

Research Article

An Improved Lagrange Particle Swarm Optimization Algorithm and Its Application in Multiple Fault Diagnosis

Xiaofeng Lv ^{1,2}, Deyun Zhou,¹ Ling Ma,² Yuyuan Zhang,² and Yongchuan Tang ^{1,3}

¹School of Electronics and Information, Northwestern Polytechnical University, Xi'an, Shaanxi 710072, China

²Naval Aviation University, Yantai, Shandong 264001, China

³School of Big Data and Software Engineering, Chongqing University, Chongqing 401331, China

Correspondence should be addressed to Xiaofeng Lv; lx00112@mail.nwpu.edu.cn and Yongchuan Tang; tangyongchuan@mail.nwpu.edu.cn

Received 31 December 2019; Revised 22 May 2020; Accepted 6 June 2020; Published 27 June 2020

Academic Editor: Jiawei Xiang

Copyright © 2020 Xiaofeng Lv et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The fault rate in equipment increases significantly along with the service life of the equipment, especially for multiple fault. Typically, the Bayesian theory is used to construct the model of faults, and intelligent algorithm is used to solve the model. Lagrangian relaxation algorithm can be adopted to solve multiple fault diagnosis models. But the mathematical derivation process may be complex, while the updating method for Lagrangian multiplier is limited and it may fall into a local optimal solution. The particle swarm optimization (PSO) algorithm is a global search algorithm. In this paper, an improved Lagrange-particle swarm optimization algorithm is proposed. The updating of the Lagrangian multipliers is with the PSO algorithm for global searching. The difference between the upper and lower bounds is proposed to construct the fitness function of PSO. The multiple fault diagnosis model can be solved by the improved Lagrange-particle swarm optimization algorithm. Experiment on a case study of sensor data-based multiple fault diagnosis verifies the effectiveness and robustness of the proposed method.

1. Introduction

The problem of fault detection and diagnosis in complex equipment is a hot topic for theory and practical research [1–5]. On one hand, if the complex electromechanical system of existing weapons or equipment fails, it is difficult to address the problems solely with traditional diagnostic methods or the experience of maintenance personnel, because the system is composed of very complicated mechanical and electronic systems [6–8]. On the other hand, studies show that the available maintenance cost of complex equipment is a key issue and may be extremely high [9, 10]. Meanwhile, because of lack of effective detection and maintenance procedure, the maintenance cost of equipment is often 3–10 times bigger than the research cost. Therefore, it is of great importance to make effective fault diagnosis for weapon equipment, especially for complex electromechanical equipment. Existing studies [11–14] play a great effort in lowering the cost of maintenance and test,

simplifying the test equipment, and improving the testability of equipment.

Model-based fault diagnosis and data-driven fault diagnosis are commonly used fault diagnosis methods [15, 16], among which the dependency model of Data Sciences International (DSI) company is a typical approach [6, 17, 18]. The optimization algorithm can be used to solve the objective function under the correlation model of complex system. The correlation model is transformed into a multiple fault diagnosis model in [19]. Subsequently, the multiple fault diagnosis model is solved based on the Lagrange relaxation algorithm [20, 21]. Apart from being applied in multiple fault diagnosis, the Lagrange relaxation algorithm is also applied for other applications as partitioning problem [22], scheduling problem [23], the adjacent only quadratic minimum spanning tree problem [24], the supply chain network problem [25], and so on [26–28].

Lagrangian relaxation algorithm can be adopted to solve multiple fault diagnosis models. Traditionally, the method of

updating Lagrange multiplier is based on the subgradient method. But the mathematical derivation process in the algorithm is complex. In addition, the updating method for Lagrangian multiplier is limited, and the traditional Lagrangian relaxation algorithm may fall into a local optimal solution. In [29], the Lagrange relaxation algorithm is improved and applied to multiple fault diagnosis problem, which brings about a better result. In this paper, the particle swarm optimization (PSO) algorithm will be adopted to update the Lagrange multiplier. The PSO algorithm is a global search algorithm [30, 31]. As a result, the PSO algorithm has a good adaptability and simplicity in practical applications [32–35]. We adopt the PSO algorithm to update the Lagrange multiplier in this paper.

