

# Optimal distributed decision in wireless sensor network using gray wolf optimization

Ibrahim Ahmed Saleh<sup>1</sup>, Omar Ibrahim Alsaif<sup>2</sup>, Maan A. Yahya<sup>3</sup>

<sup>1</sup>Department of Software, College of Computers and Mathematics, University of Mosul, Iraq

<sup>2</sup>Mosul Technical Institute, Northern Technical University, Iraq

<sup>3</sup>Ninevah Technical Institute, Northern Technical University, Iraq

---

## Article Info

### Article history:

Received Mar 29, 2020

Revised Jun 2, 2020

Accepted Oct 7, 2020

### Keywords:

Bayesian risk

Distributed object decision

Distributed fusion sensor

Grey wolf optimization

Wireless sensor network

---

## ABSTRACT

The distributed object decision (DOD) was applied to choose a single solution for problem among many complexes solutions. Most of DOD systems depend on traditional technique like small form factor optical (SFFO) method and scalable and oriented fast-based local features (SOFF) method. These two methods were statistically complex and depended to an initial value. In this paper proposed new optimal technical called gray wolf optimization (GWO) which is used to determine threshold of sensor decision rules from fusion center. The new algorithm gave better performance for fusion rule than numerical results. The results are providing to demonstrate of fusion system reduced of bayes risk by a high rate of 15%-20%. This algorithm also does not depend on the initial values and shows the degree of complexity is better than other algorithms.

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*



---

## Corresponding Author:

Omar Ibrahim Alsaif

Mosul Technical Institute

Northern Technical University

Mosul, Iraq

Email: omar.alsaif@ntu.edu.iq

---

## 1. INTRODUCTION

Wireless sensor network (WSN) consists of collection of distributed independent sensors (or sensor nodes) which distributed randomly [1]. These sensors are independence similar, each one has transducer, power supply, microcomputer and transceiver to record and process sensory data. These sensors gather data and send them to fusion center in order to monitor some physical or environmental phenomenon [2]. The distributed object detection consists of a set of sensor nodes spread randomly. The nodes receive the information about nature based on its opinion states; the sensor node chooses one of possible messages and sends it to the fusion center via it's devoted channel. The fusion center collect the received messages from all sensors and estimate of the state by choosing one of the possible hypotheses; simple decentralized detection is shown in Figure 1.

Distributed decision making process which involves reaching a single solution for problem among a few complexes alone to solve it. The fundamental goal of distributed object detection system is to get high generalization ability for application of sensor networks environment [3]. It applied in a lot of application because it provides fine bandwidth. It is used in many fields (military and civilian) for wireless sensor networks such as sonar, radar and IFF (which is widely used in modern measurement technology). The advantages of decision-making are using narrow communication bandwidth, high reliability, low cost and easy to implement. The system of sensors decides the fusion center from threshold decision which focused on distributed parallel detection system [4].

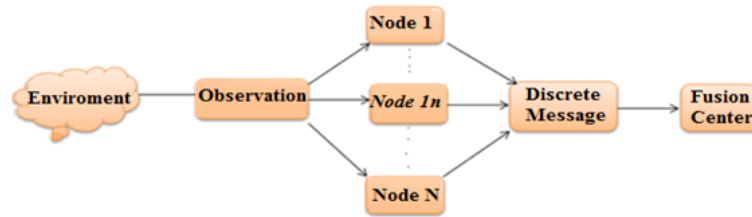


Figure 1. Represent the decentralized detection framework

Consider network content a multiple-sensor decision with fusion problem. A set of sensors randomly collected and transfer these local decisions through channels to fusion's channel center. Each of the peripheral sensors sends their information (using a transmission function) to a fusion center. All these local decisions are collected and treated at the fusion center where a final decision is made concerning the original inference problem, e.g.... content or not content of target [5]. The features of data fusion system are as follows:

- The network topology was designed so as not have any global information from the node sensor.
- There is no any global information for sensor nodes in these types of network topology.
- Communication of node must be kept on a strictly, it not allow node to broadcast results.
- Nodes must only know about connections in their neighborhoods nodes [6].

