Journal of
**Mathematics and**
**Informatics**

# A New Algorithm Based on the Improved Artificial Fish Swarm for Learning Thresholds in TDRS Model

*Qiang Zhang*[1] and *De-you Xia*[2]

School of Science, Chongqing University of Posts and Telecommunications
Chongqing-400065, Chongqing, China.
[1]E-mail:13608363654@163.com; [2]E-mail:xiadeyou12@163.com

***Abstract.*** The three-way decision-theoretic rough set model needs the loss function in advance and requires the knowledge and experience of experts. This hinders the application of the model in practice. To solve this problem, based on the artificial fish swarm method, the process of optimization is carried out by dynamically adjusting the step and visual. Then the rules of the fish behavior are improved on the basis of the optimization problem of the three-way decision-theoretic rough set model. The experiments show that this algorithm has achieved a faster run-time and smaller cost than the artificial fish swarm algorithm and the adaptive learning algorithm.

## 1. Introduction
Rough set model, was proposed by Pawlak [1] in 1982, which is used to deal with the uncertain problems [2]. As a theory of data analysis and processing, it has been widely applied in data mining, machine learning, internet of things, cloud computing and so on [3-6]. It is also an important intelligent information processing technology [7-8]. As rough set model was too strict and lack of fault-tolerance ability, decision-theoretic rough set model was proposed by Yao [9] in 1990 which had been summarized to calculate the threshold of decision-theoretic rough set model under the loss function. In [10], Yao redefined the semantics interpretation of three-way decision-theoretic rough set (TDRS) model.

In three-way decision-theoretic rough set model, each action has to take the corresponding loss. How to minimize the decision cost is one of the most important issues [11]. Jia [12-13] analyzed the relationship between loss function and threshold in three-way decision-theoretic rough set model. He also presented an optimum problem and proposed an adaptive learning algorithm which took much time and cost a lot when it calculated threshold in an experiment. Hu [14] proposed an artificial fish swarm algorithm to automatically determine thresholds in three-way decision-theoretic rough set model which didn't take into account the impact of capacity and accuracy. In this paper, based on the artificial fish swarm method which adjusted the step and visual, we improved the rules of the fish behavior action. It was a new method to learn thresholds in

three-way decision-theoretic rough set model which had achieved a faster run-time and smaller cost than the artificial fish swarm algorithm (AFSA) [12] and the adaptive learning algorithm (ALA) [14].

## 2. Preliminaries
### 2.1. Three-way decision-theoretic rough set model

This section introduces the basic concepts of decision-theoretic rough set model [15-17]. Given an information system $M = (U, At, V, f)$, where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_s\}$ denotes the set has $s$ different states. $A = \{a_1, a_2, \ldots, a_n\}$ denotes $n$ different actions are taken, $p(\omega_i \mid x)$ is a conditional probability when the state of object $x$ is $\omega_i$, and $\lambda(a_j \mid \omega_i)$ is a loss function when $\omega_i$ takes action $a_j$. When $x$ takes action $a_j$, the expected risk defines as follows:

$$R(a_j \mid x) = \sum_{i=1}^{s} \lambda(a_j \mid \omega_i) \cdot p(\omega_i \mid x) \tag{1}$$

In an approximation space, object $x$ is usually described by equivalence class $[x]$. $\Omega = \{X, X^c\}$ denotes whether object $x$ belongs to decision class $X$, $p(X \mid [x]) = \dfrac{|X \cap [x]|}{|[x]|}$ is the conditional probability when object $x$ belongs to $X$, and $p(X^c \mid [x]) = 1 - p(X \mid [x])$ is the conditional probability when object $x$ doesn't belong to $X$. If object $x$ belongs to $X$, $\lambda_{PP}, \lambda_{NP}, \lambda_{BP}$ denotes respectively the loss function when classifying an object into three regions: $POS(X), BNG(X), NEG(X)$. On the contrary, if object $x$ doesn't belong to $X$, $\lambda_{PN}, \lambda_{NN}, \lambda_{BN}$ denotes respectively the loss function when classifying an object into same regions.

Usually, if object $x$ belongs to $X$, the loss of dividing $x$ into positive region is less than or equal to boundary region, and they are less than the loss of negative region. If object $x$ doesn't belong to $X$, the loss of dividing $x$ into negative region is less than or equal to boundary region, and they are less than the loss of positive region. So supposing $\lambda_{PP} \le \lambda_{BP} < \lambda_{NP}$ and $\lambda_{NN} \le \lambda_{BP} < \lambda_{PN}$.

