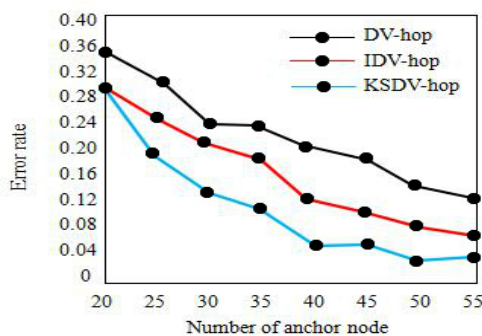


Figure 2. The effect of number of anchor node on error rate

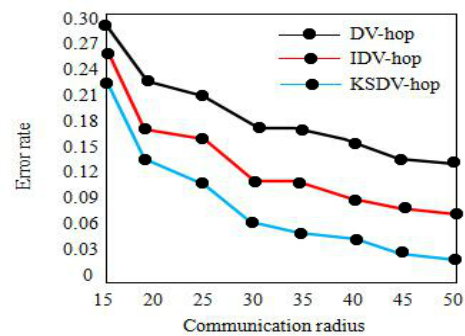


Figure 3. The effect of communication radius on error rate

From Figure 2, we can know that this new method is superior to DV-hop algorithm. When the number of anchor node is very small, the error rate decreases rapidly. When the anchor node number is steady, the error rate remains unchanged, which indicates that KSDV-hop algorithm has little requirement for anchor node number. As we all know, anchor node cost with positioning function is far larger than unknown node in WSN, as well as the energy consumption. Therefore, reducing anchor nodes plays an important role in reducing network cost and improving network life cycle. Table 1 is the corresponding data of Figure 3. Keeping the anchor node 50 unchanged, we change the communication radius from 15m to 50m. And it conducts 100 Monte-Carlo simulations as Figure 3.

Table 1. Error rate value with different anchor nodes

Algorithm	Number of anchor nodes							
	20	25	30	35	40	45	50	55
DV-hop	0.361	0.318	0.251	0.262	0.212	0.202	0.168	0.124
IDV-hop	0.301	0.245	0.221	0.197	0.131	0.109	0.105	0.079
KSDV-hop	0.301	0.199	0.146	0.118	0.057	0.061	0.035	0.038

Table 2 is the corresponding data of Figure 4. It shows that Max-Hop of network is decreasing from 15m to 50m. So the position accuracy will improve. The whole of error rate experiences a decreasing trend. On the whole, KSDV-hop algorithm is superior to DV-hop algorithm under any communication radius. Compared to [8], the error rate is very small with our new method. And with proper communication radius, results of KSDV-hop are more stable.

Table 2. Error rate value with different communication radius

Algorithm	Communication radius							
	15	20	25	30	35	40	45	50
DV-hop	0.291	0.235	0.232	0.183	0.184	0.179	0.152	0.151
IDV-hop	0.268	0.178	0.173	0.117	0.118	0.093	0.089	0.083
KSDV-hop	0.229	0.136	0.111	0.062	0.058	0.051	0.039	0.028

What's more, we make experiments with 10 standard testing functions (Multimodal functions: Ackley  $f_1$ , Fletcher-Powell  $f_2$ , Griewank  $f_3$ , Penalty  $f_4$ , Quartic  $f_5$ , Rastrigin  $f_6$ ; Unimodal functions: Rosenbrock  $f_7$ , Schwefel 12  $f_8$ , Sphere  $f_9$ , Step  $f_{10}$ ) to compare DV-hop, IDV-hop, KSCO (Krill swarm crossover operator in [9]), CA (cuckoo algorithm [10]), PSO (particle swarm optimization in [11]) to our KSDV-hop method. Supposing that population size  $NP=50$ , and maximum generation  $Maxgen=50$  for each algorithm. We ran 100 Monte Carlo simulations of each algorithm on each bench mark function to get representative performances and the results are as table4 including average minima value ( $amv$ ) and best minima value ( $bmv$ ).

