Optimization Strategy of Anti-interference Performance Based on BPSO

Wei Baohua, Fan Shuyi, Wan Cheng, Wang Qiang
Department of Civil Engineering, Ningbo University, Ningbo 315211, China
wbhlzx@139.com

Abstract. In order to solve a problem of the anti-interference performance optimization strategy for electronic equipment, an optimization strategy based on BPSO is proposed and studied. A systematic analysis to the anti-interference performance optimization strategy for electronic equipment is made, and then the mathematical description of the optimization strategy are built. The general method and process of the solution based on BPSO to anti-interference performance optimization strategy is presented. What’s more, the method is tested and verified by a specific example. The results show that the way based on BPSO to search anti-interference performance optimization strategy is feasible. Furthermore, not only do the algorithms have good global search performance and robustness, but also the optimization efficiency and performance have advantages.

1. Introduction
Nowadays, the electronic equipment faces an increasingly complicated operational condition, especially the deteriorating electromagnetic environment, which brings a new challenge for equipment to survive in warfare[1]. With the development of science and technology in electronic warfare, electronic equipment, as an important object, is inevitably facing the threat of electronic interference. Therefore, anti-interference performance has become an extremely considerable tactics and technology index. In words, the operational effectiveness of electronic equipment depends heavily on its anti-interference performance. Therefore, accurate and objective evaluation for the performance of electronic equipment with consideration of electronic interference is the key work to explore the proper counter-measures and improve the operational effectiveness. Since the 1960s, scholars have proposed a series of anti-interference performance evaluation criteria from different perspectives, and more than 100 evaluation methods are further developed[2-7].

If the evaluation result is "good" or "pretty good", the electronic interference has no or little impact on the operational effectiveness of electronic equipment. Otherwise, it is difficult to guarantee the stability. Consequently, some measures have to be taken to optimize the anti-interference performance. This is also one of the main goals of anti-interference performance evaluation.

In this paper, the anti-interference performance optimization in technology is addressed. According to the analysis on the mechanism of electronic confrontation and electronic equipment, some technical approaches can be tried to improve its working performance under the interference. That content is also named as anti-interference performance optimization counter-measure. And the Binary Particle Swarm Optimization (BPSO) is introduced to choose the appropriate optimization counter-measures to improve anti-interference performance of electronic equipment.
2. Introduction Problem Analysis and Description

2.1. Optimization Counter-measure
Optimization counter-measure mentioned in this paper refers to the Measure of Improvement on Technology (MIT) to promote the performance in some aspects. Appropriate optimization counter-measure should focus on the weakness of anti-interference performance of electronic equipment. The weakness can be determined through analyze structure composition and work principle of electronic equipment. It may be some circuit boards, parameters, technologies or methods, etc. Then, the technical improvement counter-measure should be selected to form the selection set. Assume that \( h \) optimization counter-measures are selected. Then the section set can be written as

\[
MIT = \{ m_1, m_2, \ldots, m_h \}
\]

(1)

2.2. Benefit model
Then the benefit model is established for improvement counter-measure. With the consideration of improvement degree, benefit can be divided into absolute benefit and relative benefit. The benefit set can be depicted as

\[
BE = \{ b_1, b_2, \ldots, b_h \}
\]

(2)

where \( b_k \) is the benefit for \( m_k \), \( 1 \leq k \leq h \).

If using the Fuzzy Comprehensive Evaluation to evaluate the anti-interference performance, the evaluation grade can be divided into five degrees: "good", "pretty good", "general", "poor" and "pretty poor". Then the evaluation result can be written as

\[
\tilde{E} = [e_1, e_2, e_3, e_4, e_5]
\]

(3)

where \( e_i \) represent the affiliation weight for \( i \)-th evaluation grade.

A typical example for the absolute benefit model function can be expressed as

\[
f_1(\tilde{E}) = \sum_{i=1}^{5} \lambda_i \cdot e_i
\]

(4)

where \( \lambda_1 > \lambda_2 > \lambda_3 > \lambda_4 > \lambda_5 \).