Let

$$\alpha = \frac{(\lambda_{PN} - \lambda_{BN})}{(\lambda_{PN} - \lambda_{BN}) + (\lambda_{BP} - \lambda_{PP})},$$

$$\gamma = \frac{(\lambda_{PN} - \lambda_{NN})}{(\lambda_{NP} - \lambda_{PP}) + (\lambda_{PN} - \lambda_{NN})}, \tag{2}$$

$$\beta = \frac{(\lambda_{BN} - \lambda_{NN})}{(\lambda_{BN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{BP})}.$$

According to the relationship of the loss functions, let $\alpha \in (0,1], \beta \in [0,1), \gamma \in (0,1)$. In this paper, only one case is discussed: $p(X \mid [x]) + p(X^c \mid [x]) = 1$, if it satisfies the relationship: $(\lambda_{PN} - \lambda_{BN})(\lambda_{NP} - \lambda_{BP}) > (\lambda_{BP} - \lambda_{PP})$, then: $\alpha > \gamma > \beta$.

The Bayesian decision procedure suggests the following rules:

（P):If $p(X \mid [x]) \ge \alpha$, then $x \in POS(X)$;

A New Algorithm Based on the Improved Artificial Fish Swarm for Learning Thresholds
in TDRS Model

（B): If $\beta < p(X \mid [x]) < \alpha$, then $x \in BND(X)$;

（N): If $p(X \mid [x]) \leq \beta$, then $x \in NEG(X)$.

## 2.2. Decision-making risk minimization problem

Jia [12] presented the decision-making risk minimization problem. Supposing there are only two decision classes, in the decision table $M : U = \{x_1, x_2, \ldots, x_n\}$，the conditional probability is $p_i$ when object $x_i$ belongs to $X$, which can be calculated by equivalence class. Let correct classification cost be 0, namely $\lambda_{PP} = \lambda_{NN} = 0$, so decision cost in the entire decision table [12] are shown as follow:

$$f = \sum_{x_i \in POS(X)} \lambda_{PN} \cdot (1 - p_i) + \sum_{x_j \in NEG(X)} \lambda_{NP} \cdot p_j + \sum_{x_k \in BND(X)} (\lambda_{BN} \cdot (1 - p_k) + \lambda_{BP} \cdot p_k) \tag{3}$$

According to formula (2), the thresholds [12] can be calculated by the six loss functions. Suppose $\lambda_{PP} = \lambda_{NN} = 0$, then $\alpha, \beta, \gamma$ and $\lambda_{PN}$ can be shown as follows:

$$\begin{cases} \lambda_{PN} = \lambda_{PN}; \\ \lambda_{NP} = \dfrac{1 - \gamma}{\gamma} \cdot \lambda_{PN}; \\ \lambda_{BN} = \dfrac{\beta \cdot (\alpha - \gamma)}{\gamma \cdot (\alpha - \beta)} \cdot \lambda_{PN}; \\ \lambda_{BP} = \dfrac{(1 - \alpha) \cdot (\gamma - \beta)}{\gamma \cdot (\alpha - \beta)} \cdot \lambda_{PN}. \end{cases} \tag{4}$$

Suppose $\lambda_{PN} = 1$, according to formula (4), formula (3) can be deduced as follows [12]:

$$f_{min} = \sum_{p_i \geq \alpha} (1 - p_i) + \sum_{p_j \leq \beta} \frac{1 - \gamma}{\gamma} \cdot p_j + \varepsilon \cdot \sum_{\beta < p_k < \alpha} [\frac{\beta \cdot (\alpha - \gamma)}{\gamma \cdot (\alpha - \beta)} \cdot (1 - p_k) + \frac{(1 - \alpha) \cdot (\gamma - \beta)}{\gamma \cdot (\alpha - \beta)} \cdot p_k] \tag{5}$$

In order to avoid more objects being divided into the boundary region, penalty factor $\varepsilon$ ($\varepsilon \geq 1$)is adopt, where $0 \leq \beta < \gamma < \alpha \leq 1$. Thus, the problem of solving the thresholds is transformed into an optimization problem.

