Application of Particle Swarm Optimization on Reliability Sampling Inspection Program for Success or Failure Product

Yuanbo Xiong¹, Nuo Cheng², Yunzi Liu¹

¹. PLA Trop 63856, Baicheng 137001, China
². PLA Trop 63853, Baicheng 137001, China
xybnjust@163.com

Abstract. Reliability sampling inspection program for success or failure product is an intractable problem in the reliability test field. Meta-heuristic algorithm provides a new idea to solve the problem. Particle swarm algorithm is a new meta-heuristic algorithm in recent years, which has been widely used in many fields because of its simplicity but efficiency. In this paper, a reliability sampling inspection program model is developed to minimize the distance of risk. In addition, the mechanism of the particle swarm algorithm is expounded, and a case study reveals that the method in this paper is feasible and effective.

1. Introduction

Sampling Inspection test is a special kind of hypothesis test, which uses statistical methods to develop test program for a batch of products’ inspection according to the concerted request of the producer and the user and make acceptation or rejection judgment according to the testing results of this batch of products.[1] In the sampling inspection test, every application of the tests is independent from each other. And for those products that have only two kinds of result as success and failure are called success or failure products, such as explosives, missiles and rockets. At present, the most close to the requirement of the producer and the user is picked as the design of success or failure product sampling inspection program from GB5080.5-85, the success rate of equipment reliability test validation test plan. And GB5080.5-85 gives four kinds of condition of optimal solution when both sides of risk are equal to 5%, 10%, 15% and 20%. However, the risk in fact might be other values, for example 6%, 15% and 18% etc. or the two sides of risk might not be the same, which makes the application of the standard has certain limitations.

The sampling inspection program is determined by n and Ac (when the product batch is large), where n is the sample size extracted and Ac is the number of conformity determined. In essence, the design of reliability sampling inspection program is an integer optimization problem with constraints, and there is still no general analytical method for accurate solution. If the traversal of n and Ac is adopted, it needs to traverse \( n \times Ac \) times, with very low efficiency. For example, if n is 500 and Ac is 40, it needs to traverse \( 2 \times 104 \) times. Reference [3] studied the testability verification test program based on the development information by using the evidence theory method on the basis of the success or failure fixed-number verification model considering the risks of both parties, and achieved good results in testability verification. Reference [4] proposed equivalent deformation of the identification program equations and designed relevant algorithms to solve them, but the algorithm process was relatively complicated. Reference [5] proposed a method of dynamic step size selection to solve the equations, but the solution requires the scheme in GB5080.5-85 and has high requirements for users.
These research methods are complicated and cannot simplify the solution process of scheme design effectively. Therefore, it is extremely urgent to design a practical and efficient algorithm to solve this problem.

Particle swarm optimization is a new swarm intelligence algorithm, which is favored and widely used by researchers due to its high performance, few parameters and strong robustness [6]. As an efficient and novel optimization algorithm, particle swarm optimization has not been introduced into the design of reliability sampling inspection program. In this paper, the design model of sampling inspection program is established first, then the mechanism of the standard particle swarm optimization algorithm is described, and the solution process is designed. Finally, the improved algorithm is applied to the design of inspection program.

2. Design model of success or failure product sampling inspection program

2.1. Problem

Success or failure product sampling inspection program design process: randomly select \( n \) samples from a batch of products that total amount is \( N \) for inspection. Hypothetically, it is tested out \( d \) unqualified products, then when \( d \) is less than or equal to \( Ac \), receive this batch of products, otherwise refuse. Usually this test plan is written as \((N, n, Ac)\), where \( N \) as the known quantity, and as long as \( n \) and \( Ac \) are determined, the sampling inspection program is determined. When \( N \) is relatively large compared with \( n \), the influence of batch \( N \) is not significant. Therefore, under this condition, the sampling inspection program can be denoted as \((n, Ac)\), and \( Ac < n \) is obviously available.