Relative benefit model reflects the improvement degree of evaluation result. Assume that the original evaluation result is \( \tilde{E}_0 \), then the relative benefit model function can be expressed as

\[
\Delta \tilde{E} = \tilde{E} - \tilde{E}_0, \text{ named as } f_2(\Delta \tilde{E}).
\]

Similarly, \( f_2(\Delta \tilde{E}) \) can also be written as

\[
f_2(\Delta \tilde{E}) = \sum_{i=1}^{5} \lambda_i \cdot \Delta e_i
\]

(5)

2.3. Cost Model
The cost can be determined comprehensively by the factors as fee, time, collateral effects and constraint. Corresponding to the strategy section set, the cost section set can be expressed as

\[
CO = \{ c_1, c_2, \ldots, c_h \}
\]

(6)

where \( c_k \) is the cost for \( m_k \).

2.4. Optimization Strategy
The optimization strategy in this paper means to select one or more technical improvement counter-measures from the counter-measure section set to form the optimal improvement scheme. According to different requirements, the choice of technical improvement counter-measure can be divided into three types: benefit type, cost type and cost-effectiveness ratio type. In this study, cost-effectiveness ratio type was chosen.

That is, under the condition of given cost \( C \), benefit \( B \), or no constraints, to find the improvement counter-measure set \( MI \) by comparing the cost factor and benefit factor synthetically. Such
optimization problems can be described as:

**Solution field:** $MI \subseteq MIT$

**Objective function:** $\max [f(MI)]$

**constraint condition:** $f_i(\bar{E} \mid MI) \geq B$, $\sum w_k \cdot c_k \leq C$ or no constraints

$w_k$ is the weight to calculate the total cost $m_k$ of $MI$. It reflects the importance of each strategy. If there is no sufficient evidence, it can be simply thought that $w_k=1$.

The objective function includes both benefit and cost factors. Here a typical objective function expression can be given:

$$f[M] = p_2 \cdot f_i(\bar{E} \mid MI) - p_1 \cdot \sum w_k \cdot c_k$$

where $p_1>0$, $p_2>0$.

It is shown that the choice of technical improvement counter-measure is a typical set covering problem, which belongs to the category of NP-hard problem. Due to the problem of computation, etc, it is hard to get the optimal global solution, especially when the size of the alternative set is large. Generally, the suboptimal global solution can only be got.

Particle Swarm Optimization (PSO) algorithm has the advantages of simplicity, high efficiency and strong optimization ability. In particular, the BPSO algorithm is proposed to solve the combinational optimization problem in engineering practice. It is expected to be more suitable for the problem of optimization counter-measure selection for anti-interference performance in this paper. In this view, an anti-interference performance optimization strategy based on BPSO algorithm can be proposed.

3. **Solution for Optimization Strategy**

3.1. **BPSO Algorithm**

PSO is a new global optimization algorithm proposed by Eberhart and Kennedy in 1995[8, 9]. In 1997, Kennedy and Eberhart proposed a discrete binary version of PSO-BPSO algorithm to solve combinational optimization problems[10, 11]. In PSO algorithm system, each alternative set is called a "particle". It can be regarded as a particle without weight and volume in the multi-dimensional search space, and can fly at a certain speed in the search space. The flight speed is dynamically adjusted by the flight experience of particles and groups[12, 13]. BPSO follows the basic principle of PSO, the result of particle optimization is actually the solution of combinational optimization problem.

3.2. **The Solving Method Based on BPSO**

(1) **Particle Construction**

The position vector of each particle is composed by binary code with length of $h$, which is consistent with the dimension of the technical improvement counter-measure set, to yield

$$X_i(k) = [x_{i1}(k), x_{i2}(k), \ldots, x_{ih}(k)]$$

where $1 \leq j \leq h$. The entry $x_{ij}(k)$=1 represents that the corresponding counter-measure is selected, whereas $x_{ij}(k)$=0 means that it is not selected.

Meanwhile, the velocity vector of a particle is constructed as

$$V_i(k) = [v_{i1}(k), v_{i2}(k), \ldots, v_{ih}(k)]$$

Theoretically, the entry $v_{ij}(k)$ satisfies $v_{ij}(k) \in (-\infty, +\infty)$. For a particle, the larger the particle is, the greater the probability is for a one-dimensional value to takes value 1 in the corresponding position vector, and vice versa.