## 3. IAFSA for learning thresholds in three-way decision-theoretic rough set model
### 3.1. Artificial fish swarm algorithm
Artificial fish swarm [18-21] can be described as follows: The state of artificial fish is indicated as vector $X = (x_1, x_2, \ldots, x_n)$ , and $x_i (i = 1, 2, \ldots, n)$ is the artificial fish. $Y = f(x)$ is the food concentration at the current position of the artificial fish. $d_{ij} = \| X_i - X_j \|$ is the distance of artificial fish. *Visual* is the range of view. *Step* is the step-length. $\delta$ is the congestion factor. And *Try_number* is the maximum number of each searching test.

### 3.2. Fish behavioral rules
1) Foraging behavior: The artificial fish $X_i$ selects the state $X_j$ randomly in its field of

version, in the problem of finding a minimum, if $Y_j < Y_i$ (when finding a maximum, $Y_j > Y_i$), it moves one step towards $X_j$. Otherwise, the artificial fish $X_i$ re-selects the state $X_j$. If it still cannot find the suitable location after $Try\_number$ times selecting, then moves randomly to a new state.

2) Bunching behavior: $N$ represents the number of partners of the artificial fish $X_i$ in its range of version, and $X_{center}$ represents the center position. If $Y_{center}N < \delta Y_i$, namely, it isn't crowd and retained less food, so moves one step towards $X_{center}$; otherwise foraging action will be taken.

3) Tailgating behavior: The artificial fish $X_i$ selects the best state $X_{best}$ in its field of version. If $Y_{best}N < \delta Y_i$, it doesn't cause congestion, so $X_i$ moves one step towards $X_{best}$; otherwise, foraging action will be taken.

4) Random behavior: The artificial fish $X_i$ selects the new state $X_j$ randomly in its field of version, then moves towards $X_j$;

5) Bulletin board: The status of the optimal artificial fish population is recorded. If the information is better than the bulletin board after each action, it replaces the above information. When the algorithm is finished, the value of bulletin board is output.

### 3.3. The improvement of artificial fish swarm algorithm
### 3.3.1. Improving visual and step

Li [19] analyzed that the visual had a great influence on the behavior and convergence performance in the algorithm. At the beginning, in order to speed up and avoid trapping in local optimum, each artificial fish is optimized by a large step and visual. Further, reducing step and visual can increase accuracy and speed up. According to [20], the step and visual can be adjusted dynamically by formula (6)

where $Step = Visual / 10$, $Visual_{max} = 2$, $Step_{max} = 0.5$, $t$ is the current iteration times, $T_{max}$ is the maximum iteration times. The visual and step is reduced dynamically. The function $a$ is incremented. The value of $s$ ranges from 1 to 20.

$$\begin{cases} Visual = Visual_{max} - Visual \cdot a \\ Step = Step_{max} - Step \cdot a \\ a = \exp(20 \times (t / T_{max})^s) \end{cases} \tag{6}$$

### 3.3.2. Improving fish behavior action

In foraging behavior, the artificial fish $X_i$ selects the state $X_j$ randomly in its field of version. If $Y_i < Y_j$, it can be moved directly to $X_j$. Conversely, it can re-select a new location $X_j$ randomly. If its condition isn't satisfied after $Try\_number$ times searching, and $X_i$ isn't the best state in its field of version, then $X_i$ moves randomly to a new state. If it is the best state, then retains. It can effectively reduce the operation time of the algorithm through the above improvements.

### 3.4. The new algorithm based on the IAFSA for learning thresholds in TDRS model
**Input:** Conditional probability value of each object.

A New Algorithm Based on the Improved Artificial Fish Swarm for Learning Thresholds
in TDRS Model

**Output:** Minimum risk cost $f_{\min}$ of the entire decision table and its corresponding threshold $\alpha', \beta', \gamma'$.

**Step1.** Initializing the parameter: artificial fish scale $D$ ,maximum iteration times $N$ ,visual, step, congestion factor $\delta$ and so on. The $(\alpha_i, \beta_i, \gamma_i)$ was generated randomly.

**Step 2.** Calculating the food concentration of each artificial fish and comparing with the state of bulletin board. If the information is better than the bulletin board, then replaces the above information.