Fix the decision rules of the fusion center to optimize the sensor's decision threshold (SOFF algorithm). The literature [7-10] has conducted in-depth research, but this kind of algorithm is still a suboptimal method, computational complexity, and the results strongly depend on choice of initial value. The important research issue of the distributed parallel detection system is decision-making for fusion center. The main methods are presented as follow:

- Exhaustive algorithms were adopted to solve good performance of these methods but the efficiency was too low and not suitable for real-time systems.
- Algorithms used fixed decision threshold for each sensor, optimize the integration of the rules (SFFO algorithm) fusion center. One of the most representatives is the maximum a posteriori (MAP) method applied to the estimate of time-varying signals with collected noisy observations of a distributed nature in a sensor network [7].
- Algorithms aims to simulated hardening algorithm such as a genetic algorithm to solve MAP algorithm which solves the problem in the prior information has unknown conditions [8-10].

All these above methods have disadvantages such as the MAP algorithm which assumes that the each sensors have known decision threshold, not coupled with each other and unable to achieve the optimal system solution (minimum cost of overall bayesian system) [11]. The SFFO algorithm is suboptimal fixed decision rule fusion centers, while SOFF algorithm is optimizing the decision threshold sensor. For these two algorithm lateral have conducted in-depth research in but still kind of algorithm suboptimal method, computational complexity, and the results are strongly dependent on the initial value of the selection [12-14].

This paper is a detailed study of existing fusion system optimization decision rules proposed by GWO and is one of swarm intelligence method's algorithm which is stirred from the leadership hierarchy of grey wolves in nature for hunting mechanism. The algorithm works in two stages, the initial stage of the optimization the individuals should be fortified to scatter throughout the whole search space. In other words, they should try to explore the whole search space instead of clustering around local minima. In the second stage, the individuals have to exploit information gathered to converge on global minimum. GWO is combination of local rules and decision rules, and then the choice of global optimization algorithm decision-making for fusion center [15].

This paper is organized as follows: Section 2 provides description of the problem, how optimization the bayes risk and presents a detailed review on GWO. Section 3 includes the results and evaluation built on the simulation and Section 4 presents the conclusion and future direction for research path.

## 2. DISTRIBUTED FUSION SYSTEM OPTIMIZATION (DFSO)

### 2.1. Problem definition

Assume the environment of target tracking system includes  $M$  dynamic sensor nodes and each sensor is equipped with features. The DFSSO system for this paper includes fusion system structure consisting  $Y_k$  target state and  $n$  sensor node as illustrated in Figure 2, at time  $k$ , the  $Y_k$  and the  $i$ -th local sensors make local decisions  $u_i$ , according to decision rules for fusion center  $A$  is global decision  $D$ . Each sensor transmits

all of its comment to the fusion center. The fusion center *A* has solved a problem and decides on one of the *M* sensors information based on the messages it has received.

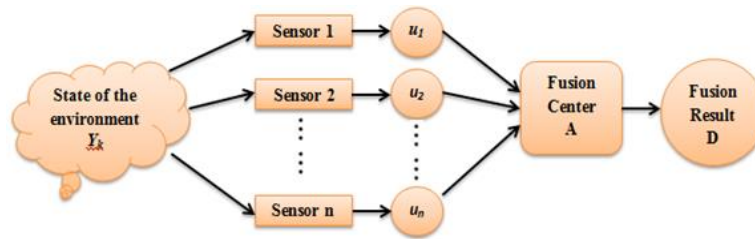


Figure 2. System composition structure diagram

The issue presented in this paper described in order to determine the optimum threshold sensor resolution. The system uses bayes risk to detect the minimum objective namely (*I*) in the function of the decision class. All possible decisions set the function in the presence of *r\** such that:

$$R_B(r^*) = \inf R_B(r) \tag{1}$$

bayes risk a public decision making optimal principle for probability of the potential outcomes state is known as *priori*. It is computed as the sum of the expected costs of every outcome, which is equal to the cost of each outcome multiplied by the probability of that outcome happening. The general binary equation of bayes risk detection equation is:

$$R_B = \sum_{i=0}^1 \sum_{j=0}^1 C_{ij} P_{(H_i|D_j)} = C_{00} P_{(H0)} P_{(D0|H0)} + C_{10} P_{(H1)} P_{(D0|H1)} + C_{10} P_{(H0)} P_{(D1|H0)} + C_{11} P_{(H1)} P_{(D1|H1)} \tag{2}$$