Table 3. Simulation results

	DV-hop		IDV-hop		KSCO		CA		PSO		KSDV-hop	
	$amv$	$bmv$	$amv$	$bmv$	$amv$	$bmv$	$amv$	$bmv$	$amv$	$bmv$	$amv$	$bmv$
$f_1$	4.42	7.64	3.75	5.38	3.52	4.13	2.58	5.84	3.64	5.87	1.42	1.29
$f_2$	6.87	6.59	4.62	3.35	2.38	2.84	3.64	1.62	4.73	7.66	1.23	1.08
$f_3$	32.64	36.12	18.97	18.05	9.65	16.87	11.32	20.85	20.56	22.37	4.26	1.59
$f_4$	$2.3e^3$	$3.2e^3$	25.63	18.67	18.95	15.66	$1.4e^4$	$3.6e^4$	$3.1e^3$	$2.8e^3$	5.57	3.17
$f_5$	86.34	67.59	52.87	48.67	35.64	35.65	44.83	45.84	96.57	87.34	18.23	9.63
$f_6$	86.34	78.65	75.12	65.34	46.95	45.77	50.28	46.29	87.64	60.85	20.31	1.67
$f_7$	8.87	18.36	6.53	11.78	5.55	6.53	6.13	5.78	7.86	10.46	1.25	1.45
$f_8$	6.95	12.47	5.21	12.31	3.37	5.87	4.29	6.75	5.28	8.87	1.46	1.11
$f_9$	12.64	9.76	10.28	8.76	8.98	6.54	6.51	6.66	13.74	7.85	1.12	1.13
$f_{10}$	7.87	13.81	3.94	8.57	3.87	5.36	4.62	4.21	9.55	4.45	1.57	1.05

From Figure 3, we can know that KSDV-hop is the most effectiveness algorithm to get objective function minimum on average value and best value on the ten functions. What's more, to further prove the performance of proposed algorithm, we use the above different algorithms

to get best function value as Figure 4-13 for the ten functions. And the position value as Figure 14. Figure 4-13 show that KSDV-hop can get optimization results in a short time with a low generation number. Among the six optimization algorithms, our method performs the best and most effectively when solving the global numerical optimization problems and significantly outperforms the other five approaches.