**Step 3.** Calculating the visual and step by formula (6).

**Step 4.** $N$ represents the number of partners of the artificial fish $X_i$. $X_{center}$ represents the center position. If $Y_{center} N < \delta Y_i$ , it takes bunching action, then $X_i$ moves one step towards $X_{center}$ ; otherwise turn step 6.

**Step 5.** The artificial fish $X_i$ selects the best state $X_{best}$ . If $Y_{best} N < \delta Y_i$ , it takes tailgating action, then $X_i$ moves one step towards $X_{best}$ ; otherwise turn step 6.

**Step 6.** The artificial fish $X_i$ selects the state $X_j$ randomly. If $Y_j < Y_i$ , it takes foraging action, $X_i$ moves directly to $X_j$ ; otherwise, the artificial fish $X_i$ re-selects the state $X_j$ . If attempting $Try\_number$ times select, it also cannot find the suitable location which replaces the information in the bulletin board.

**Step 7.** Checking the termination condition. If $passed\_time \geq N$, turn step 8; Otherwise $passed\_time = passed\_time + 1$ , turn step2.

**Step 8.** The algorithm ends. Take out the minimum cost and its corresponding threshold in the bulletin board.

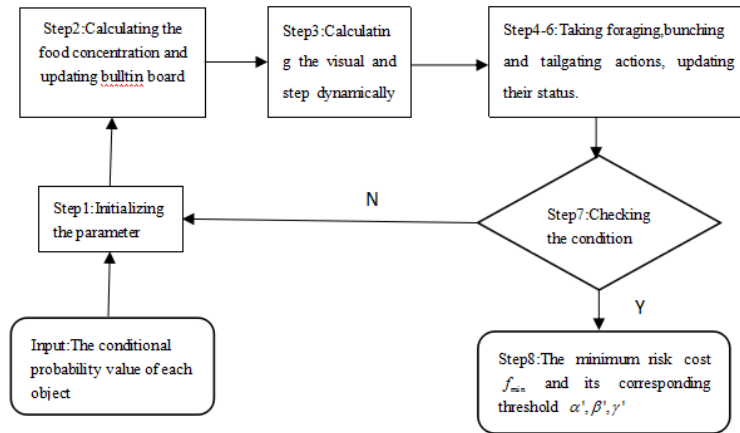The flow chart of the new algorithm is shown in Figure 1.



**Figure 1:** The flow chart of the new algorithm

## 4. Experimental result

In this section, the method in this paper is contrasted with the artificial fish swarm algorithm and the adaptive learning algorithm in the aspects of computing time and minimum cost. Because IAFSA is a random algorithm, we run the data sets 10 times and take their average as the final result. Experimental operating environment: CPU IntelI3-230M, MHZ 3.40GHz, RAM 4G，32 bites Windows7 system, R language. Experimental data: 14 data sets from UCI.

**Table 1:** Comparative analysis on computing time among three algorithms

| | running time/ms | | |
|---|---|---|---|
| | ALA | AFSA | IAFSA |
| wdbc | 6.98 | 1.12 | 0.51 |
| wpdc | 3.11 | 0.83 | 0.28 |
| monks-1 | 5.49 | 0.92 | 0.44 |
| monks-2 | 5.13 | 0.74 | 0.48 |
| monks-3 | 5.25 | 0.54 | 0.49 |
| transfusion | 12.41 | 0.95 | 0.81 |
| credit | 8.31 | 0.74 | 0.54 |
| innosphere | 4.13 | 0.58 | 0.41 |
| bands | 5.22 | 0.82 | 0.56 |
| hepatitis | 4.82 | 0.97 | 0.84 |
| musk | 3.86 | 0.94 | 0.61 |
| voting | 3.57 | 0.81 | 0.45 |
| agaricus | 47.85 | 1.76 | 0.73 |
| bank | 563.28 | 1.94 | 0.86 |