Where: *C<sub>ij</sub>* is the cost of judging *i* when *j* is true. The principle of conditional probability density can be defined as follows:

$$R_B = \int [P_{(H0)}(C_{01} - C_{11}) \left( P_{(u|H1)} - P_{(H0)}(C_{10} - C_{00}) P_{(u|H0)} \right)] dz + C_{00} P_{(H0)} + C_{11} P_{(H0)} \tag{3}$$

The simplest optimization algorithm is exhaustive. The basic idea thorough fusion of all rules of the system to calculate bayesian risk for all rules have to select the optimal rule (*r\**). Although this method can find most excellent solution but a large amount of computation will apperar, especially when the system is more complex. Secondly a linear iterative method is fixed the part of the rules of judgment and seeking another part of the rules of judgment. It continues repeated iteration until ultimately can solve the system. The optimal rule of judgment does not exhaust all the integration system rules, and the algorithm complexity is greatly reduced, as illustre in the following steps [16]:

1. The center fusion rules known  $\gamma_0$  and other sensors decision threshold  $Th_f(1,2,..,N, j \neq k)$  then the decision threshold of system is the smaller sensor *k* detected by bayesian risky, as illustrated by (4) [17]:

$$Th_k = \frac{\sum_{u^k} C_f A_{(u^k)} \prod_{i=1, i \neq j}^N P_{(u_i|H_0)}}{\sum_{u^k} C_d A_{(u^k)} \prod_{i=1, i \neq j}^N P_{(u_i|H_1)}} \tag{4}$$

- Where  $P_{(y_k|H_1)} > Th_k$  and  $P_{(y_k|H_0)} < Th_k$ ;  $C_f = P_{(H1)} (C_{01} - C_{11})$ ;  $C_d = P_{(H1)} (C_{10} - C_{00})$
2. When the system detects all sensors are sending their data, the decision rules of each sensors are determined. The system used bayesian risk to choose  $\gamma_0$  which minimizes fusion center of the fusion rule as determined in (5) [10]:

$$\left. \begin{aligned} P\left(\frac{u}{H_1}\right) &> C_f \text{ if } D = 1 \\ P\left(\frac{u}{H_1}\right) &< C_d \text{ if } D = 1 \end{aligned} \right\} \tag{5}$$

Equations (4-5) show that the center of fixed decision rule fusion is rectifies sensor optimal decision threshold and vice versa. Using Iterative approach optimizes the system to the final judgment rule. The other feature of (4-5) are complex and have slow convergence, while on threshold they are coupled to each other to optimize results strongly dependent on the choice initial value. Taking into account the convergence rate and the accuracy of the solution, the GWO algorithm and climb the algorithm of mountain variation (an improved algorithm of genetic algorithm) is used to solve the problem to achieve better results.

**2.2. Grey wolf optimization (GWO)**

GWO algorithm is a new optimization method employed to solve optimization problems of different types like other heuristic algorithms in the area of evolutionary computation. GWO doesn't have require the gradient of the function in its optimization process [15]. Gray wolf is the top predator, at the top of the food chain, its way of life. Most of the main group (5-12) wolves, built. Gray wolf population pyramid level, and has a strict hierarchy management system, as shown in Figure 3.

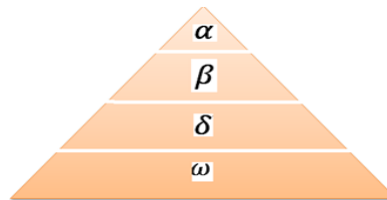


Figure 3. Hierarchy of gray wolf

1. The first level called alphas ( $\alpha$ ) are leaders (the dominant wolf). Alpha is making decisions about activities of hunting, sleeping place, time to wake, and so on. The alpha's decisions are dictated to the pack.
2. The second level called beta ( $\beta$ ). Betas are helping  $\alpha$  in decision-making and other pack activities. It plays the role of a consultant to the alpha and discipliner for the pack.
3. The third level of the pyramid is  $\delta$ , they listen to  $\alpha$  and  $\beta$  instructions, but can also refer to Play the other bottom individuals, is mainly responsible for reconnaissance, sentry, hunting, nursing and other services.
4. The lowest level of the pyramid, called  $\omega$ , is primarily responsible for balancing the internal relationships of the population and taking care of young wolf affairs. They are the last wolves that are allowed to eat. The crucial role for gray wolf's population level to play in the realization of the group is effectively kill prey. Team mode search, track, close to the prey, and then surround the prey from all directions, When the encirclement is small enough and perfected, the wolves are closest to their prey under the command of vile  $\beta$ ,  $\delta$  expand the attack, escape in the prey, the rest of the individual supplies, to achieve flock of wolves encircling the change of movement, so that the prey continue to implement all directions. In order to simulate the hunting behavior of grey wolves mathematically, it is always assumed that  $\alpha$ ,  $\beta$ , and  $\delta$  have best knowledge, as in following equations [15].