(2) **Particle Fitness Function**

According to the basic idea of particle swarm optimization, anti-interference performance optimization strategy is a cost-effective problem. The criterion to measure the particle fitness is total cost $\sum w_k \cdot c_k$, with the consideration of total benefit $f_i(\bar{E} \mid MI)$. Therefore, based on the equation (7), a new fitness function is presented as
\[ f[X_i(k)] = p_2 \cdot f_1(\tilde{E} \mid MI) - p_1 \cdot \sum_{j=1}^{h} w_j \cdot c_j \cdot x_j(k) \]  

(10)

where \( MI \) is determined by \( X_j(k) \), which means that \( MI \) consists of the counter-measures indicated by corresponding entries in \( X_j(k) \) with value 1.

In order to decrease the cost as much as possible, the fitness function is modified to satisfy the given constraint conditions as

\[
\begin{align*}
    f[X_i(k)] &= \begin{cases} 
        p_2 \cdot f_1(\tilde{E} \mid MI) - p_1 \cdot \sum_{j=1}^{h} w_j \cdot c_j \cdot x_j(k) & \text{if } f_1(\tilde{E} \mid MI) < B \\
        p_2 \cdot f_1(\tilde{E} \mid MI) - p_1 \cdot \sum_{j=1}^{h} w_j \cdot c_j \cdot x_j(k) + \delta & \text{else}
    \end{cases}
\end{align*}
\]

(11)

In the equation (11), a reward parameter \( \delta (\delta > 0) \) is added to increase the search performance under the constraint of total benefit. In addition, according to the actual operation of the algorithm, the value of weight coefficient \( p_1 \) and \( p_1 \) can be adjusted appropriately to improve the convergence speed.

(3) Initialization for Particle Swarm

The initialization aims to generate initial position and velocity vectors for \( m \) particles as

\[
\begin{align*}
    X_i(0) &= [x_{i1}(0), x_{i2}(0), \ldots, x_{ih}(0)] \\
    V_i(0) &= [v_{i1}(0), v_{i2}(0), \ldots, v_{ih}(0)]
\end{align*}
\]

(12)\hspace{2cm} (13)

The initialization of position vector for particle swarm can be generated randomly. The typical approach is Monte Carlo method, just as

\[
\begin{align*}
x_{ij}(0) = \begin{cases} 
    1 & \text{if } p > \text{rand} \\
    0 & \text{else}
    \end{cases}
\end{align*}
\]

(14)

where \( p \) belongs to \((0, 1)\). In essence, \( x_{ij}(0) \) takes the value 1 at the probability of \( p \), which means that the corresponding counter-measure is selected at the probability of \( p \). In particular, if there is no prior information of improvement counter-measure, set \( p=0.5 \). According to the prior information of each strategy, a tendentious threshold can be presented. For the counter-measure that may generate big benefit, a large threshold and vice versa are set. Then the initial position tends to choose the counter-measure set that can generate big benefits, which will contribute to the convergence speed.

For the initial velocity vectors of particles, they can also be generated randomly. Or all the velocities can be simply initialize to 0. Adopting the random generation, it is assumed that the range of the \( j \)-th (\( 1 \leq j \leq h \)) entry in velocity vector is limited to \([V_{\text{min}j}, V_{\text{max}j}]\). Then the corresponding initial speed value is given by

\[
v_{ij}(0) = V_{\text{min}j} + \text{rand} \cdot (V_{\text{max}j} - V_{\text{min}j})
\]

(15)

(4) Searching for the Optimal Solution

Generally, there are two steps. Firstly, the individual optimum pbest of each particle position is searched, and then the whole optimum gbest is searched, all of which take the fitness value by equation (11).

**Searching for pbest:** For each particle, the current fitness value is compared with the fitness of individual optimum position pbest. If it is better, it will be pbest, otherwise the original pbest will be maintained. Obviously, the initial pbest of any particle is its initial position, which can be expressed as

\[
P_i(0) = [p_{i1}(0), p_{i2}(0), \ldots, p_{ih}(0)] = X_i(0)
\]

(16)

Then, the pbest of any particle in step \( k \) can be obtained by

\[
P_i(k) = \begin{cases} 
    X_i(k) & \text{if } f[X_i(k)] \geq f[X_i(k-1)] \\
    P_i(k-1) & \text{else}
    \end{cases}
\]

(17)

**Searching for gbest:** For each particle, the current fitness value is also compared with the fitness value of the global optimum position gbest. If it is better, it will be the gbest, otherwise the original
$gbest$ will be maintained. Similarly, $gbest$ can be initialized as

$$P_g(0) = [p_{g1}(0), p_{g2}(0), ..., p_{gn}(0)]$$  \hspace{1cm} (18)