There are two kinds of risks in the sampling inspection program: producer’s risk and user’s risk. The producer’s risk is the probability that the conforming critique is a nonconforming batch, and the user’s risk is the probability that the nonconforming critique is a conforming batch. The calculation methods of and are shown in equation (1)[7], where \( R_0 \) is the superior limit of the test and \( R_1 \) is the inferior limit of the test.

\[
\begin{align*}
\alpha &= 1 - \sum_{i=0}^{Ac} \binom{n}{i} (1 - R_0)^i R_0^{n-i} \\
\beta &= \sum_{i=0}^{Ac} \binom{n}{i} (1 - R_1)^i R_1^{n-i}
\end{align*}
\]  

In the design process, both sides want to keep their risk as low as possible, but theoretical research shows that to making \( \alpha \) small larges \( \beta \), and to make \( \beta \) small leads to large \( \alpha \). It is important for both sides to design a program that can balance the risks of both sides. In practice, the producer and user first agree on a mutually acceptable risk rating of \( \alpha_0 \)(producer rated risk) and \( \beta_0 \)(user rated risk) based on the inspection superior and inferior limits, and then weigh the program by comparison.

2.2. The objective function of the problem

In the design of the sampling program, the producer wants the lower \( \alpha \) and the closer to \( \alpha_0 \)the better; the user hopes that the lower \( \beta \) and the closer to \( \beta_0 \)the better, both sides want their risk as close as possible to the rated risk, therefore, sampling scheme design process is the game of risk balance both sides, the objective function can not just consider one side of risk and risk distance minimum, but should consider both \( \alpha \) close to \( \alpha_0 \), and to ensure \( \beta \) close to \( \beta_0 \). Based on the risk and risk rating on both sides of the Manhattan distance minimum as objective function, defined as shown in formula (2), so we can make the design conforms to the interests of both parties, that both sides can accept.

\[
\min f(n, Ac) = \sqrt{(\alpha - \alpha_0)^2 + (\beta - \beta_0)^2}
\]  (2)
2.3. The constrain condition of the problem  
Taking \( N = 500 \) and \( Ac = 40 \) as examples, the response surface of sample size, qualified decision number to risk distance and sample size - qualified decision number profile view are made, as shown in Fig.1.

![Fig.1 Response surface of Sample size, acceptance number and risk distance](image_url)

(a) Vertical view   
(b) Three dimension view

From Fig.1 (a), we can find that the good feasible solutions are concentrated in a very narrow region. Theoretically, the following analysis is made: if the estimated boundary point value \((1 - AC/n)\) is less than the inferior limit the user's risk will increase. Then if \((n, Ac)\) falls in the red area at the upper left of Fig. 1 (b), the risk difference is relatively large, and the user is obviously not satisfied. If the estimated boundary point value is greater than the superior limit value, then the producer risk is too large. At this time, if \((n, Ac)\) falls in the red area in the lower right and the producer is obviously not satisfied. When the estimated boundary point is between the upper limit and the lower limit of the test, as it falls in the blue triangle area in Fig. 1 (b), the risk difference between the two sides becomes smaller, and the risk distance can be balanced. Based on the above analysis, the value range of \( Ac \) is determined as follows:

\[
(n(1 - R_0) \leq Ac \leq n(1 - R_1))
\]

(3)

2.4. Optimization model

Based on the above analysis, the design model of reliability sampling inspection program for success or failure products targeting risk distance is as follows:

\[
\begin{align*}
\min f(n, Ac) &= \sqrt{(\alpha - \alpha_0)^2 + (\beta - \beta_0)^2} \\
&= \begin{cases} 
    n(1 - R_0) \leq Ac \leq n(1 - R_1) \\
    0 < n \leq n_{\text{max}} \\
    \alpha = 1 - \sum_{i=0}^{\left[\frac{n}{Ac}\right]} \binom{n}{i} (1 - R_0)^i R_0^{n-i} \\
    \beta = \sum_{i=0}^{\left[\frac{n}{Ac}\right]} \binom{n}{i} (1 - R_1)^i R_1^{n-i} \\
    Ac \in \mathbb{N}^+, n \in \mathbb{N}^+
\end{cases}
\end{align*}
\]

(4)

Where, \( n_{\text{max}} \) is the maximum sample size in the formula.