$$\begin{aligned} \vec{D}_\alpha &= |C_1 - \vec{X}|, \\ \vec{D}_\beta &= |C_2 - \vec{X}|, \\ \vec{D}_\delta &= |C_3 - \vec{X}|, \end{aligned} \tag{6}$$

$$\begin{aligned} \vec{X}_1 &= \vec{X}_\alpha - \vec{A}_1 \cdot (\vec{D}_\alpha), \\ \vec{X}_2 &= \vec{X}_\beta - \vec{A}_2 \cdot (\vec{D}_\beta), \\ \vec{X}_3 &= \vec{X}_\delta - \vec{A}_3 \cdot (\vec{D}_\delta), \end{aligned} \tag{7}$$

$$\vec{X}(t + 1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{8}$$

Where:  $(\vec{X}_\alpha, \vec{X}_\beta, \vec{X}_\delta)$  represent positions of alpha, beta and delta respectively.  $(C^1, C^2, C^3)$  and  $(A^1, A^2, A^3)$  are all random vectors,  $X$  is the position of the current solution,  $t$  indicates the number of iterations.

The "A" is an arbitrary value in the gap  $[-2a, 2a]$ . When  $|A| < 1$ , the wolves are forced to attack the prey. Attacking the prey is the exploitation ability and searching for prey is the exploration ability. The random values of "A" are utilized to force the search agent to move away from the prey. When  $|A| > 1$ , the grey wolves are enforced to diverge from the prey. The final position would be in a random position within a circle which is defined by the positions of alpha, beta, and delta in the search space. In other words alpha, beta, and delta estimate the victim position and other wolves update their positions randomly around the victim. The main algorithm for GWO are illustrated in Figure 4 [18].

```

Initialize the gray wolf population  $X_i (i=1,2,\dots,n)$ 
Initialize a, A and C
Calculate the fitness of each search agent
 $X_\alpha$  = the best search agent.
 $X_\beta$  = the second best search agent.
 $X_\delta$  = the second best search agent
While (t < Max. number of iterations)
  For each search agent
    Update the position of current search agent by (8)
  End for
  Update a, A and C
  Calculate the fitness of all search agents.
  Update  $X_\alpha$ ,  $X_\beta$ , and  $X_\delta$ 
  t=t+1
End while
Return  $X_\alpha$ 

```

Figure 4. GWO algorithm

### 2.3. Distributed fusion with GWO

It is easy to implement the system rules of choices. The specific steps should be followed as below:

1. The sensor's of threshold will be  $\{Th_{1(t)}^k, Th_{1(t)}^k, \dots, Th_{N(t)}^k\}$  respectively, with the word length of  $M$  binary code string  $\{0, b_M, b_{M-1}, \dots, b_2 b_1\}$ . Where  $b_j \in \{0, 1\}$ ,  $1, 2, j \in j = N$ .
2. Draw a directed graph  $G$  and define it  $G = (C, V)$ , where top points set to:

$$C = \left\{ c_0(v_s), c_1(v_M^0), c_2(v_M^1), c_3(v_{M-1}^0), c_4(v_{M-1}^1), \dots, c_{2M-3}(v_2^0), c_{2M-2}(v_2^1), c_{2M-1}(v_1^0), c_{2M}(v_1^1) \right\} \quad (9)$$

The vector collection can be presented as:

$$V = \left\{ (v_s, v_M^0), (v_s, v_M^1), (v_M^0, v_{M-1}^1), (v_M^0, v_M^1), (v_M^1, v_{M-1}^0), (v_M^1, v_{M-1}^1), \dots, (v_2^0, v_1^0), (v_2^0, v_1^1), (v_2^1, v_1^0), (v_2^1, v_1^1) \right\} \quad (10)$$

Where:  $v$  is the initial vertex. Vertex  $(v^0$  and  $v^1)$  denote binary code string  $b_j$  value of 0 and 1 state [19].