Obviously, the initial $gbest$ of particle swarm is the one with the largest fitness value among all initialized particles. Assume that the initial $gbest$ is $X_{i}(0)$, where $(1 \leq i \leq n)$. Then $P_g(0) = X_{i}(0)$, the condition is fulfilled

$$f[X_{i}(0)] = \max_{i=1}^{n} f[X_{i}(0)]$$  \hspace{1cm} (19)

Therefore, the $gbest$ for the $k$-th step is expressed as

$$P_g(k) = \begin{cases} P_g(k-1) & \text{if } \max_{i=1}^{n} f[X_{i}(k)] \leq f[P_g(k-1)] \\ X_{i}(k) & \text{else} \text{ and } f[X_{i}(k)] = \max_{i=1}^{n} f[X_{i}(k)] \end{cases}$$  \hspace{1cm} (20)

(5) Searching Iteratively

In BPSO algorithm, each dimension of position vector is strictly limited to 0 or 1. And corresponding velocity $v_{ig}(k)$ reflects the probability of $x_{ig}(k)$ to take value 1. In the iterative search process, the update of binary particle position and velocity can be depicted as

$$v_{ig}(k+1) = \omega \cdot v_{ig}(k) + d_1 \cdot r_1 \cdot [p_{ig}(k) - x_{ig}(k)] + d_2 \cdot r_2 \cdot [P_g(k) - x_{ig}(k)]$$

$$x_{ig}(k+1) = \begin{cases} 1 & \text{if } S[v_{ig}(k+1)] > \text{rand} \\ 0 & \text{else} \end{cases}$$  \hspace{1cm} (21)

where $\omega$ is the inertia weight\cite{14,15}, which can be linearly reduced from the initial large value $\omega_{\text{max}}$ to the small value $\omega_{\text{min}}$ during iteration, so as to realize the fast global and local search. $d_1$ and $d_2$ are normal positive numbers, called as acceleration factors. $d_1$ is used to adjust the step size for flying to their individual optimum position. $d_2$ is used to adjust the step size for flying to the global optimum position\cite{16,17}. $r_1$ and $r_2$ are random numbers belonging to the interval [0,1], which are independent with each other. The function $S(\cdot)$ is usually the Sigmoid function as

$$S[v_{ig}(k)] = \frac{1}{1 + \exp[-v_{ig}(k)]]}$$  \hspace{1cm} (22)

When $|v_{ig}(k)|$ is large, $S[v_{ig}(k)]$ will quickly approach 0 or 1, and saturation will occur. Then the iteration will fall into local optimum. To avoid it, an upper bound $V_{\text{max}}$ is usually set. When the value $V_{\text{max}}$ is large, the convergence speed is high. But it tends to local search of the current optimal solution. When $V_{\text{max}}$ is small, the algorithm tends to global search, but its convergence speed is slow. Therefore, a balance consideration for $V_{\text{max}}$ should be given. For example, in the work \cite{18}, $V_{\text{max}}$ is set to 4, $S(\cdot)$ falls into the interval [0.018, 0.982]. In the iteration, $x_{ig}(k)$ will take values 0 or 1 with an appropriate probability, which will avoid the premature of the algorithm. That is conducive to the search of the global optimum solution.

(6) Algorithm Implementation

In conclusion, the basic implementation process of the counter-measure selection to improve the anti-interference performance is as follows:

**STEP.1** Set parameters for BPSO algorithm;

**STEP.2** Initialize particle swarm including random setting of each particle's initial position and initial speed;

**STEP.3** Calculate the fitness value of each particle utilizing the equation (11);

**STEP.4** Search for $p_{best}$;

**STEP.5** Search for $gbest$;

**STEP.6** Update particles with equation (21);

**STEP.7** End and turn to **STEP.8**, if the end condition is met. Otherwise, turn to **STEP.3**;

**STEP.8** Output $gbest$ to determine the counter-measure set $MI$.  

5
4. Verification and analysis

In order to verify its effectiveness of anti-interference performance optimization strategy based on BPSO algorithm, an experiment has been fulfilled and further discussion has been made.