3. Particle swarm algorithm

3.1. The basic principle

Suppose in a d-dimensional target search space, there are m particles forming a community, in which the position of the \( ith \) particle in the d-dimensional search space is represented as a d-dimensional vector, and the position of each particle represents a potential solution. Let \( X_i = (x_{i1}, x_{i2}, \ldots, x_{id}) \) is the
current position of particle \( i \); \( v_i=(v_{i1}, v_{i2}, \ldots, v_{id}) \) is particle \( i \)'s current flight speed; \( p_i=(p_{i1}, p_{i2}, \ldots, p_{id}) \) is the best position experienced by particle \( i \), that is, the position with the best adaptive value experienced by particle \( i \), which is called the individual optimal position; \( g_d=(g_{d1}, g_{d2}, \ldots, g_{dD}) \) is the optimal location searched so far by the whole particle swarm, which is called the global optimal location. If \( X_i \) is plugged into the target function, its adaptive value can be calculated, and \( X_i \) can be evaluated according to its adaptive value. The position and velocity of each particle are iterated by following equations (5) and (6) [8].

\[
V_i(t+1) = wV_i(t) + c_1r_1(t)(p_i(t) - X_i(t)) + c_2r_2(t)(g_d(t) - X_i(t)) \tag{5}
\]

\[
X_i(t+1) = X_i(t) + V_i(t+1) \tag{6}
\]

The subscript \( j \) is the \( j \)th dimension of the particle \( (j=1,2,\ldots,d) \), and \( i \) represents the \( i \)th particle \( (i=1,2,\ldots,m) \), \( t \) represents the \( t \) generation, \( w \) is the inertia weight coefficient, \( c_1 \) and \( c_2 \) are the acceleration coefficient, \( c_1 \) regulates the step size of the particle flying to its own optimal position, \( c_2 \) regulates the step size of the particle flying to the global optimal position. \( r_1,U(0,1) \) and \( r_2,U(0,1) \) are two independent random numbers. In order to reduce the possibility of particles leaving the search space during iteration, \( v_i \) is usually limited to a certain range, namely \( v_i \in [-v_{max},v_{max}] \). If the search space of the problem is limited within \([v_{max},v_{max}]\), then \( v_{max}=k\times x_{max} \), \( 0.1 \leq k \leq 1 \). In iteration, if the position and velocity of the particle are beyond the limit range, the boundary value is adopted. The initial position and velocity of all particles are randomly generated, and then the equations (5) and (6) are used to iterate, constantly changing their speed and position until a satisfactory solution is found or the maximum number of iterations is reached. For particle swarm optimization algorithm, there are usually several important control parameters as follows [9].

- Particle swarm size is how many particles contained in number;
- Acceleration constants \( c_1 \) and \( c_2 \), generally value between 0 ~ 2;
- Inertia weight coefficient \( w \), general take \([0.9, 1.2]\);
- Control particle speed \( v_{max} \), usually set the superior limit for the variable.

3.2. Algorithmic solution flow

The main calculation steps of particle swarm algorithm are as follows:

1. Parameters were set, acceleration constants \( c_1 \) and \( c_2 \) were set, the maximum number of iterations was \( iter_{max} \), and the initial population position \( X(0) \) and velocity \( V(0) \) were randomly generated;
2. Evaluate the population \( X(t) \) and calculate the adaptive value of each particle in each dimensional space;
3. Calculate the fitness value of each particle according to the fitness function, store the position and fitness of each particle in \( lbest \), and store the position and fitness of the individual with the best fitness in \( lbest \) or \( gbest \);
4. Compare the particle fitness value with the optimal value of the population. If the current value is better than \( gbest \), assign the current optimal value to \( gbest \);
5. Update the velocity and position formulas of particles according to formulas (5) and (6) to generate a new population \( X(t+1) \);
6. Check whether the end condition is satisfied. If it is, the optimization calculation is finished; otherwise, \( t = t + 1 \) and then go to (2). The ending condition is generally optimized to achieve the maximum evolutionary algebra \( iter_{max} \), or the change of the objective function value is less than the given accuracy \( \epsilon \).