3. Consider the number of wolfs is  $q$ . they are starting from vertex  $v_s$ , at  $t$  time. the wolf  $\alpha$ ,  $\beta$  and  $\delta$  are selected a vertex  $v_j^i$ , where  $i = 0, 1$  and consider the fittest solution for value of alpha( $\alpha$ ) wolfs is  $k$  to choose  $v_j^i$ , the probability illustrated in (11) [20]:

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{[\tau_{0j}(t)]^\alpha [\eta_{0j}(t)]^\beta + [\tau_{1j}(t)]^\alpha [\eta_{1j}(t)]^\beta} & i = 0, 1 \\ 0 & \text{other} \end{cases} \quad (11)$$

Where  $\tau_{ij}(t)^\alpha$  and  $[\eta_{ij}(t)]^\beta$  represents probability of solution of  $\alpha$  and  $\beta$  represently.

4. The vertices  $(v_j^{i(k)}(t))$  selected by wolf composed of binary code string  $\{0, v_M v_{M-1} \dots \dots v_2 v_1\}$  parameter maps can be obtained at time t wolfs groups (k) find the parameter  $\{Th_{1(t)}^k, Th_{2(t)}^k, \dots, Th_{N(t)}^k\}$  and then by the (9) can be calculated  $(Y_{0(t)}^k)$ .
5. The parameters  $\{Y_{0(t)}^k, Th_{1(t)}^k, Th_{2(t)}^k, \dots, Th_{N(t)}^k\}$  into (3) are available. The wolfs find solutions from bayesian risk  $R_{B(t)}^k$ .
6. Find the smallest value of bayesian risk  $R^*(t)$  in the  $t^{th}$  search period. This value is searched by wolfs cyclic  $D$  and  $A$  is fluctuation range decrease  $\alpha$ . It allows its search agents to update their position dynamically based on the location of the  $(\alpha, \beta,$  and  $\delta)$  and attack towards the prey and give best convergence of this by (6-8).
7. If the position of prey is changed, the fitness value is calculated as new prey's position as [21]:

$$X_p^{t+1} = v_j^{i(w\alpha)}(t + 1) \overline{X_\alpha^{t+1}} + v_j^{i(w\beta)}(t + 1) \overline{X_\beta^{t+1}} + v_j^{i(w\delta)}(t + 1) \overline{X_\delta^{t+1}} \tag{12}$$

Where:

$$v_j^{i(w\alpha)}(t + 1) = \frac{v_j^{i(w\alpha)}(t)}{v_j^{i(w\alpha)}(t) + v_j^{i(w\beta)}(t) + v_j^{i(w\delta)}(t)}$$

$$v_j^{i(w\beta)}(t + 1) = \frac{v_j^{i(w\beta)}(t)}{v_j^{i(w\alpha)}(t) + v_j^{i(w\beta)}(t) + v_j^{i(w\delta)}(t)} \tag{13}$$

$$v_j^{i(w\delta)}(t + 1) = \frac{v_j^{i(w\delta)}(t)}{v_j^{i(w\alpha)}(t) + v_j^{i(w\beta)}(t) + v_j^{i(w\delta)}(t)}$$

8. Determine whether the preset number of iterations  $n_c$  match the output, Then consider as an optimal solution, otherwise go to step 3.

**3. EXPERIMENTAL ANALYSIS**

Verification of the proposed variation for GWO to distribute headings is standard test system optimization capabilities. The following test are parallel detection system consisting of two sensors as shown in Figure 5,

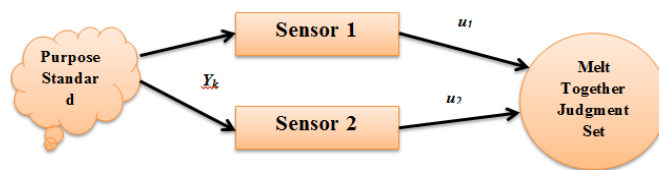


Figure 5. Detecting system model

Where time (k), for target state is  $Y_k$ , the value of  $Y_k$  can be 0 or 1 (binary check test), the prior probability is  $P_0$  ( $P_i$  indicates that the target state  $Y_k$  is  $i$ ) Probability value calculated from ( $P_0 + P_1 = 1$ ). The observation of the sensor obeys the Gaussian distribution with the conditional probability density.