Firstly, a technology improvement counter-measure set MIT for anti-interference performance of electronic equipment is constructed, which is composed of 30 strategies. Then the fuzzy comprehensive evaluation result \( \tilde{E} \) of anti-interference performance is given after implementing each improvement counter-measure. The parameter setting is \( \lambda_1=2, \lambda_2=1, \lambda_3=0, \lambda_4=-1 \) and \( \lambda_5=-2 \). After utilizing equation (4), the benefit can be calculated to construct the benefit set \( BE \). Meanwhile, corresponding cost set is established as \( CO \).

To present the experiment condition, two assumptions about the benefit and cost is given as: (1) The cost of each improvement counter-measure is independent and additive, and the weight of each counter-measure cost is 1, which means \( w_j=1.4 \); (2) As for the acquisition of \( f_1(\tilde{E} | MI) \), \( f_{max} \) is the maximum benefit for a single improvement counter-measure.

Then the following model is constructed to depict the total benefit for multi counter-measure as

\[
f(\tilde{E} | MI) = f_{max} \cdot [1 - \prod_{j=1}^{30} \left( 1 - \frac{f_1(\tilde{E} | m_j)}{f_{max}} \right)]
\]

To construct the BPSO algorithm model, set particle dimension \( h=30 \), particle number \( m=10 \). The other parameters are set as \( K=200, V_{min}=-4, V_{max}=4, d_1=d_2=2, \omega_{max}=1.4, \omega_{min}=0.4, p_1=p_2=0.5, B=1.5 \) and \( \delta=0.1 \).

The particle position vector is composed of 30 entries. Every entry takes the values 0 or 1, which means the counter-measure is selected or unselected. The dimension of particle velocity vector is the same, and the value is in the range of \([-4,4]\). The fitness function is:

\[
f[X_j(k)] = \begin{cases} 
0.5 \cdot f_1(\tilde{E} | MI) - 0.5 \cdot \sum_{j=1}^{30} c_j \cdot x_j(k) & \text{if } f_1(\tilde{E} | MI) < 1.5 \\
0.5 \cdot f_1(\tilde{E} | MI) - 0.5 \cdot \sum_{j=1}^{30} c_j \cdot x_j(k) + 0.1 & \text{else} 
\end{cases}
\]

The initial position vectors and velocities vector of 10 particles are randomly generated, and the iterative search is carried out according to the steps of STEP.3-STEP.8. It is shown that the satisfactory solution can be obtained by 80-90 steps on average and the same result can be obtained by several experiments. As shown in Fig. 1, when the experiment iteration is 89, satisfactory solution is found. Fig.1(a) shows the change of fitness value of global optimum during iteration. Fig.1(b) shows the change of fitness value of individual optimum during iteration.

![Fig.1 Change of Fitness Value in Iteration Course of BPSO Algorithm](image)

It is shown in the figure that the convergence and global search performance of BPSO algorithm in this example are good. The global optimum is

\[
gbest = \{00000000000000000000000001001010\}
\]

The counter-measure set obtained by searching is \( MI = \{m_{24}, m_{27}, m_{29}\} \). Then the fitness value is 1.0215, benefit is 1.9347, cost is 0.092. That means when the counter-measures 24, 27 and 29 in...
Table 1 are chosen as the improvement counter-measures, the maximum cost-effectiveness ratio can be obtained at the promise of benefit larger than 1.5. This is consistent with the data in Table 1. The examples above show that the BPSO algorithm can effectively search the satisfactory solution when it is used in the selection of counter-measure for improving the anti-interference performance for electronic equipment. Compared with the traditional method, the searching process is completed in a few iterations. Meanwhile, the global search ability and robustness are good. It also shows unique advantages in efficiency and effectiveness.

5. Conclusion
The choice of the anti-interference performance optimization counter-measure for electronic equipment is a combinatorial optimization problem. It is also a typical set covering problem, which belongs to the category of NP-hard problem. Due to the limitation of computation, robustness and global optimization ability, it is difficult to solve this problem by using traditional algorithm. The BPSO algorithm has the advantages of simplicity, high efficiency and strong optimization ability. It is suitable to solve the optimization problem to improve the anti-interference performance.

In this paper, the essence of anti-interference performance optimization strategy of electronic equipment is systematically analyzed, and the mathematical description is established. The general method and process of solving anti-interference performance optimization strategy based on BPSO are given, and the effectiveness is verified and analyzed based on the experimental example. The results show that this scheme based on BPSO is feasible. Moreover, the global search ability and robustness of this algorithm are good. It shows unique advantages in efficiency and effectiveness. It can be instructive to solve the combinatorial optimization problems. When the size of the alternative counter-measure set is large, the scheme proposed in this paper will have great advantages in the global optimization ability and operation speed.