4. The example analysis

A new intelligent ammunition requires batch inspection, including batch \( N = 10000 \), the largest sample size \( n_{max} \) does not exceed 500 (large batch), reliability index in the contract for \( R_0 = 0.9 \), \( R_1 = 0.8 \), the
risk of producer and user is $\alpha_0=10\%$ and $\beta_0=15\%$. The reliability appraisal test solution model as shown in formula (7).

$$\min f (n, Ac) = \sqrt{\left(\alpha - \alpha_0\right)^2 + \left(\beta - \beta_0\right)^2}$$

$$0.1n \leq Ac \leq 0.2n$$

$$0 < n \leq 500$$

$$\alpha = 1 - \sum_{i=0}^{n}\left(\frac{n!}{i!}\right)0.1^i0.9^{n-i}$$

$$\beta = \sum_{i=0}^{n}\left(\frac{n!}{i!}\right)0.2^i0.8^{n-i}$$

$$Ac \in N^+, n \in N$$

In order to verify the rationality of the proposed model, the three groups were given the same risk: $\alpha_0=\beta_0=5\%$, $\alpha_0=\beta_0=10\%$ and $\alpha_0=\beta_0=20\%$. Then the optimal sampling inspection program designed by the method in this paper was compared with the scheme in GB5080.5-85. The results are listed in Table 1.

| $R_0$ | plans in GB5080.5-85 | plans in this paper |
|-------|----------------------|---------------------|
| $n=1$ | $\alpha=4.8\%$      | $\alpha=4.8\%$      |
| $n=2$ | $\beta=9.9\%$       | $\beta=9.9\%$       |
| $n=3$ | $\alpha=19.0\%$     | $\alpha=19.0\%$     |
| $n=4$ | $\beta=20.4\%$      | $\beta=20.4\%$      |

TABLE 1 Comparison of plans in GB5080.5-85 and this paper

As it showed in the Table 1, when the rated risk is 5%, the program in this paper is the same as the GB program. When the rated risk is 10%, $\alpha$ is 10%, which equal to the rated risk; $\beta$ is 10.1%, which is 1% different from the rated risk. While in GB program the $\alpha$ is 8.6% which is 14% different from the rated risk; $\beta$ is 9.9%, which has 1% different from the rated risk. When the rated risk is 20%, the $\alpha$ and $\beta$ of the proposed program differ from the rated risk by 6% and 2% respectively, but $\alpha$ and $\beta$ of the GB program differ by 5% and 10% respectively. The program in this paper is closer to the rated risk than the national standard program, which shows the advance of the model in this paper.

5. Conclusions

The key of success or failure product reliability sampling inspection program design is how to balance the risk of producer and user. In this paper, the reliability sampling inspection program optimization model targeting at risk distance can accurately quantitatively describe the problem and quantify the design process, which avoid the problem that can be solved only by trying method. The result of the example shows that particle swarm algorithm can solve the integer optimization model in this paper, and it is easier to find the optimal solution compared with other algorithms, which provides a new way to solve the problem of reliability sampling inspection program for success or failure products.

References

[1] MAO Shi-song, TANG Yin-cai, WANG Ling-ling. Reliability Statistics [M]. Beijing: Higher education Press, 2008.
[2] GB5080.5-85, Equipment reliability testing compliance test plans for success ratio [S].
[3] C Chang, J Yang. Study on the scheme of testability demonstration test based on development information [J]. Acta Aeronautica Et Astronautica Sinica, 2012, 33(11):2057-2064.
[4] LI Xiao-yang, JIANG Tong-min, XIAO Liang-hua. Development algorithm of single sampling inspection plan of pass-fail experiment [J]. Journal of Beijing University of Aeronautics and Astronautics, 2012, 33(11):2057-2064.
[5] DU Xu, WANG Zhao-jian, MA Tao, et al. Optimization Algorithm of Type-II Censoring of Considering both Sides Risk Inspection Plan of pass-fail Experiment [J]. Fire Control&
Command Control, 2015, 40(8): 52-55.

[6] TIAN Xinghua, ZHANG Jihui, Li Yang, An Adaptive Annealing Particle Swarm Optimization Based on Chaotic Mapping [J]. Complex Systems and Complex Science, 2020, 17(1): 45-54.

[7] HE Guo-wei. Reliability Test Technology [M]. Beijing: National Defense Industry Press, 1996.

[8] LV Xuezhi, YU Yongli, ZHANG Liu, et al. Component Repair Model of Resource Constraint and Particle Swarm Optimization Algorithm [J]. System Engineering Theory and Practice, 2013, 33(4): 1013-1018.

[9] QI Mingjun, Improvement and Application of Particle Swarm Optimization for Solving Multi-objective Optimization Problems [D]. Daqing Petroleum College, 2007.