$$R_{Y1/H1(\frac{y1}{H1})} = \frac{1}{\sqrt{2\pi\sigma_i}} \exp\left(-\frac{(y_i - m_i)^2}{2\sigma_i^2}\right) \tag{14}$$

$$R_{Y1/H0(\frac{y1}{H0})} = \frac{1}{\sqrt{2\pi\sigma_0}} \exp\left(-\frac{y_i^2}{2\sigma_0^2}\right) \tag{15}$$

The following algorithms are used to find financial decision for the fusion systems:

1. Algorithm 1 is used SOOF with fixed sensor [8], Limit threshold  $Th_1 = Th_2 = 1$ .
2. Algorithm 2 is used SFFO with integration of the center or rules. Algorithm 3 is used brute-force with iterative method, in order to reduce the amount of calculation, it is assumed  $Th_{1(0)} = Th_{2(0)} = 1.5$ .
3. Apply GWO algorithm with two stage
  - First stage, dim (population size) =30, rang= [-30 30], Number of alteration= 100 and initial value of thresholds ( $Th_1, Th_2$ ) generated randomly.
  - Second stage number of alteration=200
4. Use GWO, algorithm parameters: dim=30, rang= [-100 100], Number of alteration= 500 and, the initial value  $Th_1, Th_2$  with locally generated.
5. To compare among five algorithms, algorithms 1 and 3 are fixed threshold. In the applied step the algorithms 2, 4 and 5 the threshold are calculated. The total number of iterations is 100 times except the second stage of algorithm 4 iterations is 200 times, and algorithm 5 equal 500 alteration under different conditions to  $P_0$  with bayes risk, the results shown in Table 1.

Table 1. Show the bayesian risk with algorithm

Probability	bayesian risk					
	Algorithm 1 SOOF	Algorithm 2 SFFO	Algorithm 3 brute-force	Algorithm 4a GWO(1)	Algorithm 4a GWO(2)	Algorithm 5 GWO(3)
0.1	0.5	0.6	0.443	0.09	0.09	0.1
0.2	0.54	0.67	0.57	0.148	0.148	0.15
0.3	0.59	0.64	0.525	0.173	0.18	0.23
0.4	0.62	0.665	0.556	0.212	0.235	0.264
0.5	0.65	0.81	0.61	0.249	0.26	0.28
0.6	0.67	0.758	0.651	0.261	0.261	0.265
0.7	0.62	0.725	0.681	0.231	0.235	0.251
0.8	0.57	0.666	0.721	0.163	0.171	0.182
0.9	0.52	0.62	0.742	0.08	0.148	0.138

Figure 6 shows the bayesian risk of fusion system in algorithm - algorithm 3 while Figure 7 shown the bayesian risk of algorithm 4 and 5.

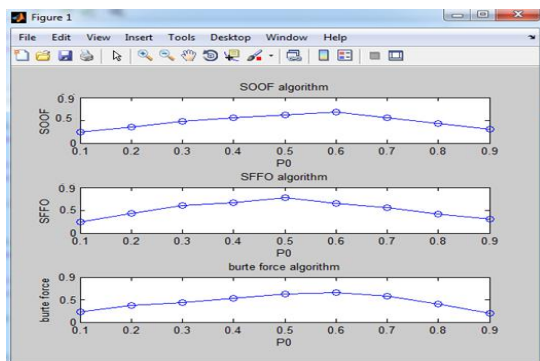


Figure 6. Bayesian risk of fusion system in algorithms (1 and 3).

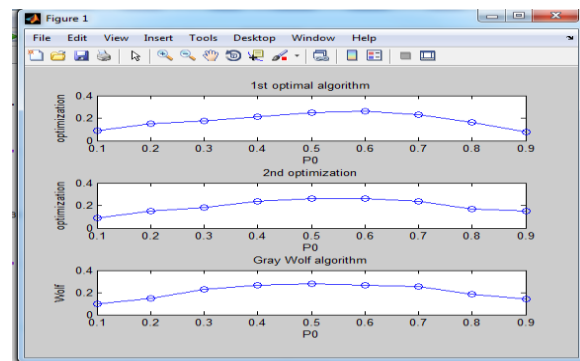


Figure 7. Bayesian risk of fusion system for algorithms (4 and 6).