Acknowledgments
This work was financially supported by the open foundation State key laboratory of CEMEE (2017Z0301B1)

References
[1] LIU Shang-he. Effect of Electromagnetic Environment to Weaponry and its Trend of Development[J]. Journal of the Academy of Equipment Command & Technology, 2005, 16 (1):1-6(in Chinese).
[2] LI Chao, ZHANG Ju-quan. The Present Situation and Prospect of Radar Anti-jamming Effectiveness Evaluation [J]. Electronic Countermeasure, 2003, 26 (5): 41-45(in Chinese).
[3] CHENG Yong, SHI Yin-shui, ZHU Yan. A Study and Assumption of Radar ECCM Efficiency Evaluationin Complicated Electromagnetic Circumstance[J]. National Defense Science & Technology, 2008, 29 (1):52-56(in Chinese).
[4] GUO Jin-yuan, HUANG Jun. Anti-jamming Effectiveness Evaluation of Airborne Radar Based on Fuzzy AHP[J]. Electronics Optics & Control, 2007, 14 (8):151-153, 167(in Chinese).
[5] LU Ying-qi, WANG Ying-long, ZHU Chang-ying. Evaluation on Radar Anti-jamming Effectivenessbased on Neural Network[J]. Fire Control and Command Control, 2006, 31(1):77-80(in Chinese).
[6] ZHOU Ying, WANG Xue-song, XU Zhen-hai, ZHAO Feng, WANG Guo-yu. General Consideration on Performance and Effectiveness Evaluation in Radar EW[J]. Systems Engineering and Electronics, 2003, 29 (5):617-620(in Chinese).
[7] PAN Chao, TANG Bin. A Neural Network-based Method of Making Performance Evaluation of Radar ECCM[J]. Radar & ECM, 2006, 26 (3):4-7, 14(in Chinese).
[8] J. Kennedy. Stereotyping. Improving Particle Swarm Performance With Cluster Analysis[C]//Congress on Evolutionary Computation. San Diego, USA: IEEE, 2000:1507-1512.
[9] Lei Kai-you. Research on Particle Swarm Optimization and Its Application[D]. Chongqing: Southwest University, 2006(in Chinese).

[10] Kennedy J., Eberhart R C. A Discrete Binary Version of the Particle Swarm Algorithm[C]//World Multi Conference on Systemics, Cybernetics and Informatics, Piscataway, NJ: IEEE, 1997:4104-4109.

[11] Jun Sun, Wenbo Xu. A Global Search Strategy of Quantum-behaved Particle Swarm Optimization[C]//IEEE 2004 Conference on Cybernetics and Intelligent Systems, Singapore: IEEE, 2004:111-116.

[12] FAN Na, YUN Qing-xia. Particle swarm optimization algorithms and its applications[J]. Information Technology, 2006(1):53-56(in Chinese).

[13] CHEN Wei, FENG Bin, SUN Jun. Simulation Study on The Parameters Optimization of Radial Basis Function Neural Network Based on QPSO Algorithm[J]. Journal of Computer Applications, 2006, 26(8):1928-1931(in Chinese).

[14] Shi Y, Eberhart R C. A Modified Paticle Swarm Optimizer[C]//Proceedings of the 1998 Conference of Evolutionary Computation. Piscataway, NJ: IEEE, 1998:69-73.

[15] Shi Y, Eberhart R C. Fuzzy Adaptive Paticle Swarm Optimization[C]//Proceedings of the 2001 Conference of Evolutionary Computation. Piscataway, NJ: IEEE, 2001:101-106.

[16] Kennedy J., Eberhart R C. Particle Swarm Optimization[C]//In:Proc. of the 1995 IEEE International Conference on Neural Networks, Australia: IEEE, 1995:1942-1948.

[17] A. Carlisle G. Dozier. An off-the shelf PSO[C]//Proceedings of the Partial Swarm Optimization Workshop, Indianapolis, USA: 1N, 2001:1-6.

[18] LIAN Guang-yao. Research on Deign and Analysis Method of Equipment Testability Based on Information Model[D]. Shijiazhuang: Shijiazhuang Machine Engineering College, 2007(in Chinese).