Comparative analysis on computing time(Time units:ms) among three algorithms is shown in Table 1. From figure 2, the IAFSA runs faster than the other two algorithms in most data sets: wdbc, wpbc, monks-1, monks-2 and voting. The adaptive learning algorithm is an iterative algorithm. Its time complexity is $O(n^2)$, and its run-time increases as the increasing samples. IAFSA is a stochastic optimization algorithm, its running time doesn't increase obviously.
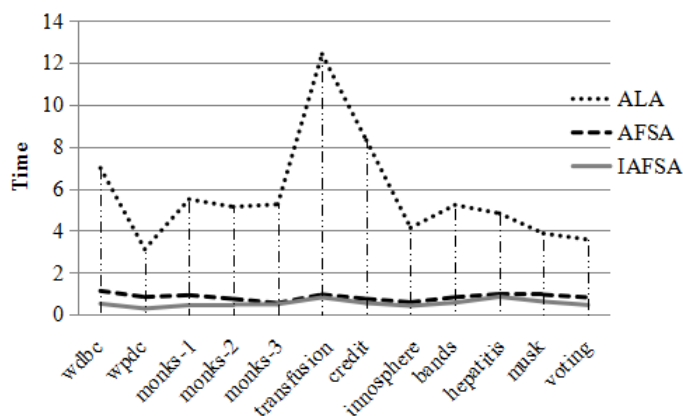


**Figure 2:** Comparative analysis on computing time among three algorithms in different data sets

A New Algorithm Based on the Improved Artificial Fish Swarm for Learning Thresholds
in TDRS Model

**Table 2:** Comparative minimum cost among three algorithms

|  | ALA | AFSA | IAFSA |
|---|---|---|---|
| wdbc | 1.32 | 0.83 | 0.32 |
| wpdc | 5.24 | 0.96 | 0.41 |
| monks-1 | 38.12 | 1.12 | 0.82 |
| monks-2 | 6.19 | 1.68 | 1.06 |
| monks-3 | 0.35 | 0.3 | 0.29 |
| transfusion | 1.11 | 0.95 | 0.72 |
| credit | 1.12 | 0.87 | 0.64 |
| innosphere | 5.56 | 1.56 | 1.28 |
| bands | 40.27 | 1.27 | 0.96 |
| hepatitis | 4.23 | 0.88 | 0.39 |
| musk | 1.96 | 0.76 | 0.74 |
| voting | 19.54 | 0.86 | 0.78 |
| agaricus | 189.35 | 1.78 | 1.56 |
| bank | 123.45 | 1.43 | 0.97 |

Comparative minimum cost among three algorithms is shown in Table 2. From figure 3, IAFSA gets the smaller cost in most data sets. The further discussion is necessary. It is immature to verify the validity of the threshold currently. Considering other evaluation factors and judging the classification results are important. We will do further research in the next work.
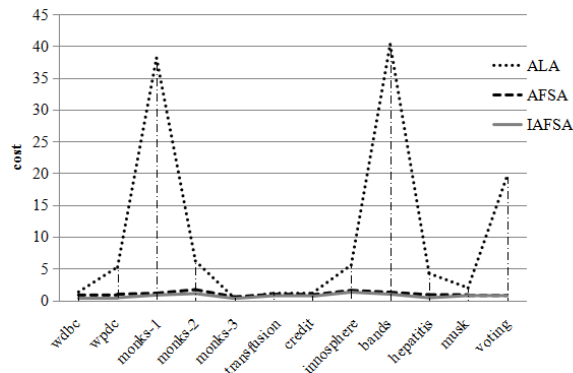


**Figure 3:** Comparative minimum cost among three algorithms in different data sets
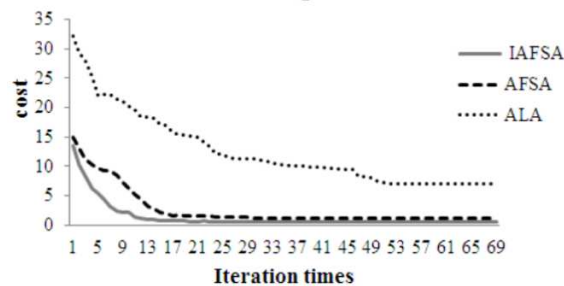


**Figure 4:** Comparative convergence curve among three algorithms

Comparative convergence curve among three algorithms is shown in Figure 4, the fitness value (risk loss) of IAFSA is obviously less than ALA and AFSA. When the iteration

13

time $N = 1,10,25$ , then $\cos t_{IAFSA} < \cos t_{AFSA} < \cos t_{ALA}$ . Besides, the algorithm has a better robustness and achieves a faster convergence. when $N = 20$ , the fitness value ( risk loss) of IAFSA remains stable and closes to 0.