As can be seen from Table 1 and illustrated in Figure 6 and Figure 7:

- When the value of the prior probability  $P_0$  is greater than (0.5), algorithm 1 and algorithm 2 have a bayesian risk significantly higher than other algorithms. At ( $P_0=0.6$ ), algorithm 3 has lower bayes risk from algorithm 1 and algorithm 2.
- The value of  $P_0$  in algorithm 3 equal (0.4 -0.8), less than the algorithm 2 about 20%.  $P_0$  values are (0.5 to 0.9), slightly higher than other methods about (5-10%), which shows the fixed sensor threshold will be in a certain way.
- The degree of influence on the system optimization performance is less than the fixed fusion heart rules. Through the above three kinds of algorithms can be seen, bayes risk in algorithm 4 is the lowest, less than 15% to 20% of other algorithms (1, 2, and 3).

- As can be seen from Figure 4, the number of iterations is 100 times, bayesian risk is significantly lower than the improved GWO (when alteration equal 500), which represent (15%-20%) from other algorithms.
- In addition, from the complexity of the algorithm view, the complexity of algorithm 1 is  $O(n_c \times qm^2n^2)$ , lower than the complexity of algorithm 3 which equal  $O(2^{2N})$  and  $O(n_c * 2^N)$  for algorithm 2, while algorithm 5 is comparable higher than the complexity of algorithm 1.

#### 4. CONCLUSION

This paper is development of optimal algorithm for distributed fusion decision, because all optimization methods are limited capacity due to independent initial calculation values according to a complex and intense fusion results. This method presents gray wolf optimization (GWO) which gives high performance optimization over than SOFF and SFFO algorithms. In experiments results, the complexity degree of GWO is lower than most of the current SFFO & SFFO algorithms. It is also found that this algorithm is better than other algorithms in optimizing performance. In addition through experiment also found the sensor threshold optimization methods is better than other fixed threshold sensors. The perform of GWO algorithm is good when alteration equation (500), which is more suitable for the performance requirements of the application. In future work, it is advised to use multi-objective grey wolf optimizer (MOGWO) for distributed fusion decision for wireless sensor network to give many solution representation for the trade-offs objective functions.

#### REFERENCES

- [1] R. V. Kulkarni, A. Förster, and G. K. Venayag, "Computational intelligence in wireless sensor networks: a survey, Communications Surveys & Tutorials", *IEEE*, 13 (1), pp: 68-96, 2011.
- [2] Kaur Ranjit and Arora Sankalap, "Nature Inspired Range Based Wireless Sensor Node Localization Algorithms" *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 4, No.6, 2016.
- [3] Gedas Bertasius, et al., "First-Person Action-Object Detection with EgoNet" Conditional random fields as recurrent neural networks" *In International Conference on Computer Vision (ICCV)*, 2015.
- [4] Helin Tapio and Burger Martin "Maximum a Posteriori Probability Estimates in Infinite-Dimensional Bayesian Inverse Problems" *arXiv: 1412.5816v3 [math.ST]* 2015.
- [5] Ruixin Niu, et al., "Decision Fusion Rules in Wireless Sensor Networks Using Fading Channel Statistics", *Conference on Information Sciences and Systems*, The Johns Hopkins University 2013.
- [6] Jakubiec Felicia Y and Ribeiro Alejandro, "D-MAP: Distributed Maximum a Posteriori Probability Estimation of Dynamic Systems," *IEEE transactions on signal processing*, Vol. 61, No. 2, January 15, 2013.
- [7] C. L. Ma, N. Liu and Y. Ruan, "A Dynamic and Energy-Efficient Clustering Algorithm in Large-Scale Mobile Sensor Networks," *International Journal of Distributed Sensor Networks*, vol 9, no. 11, pp. 1-8, 2013.
- [8] Kim Tae-Jung, et al., "Fast Mode Decision Algorithm Based on Adaptive Search Direction for Combined Scalability in Scalable Video Coding," *Advances in Computer Science and Its Applications*, 279, 1 DOI: 10.1007/978-3-642-41674-3\_1, Springer-Verlag Berlin Heidelberg 2014.
- [9] Wang Chunhua and Han Dong, "Data Mining Technology Based on Bayesian Network Structure Applied in Learning," *International Journal of Database Theory and Application* Vol.9, No.5 (2016), pp.267-274 <http://dx.doi.org/10.14257/ijda.2016.9.5.27>, 2016.
- [10] Chamberl and ean-F. and Veeravalli V. Venugopal, "Decentralized Detection in Sensor Networks" *IEEE transactions on signal processing*, Vol. 51, No. 2, Feb, 2003.
- [11] Ray P. and Varshney P. K. "Distributed Detection in Wireless Sensor Networks Using Dynamic Sensor Thresholds" *International Journal of Distributed Sensor Networks*, Taylor & Francis Group, LLC ISSN: 1550-1329 print / 1550-1477 online DOI: 10.1080/ 15501320701774659, 2008.
- [12] Essa Ibrahim Essa Al-juborie and Khalil IA Al-saif, "Optical Ring Architecture Using 4-Nodes WDM Add/Drop Multiplexer Based SSMFD", *Advanced Research in Electrical and Electronic Engineering*. Volume 2, Number 1 October-December, pp. 66-68. 2014.
- [13] K. Tsianos and M. Rabbat, "Fast decentralized averaging via multi-scale gossip," *Proc. IEEE Distributed Computing in Sensor Systems*, 2010.
- [14] Lauer G. S. and Sandell N. R. "Distributed detection with waveform observations: Correlated observation Processes," *Proceedings of the 1981 American Controls Conference*. Arlington, Virginia: [s.n.], 1981, 2: 812-819.
- [15] Mittal Nitin, et al. "Modified Grey Wolf Optimizer for Global Engineering Optimization" *Hindawi Publishing Corporation Applied Computational Intelligence and So Computing Volume Article ID 7950348*, 16 pages <http://dx.doi.org/ 10.1155/7950348>, 2016.
- [16] DQ Zeebaree, H Haron, AM Abdulazeez. "Gene selection and classification of microarray data using convolutional neural network". *IEEE, International Conference on Advanced Science and Engineering (ICOASE)*. Pages 145-150, 2018.
- [17] Ray P. and Varshney P. K. " Distributed Detection in Wireless Sensor Networks Using Dynamic Sensor Thresholds" *International Journal of Distributed Sensor Networks*, 4: 4-11, Taylor & Francis Group, LLC ISSN: 1550-1329 print / 1550-1477 online DOI: 10.1080, 2008.