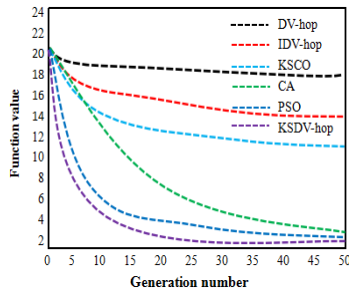


Figure 4.  $f_1$  comparison

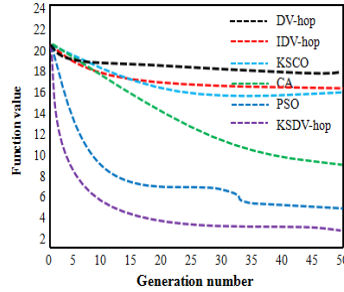


Figure 5.  $f_2$  comparison

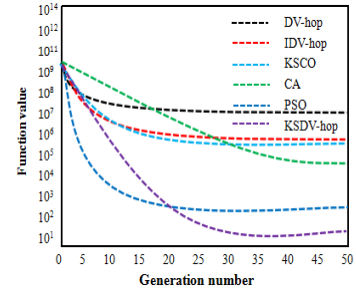


Figure 6.  $f_3$  comparison

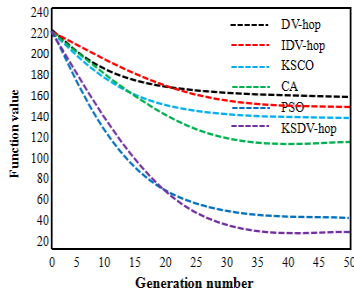


Figure 7.  $f_4$  comparison

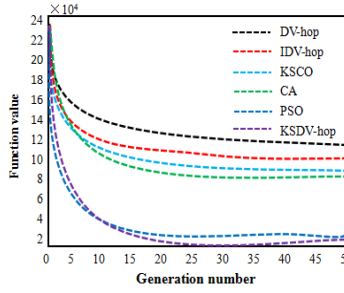


Figure 8.  $f_5$  comparison

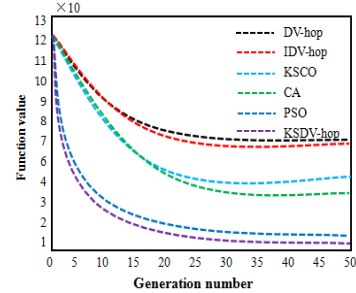


Figure 9.  $f_6$  comparison

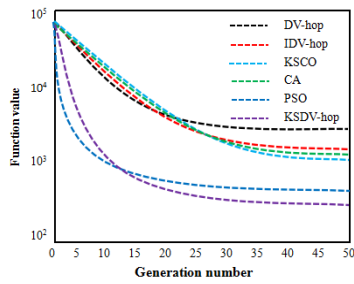


Figure 10.  $f_7$  comparison

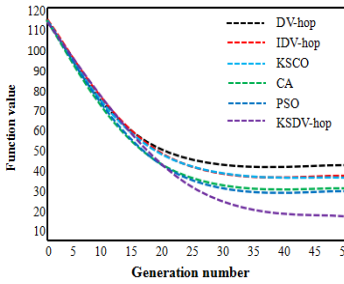


Figure 11.  $f_8$  comparison

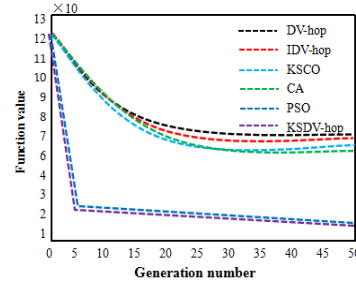


Figure 12.  $f_9$  comparison

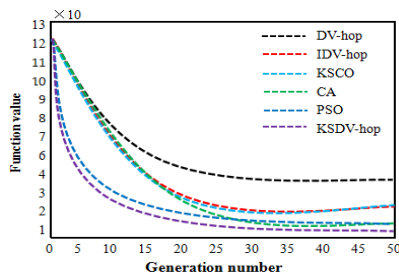


Figure 13.  $f_{10}$  comparison

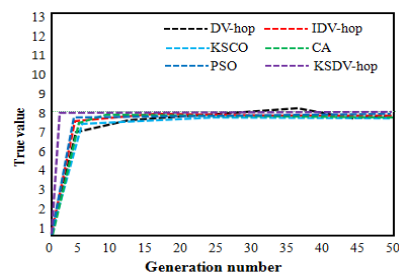


Figure 14. True value with different algorithms

#### 4. Conclusions

This paper makes an analysis for traditional DV-hop. In that there are many extra factors, such as environment, measurement error etc. So we propose a novel DV-hop based on krill swarm algorithm. This new method can speed up the global convergence rate without losing the strong robustness of the basic DV-hop. From the analysis of the experimental results, it can be concluded that the proposed DV-hop method uses the information in past solutions more efficiently when compared to the other population based optimization algorithms such as IDV-hop, CA, KSCO, PSO. Based on the results, KSDV-hop significantly improves the performance of DV-hop on most multimodal and unimodal problems. In addition, KSDV-hop is simple and easy to implement.

#### References

- [1] Shahrokhzadeh M, Haghghat AT, Mahmoudi F, et al. *A Heuristic Method for Wireless Sensor Network Localization*. *Procedia Computer Science*. 2011; 5: 812-819.
- [2] K Ren, F Zhuang. Research and improvement of mobile anchor node localization algorithm based on convex programming. *Chinese Journal of Sensors and Actuators*. 2014; 27(10): 1406-1411.
- [3] W Guo, J Wei. Optimization Research of the DV-Hop Localization Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2014; 12(4): 2735-2742.
- [4] Sathyan T, Hedley M, Humphrey D. A multiple candidate time of arrival algorithm for tracking nodes in multipath environments. *Signal Processing*. 2012; 92(7): 1611-1623.
- [5] Salem O, Guerassimov A, Mehaoua A, et al. Anomaly Detection in Medical Wireless Sensor Networks using SVM and Linear Regression Models. *International Journal of E-Health and Medical Communications*. 2016; 5(1): 20-45.
- [6] Lv T, Gao H, Li X, et al. Space-Time Hierarchical-Graph Based Cooperative Localization in Wireless Sensor Networks. *IEEE Transactions on Signal Processing*. 2016; 64(2): 322-334.
- [7] Pan MS, Lee YH. Fast convergecast for low-duty-cycled multi-channel wireless sensor networks. *Ad Hoc Networks*. 2016; 40: 1-14.
- [8] Qin Shan Zhao, Yu Lan Hu. An improved DV-Hop localisation algorithm. *International Journal of Wireless and Mobile Computing*. 2016; 10(1).
- [9] Gandomi AH, Alavi AH. Krill herd: A new bio-inspired optimization algorithm. *Communications in Nonlinear Science & Numerical Simulations*. 2012; 17(12): 4831-4845.
- [10] Wang G, Guo L, Duan H, et al. A hybrid metaheuristic DE/CS algorithm for UCAV three-dimension path planning. *Scientific World Journal*. 2012; 2012(9): 2977-2991.
- [11] Sun Y, Zhang L, Gu X. A hybrid co-evolutionary cultural algorithm based on particle swarm optimization for solving global optimization problems. *Neurocomputing*. 2012; 98(18): 76-89.