## 5. Conclusions

Based on the optimization problem of the three-way decision-theoretic rough set model, the improved artificial fish swarm algorithm is adopted to learn the thresholds in three-way decision-theoretic rough set model in this paper. The experiments show that the algorithm has achieved a faster run-time and smaller cost than the other two algorithms. In conclusion, it is an effective and optional algorithm.

<div align="center">

**REFERENCES**

</div>

1. Z.Pawlak, Rough sets, *International Journal of Computer and Information Sciences*, 11(5) (1982) 341-356.
2. D.Y.Li, C.Y.Liu and Y.Du, Artificial intelligence with uncertainty, *Journal of Software*, 15(11) (2004) 1583-1594.
3. H.Sakai, M.K.Chakraborty and A.E.Hassanien, Rough sets, fuzzy sets, data mining and granular computing, *Proceedings of the 12th International Conference (RSFD Gr C 2009),* Berlin: Springer, 2009.
4. J.Qian, D.Q.Miao and Z.H.Zhang, Research on discernibility matrix knowledge reduction algorithm in cloud computing, *Journal of Computer Science*, 34(12) (2011) 2332-2343.
5. J.T.Yao, S.Ramanna, G.Y.Wang and Z.Suraj, Rough sets and knowledge technology, *Proceedings of the 6th International Conference (RSKT 2011),* Berlin: 2011.
6. J.An, X.L.Gui and W.D.Zhang, Social relation cognitive model of mobile nodes in the internet of things, *Chinese Journal of Computing*, 35(6) (2012) 1164-1174.
7. G.Y.Wang, Y.Y.Yao and H.Yu, A survey on rough set theory and applications, *Chinese Journal of Computers*, 32(7) (2009) 1229-1246.
8. K.Srinivas, G.Rao and A.Govardhan, Rough-Fuzzy classifier:A system to predict the heart disease by blending two different set theories, *Arabian Journal for Science and Engineering*, 39(4) (2014) 2457-2868.
9. Y.Y.Yao, S.K.M.Wong and P.Lingras, A decision-theoretic rough set model, *Proceedings of the 5th International Symposium on Methodologies for Intelligent System,* North-Holland, 1990,17-24.
10. Y.Y.Yao, Three-way decisions with probabilistic rough sets, *Information Science*, 180(3) (2010) 341-353.
11. D.Liu, Y.Y.Yao and T.R.Li, Three-way investment decision with decision-theoretic roughsets, *International Journal of  Computational Intelligence Systems,* 4(1) (2011) 66-74.
12. X.Y.Jia, W.W.Li and L.Shang, An adaptive learning parameters algorithm in  three-way decision-theoretic rough set model, *Acta Electronica Sinica*, 9(11) (2011) 2520-2525.
13. X.Y.Jia, Z.M.Tang and W.H.Liao, On an optimization representation of decision-theoretic rough set model, *International Journal of Approximate Reasoning*, 55(1) (2014) 156-166.

14. P.Hu, L.X.Qing and H.M.Yao, Applying artificial fish swarm algorithm to automatically determine thresholds in three-way decision-theoretic rough set model, *Computer and Modernization*, 6 (2016) 97-102.

15. Y,Y,Yao, Decision-theoretic rough set models, rough sets and knowledge technology, *Proceedings of the 2nd International Conference on Rough Sets and Knowledge Technology*, 2007, 1-12.

16. G.Y.Wang, *Rough set theory and knowledge discovery*, Xi'an Jiaotong University Press, Xi'an, China, 2001.

17. W.X.Zhang, W.Z.Wu and J.Y.Liang, Theory and method on rough set, Science Press, Beijing, China, 2001.

18. X.L.Li, Z.J.Shao and J.X.Qian, An optimizing method based on autonomous animals: fish swarm algorithm, *Systems Engineering-Theory and Practice*, 22(11) (2002) 32-38.

19. X.L.Li, An new intelligent optimization method based on artificial fish swarm algorithm, Hangzhou, Zhejiang Univisity, 2003.

20. L.G.Wang, Y.Hong and F.Q.Zhao, Improved artificial fish swarm algorithm, *Computer Engineering*, 34(19) (2008) 192-195.

21. L.G.Wang, Y.Hong and F.Q.Zhao, An artificial fish swarm algorithm based on neighborhood orthogonal crossover operator, *Transactions of the Chinese Society for Agriculture Machinery*, 39(8) (2008) 140-144.