- [18] Nitish Chopra, *et al.*, “Hybrid GWO-PSO Algorithm for Solving Convex Economic Load Dispatch Problem” *International Journal of Research in Advent Technology*, Vol.4, No.6, 2016, Available online at [www.ijrat.org](http://www.ijrat.org).
- [19] Luo Junhai and Liu Zuoting, “Serial distributed detection for wireless sensor networks with sensor failure” *EURASIP Journal on Wireless Communications and Networking* DOI 10.1186/s13638-017-0911-6, Feb. 2017.
- [20] I. A. Saleh, Omar I. Alsaif, S. A. Muhamed, E. I. Essa. “Task scheduling for cloud computing Based on Firefly Algorithm”. *Journal of Physics: Conference Series*, Vol, 1294, Issue 4, 2019.
- [21] Zhao X Q, et al.” Energy-Efficient Routing Protocol for Wireless Sensor Networks Based on Improved Grey Wolf Optimizer” *KSII Transactions on Internet and Information Systems*, 2018, 12(6): 2644-2657.

## BIOGRAPHIES OF AUTHORS



Ibrahim Ahmed Saleh: born in Mosul-Iraq in 1963. He received his MSc. degree (in signal and image processing from the University of Mosul, Iraq in 2003 and in 2013 he received his PhD in artificial techniques and computer networking from Mosul University. He became assistant professor in 2016. From 1997 to 2005, he worked at computer center in Mosul University/ Iraq. Currently he is lecturer at the Dept. of Software Engineering, College of Computer Sciences & Math, University of Mosul, IRAQ.



Omar Ibrahim Alsaif: He received his B.Sc in electrical engineering from University of Mosul, Iraq, in 1992. He received his M.Sc. and Ph.D degrees in electrical and microelectronic engineering from Mosul University, in 2005 and 2018 respectively. He is currently an lecturer in the Mosul Technical Institute/ Northern Technical University In Mosul, Iraq. His research interests include microelectronic, DSP, renewable energy and solid state devices.



Maan A. Yahya: holds a Ph. D. degree in Digital Signal Processing and Communication, a M.Sc degree in Image processing, B.Sc degree in computer science form University of Mosul, Iraq. He is a lecturer in Technical Institute-Ninawa, Northern Technical University. E-mail: [dr.maan@ntu.edu.iq](mailto:dr.maan@ntu.edu.iq).