In this paper, we propose an improved method for multiple fault diagnosis based on the Lagrangian relaxation algorithm and PSO algorithm. The updating of Lagrange multiplier will be designed as the optimization iteration process of particle swarm. A new method for updating of speed and location is proposed based on the nonnegative characteristic of Lagrange multiplier. The fitness function in the improved optimization algorithm is designed based on the absolute value of the difference between the upper and lower bounds. An experiment based on the system state diagnosis before a launch of Apollo spacecraft is adopted for verification of the proposed method.

The rest of this paper is organized as follows. The multiple fault diagnosis model is introduced in Section 2. In Section 3, the Lagrangian model is briefly introduced. Then, the improved Lagrange-particle swarm optimization algorithm is proposed in Section 4. Section 5 presents the simulation and verification experiment of the proposed method. Section 6 draws the conclusion of this paper.

2. Multiple Fault Diagnosis Model

The multiple fault diagnosis problem consists of the following elements. The fault source vector of the system can be denoted as $F = (f_1, f_2, \dots, f_m)$, where f_i represents the i th fault state of the system. The set of test points in the system is denoted as $T = (t_1, t_2, \dots, t_n)$, where t_j represents the j th test point. The fault source-test point dependency matrix is $D = [d_{ij}]$, where the element d_{ij} represents the dependency between the fault state f_i and the test point t_j . In addition, if the fault state and the test point are dependent on each other, then $d_{ij} = 1$; otherwise, $d_{ij} = 0$. The prior probability vector of the fault is $P = (p_1, p_2, \dots, p_m)$, where p_i represents the prior probability of the i th fault state.

Basing on the aforementioned factors, the process of multiple fault diagnosis can be described as follows. Basing on the test results, a set of possible fault sources $X \subseteq F$ can be calculated to maximize the posterior probability. The posterior probability can be converted to a prior probability with Bayesian formula [19]:

$$\max P(X | T_p, T_f) = \max \frac{P(T_p, T_f | X)P(X)}{P(T_p, T_f)}, \quad (1)$$

where T_p is the test set that has passed the test. T_f is the test set that has not passed the test. The vector x of m dimension is defined to represent the fault state of the component. When the fault state f_i occurs, $x_i = 1$; otherwise, $x_i = 0$. $P(T_p, T_f)$ is a constant that is independent of X . Thus, we can get the following function:

$$\begin{aligned} \max P(X | T_p, T_f) &= \max P(T_p, T_f | X)P(X) \\ &= \max P(T_p | X)P(T_f | X)P(X), \end{aligned} \quad (2)$$

where

$$P(X) = \prod_{i=1}^m P(s_i)^{x_i} (1 - P(s_i))^{(1-x_i)}, \quad (3)$$

where P_i is the prior probability of the failure in component i , calculating the probability of the occurrence $P(X)$ regarding the fault source set X . If the fault state f_i happens, $P_i^{x_i} = P_i$, $(1 - P_i)^{(1-x_i)} = 1$, then, do multiplication of the prior probability regarding the fault state f_i . Otherwise, the fault state f_i does not happen; do multiplication of $1 - P_i$, which is the prior probability of the fault state f_i that does not happen. The values of $P(T_p | X)$ and $P(T_f | X)$ are relative to the information in the matrix D [19]:

$$\begin{aligned} P(T_p | X) &= \prod_{t_j \in T_p} P(t_j \in T_p | X) = \prod_{t_j \in T_p} \prod_{i=1}^m (1 - d_{ij})^{x_i}, \\ P(T_f | X) &= \prod_{t_k \in T_f} P(t_k \in T_f | X) = \prod_{t_k \in T_f} \left[1 - \prod_{i=1}^m (1 - d_{ik})^{x_i} \right], \end{aligned} \quad (4)$$

where t_j and t_k denote the test points j and k respectively.

Obtaining the maximum value of $P(T_p | X)P(T_f | X)$, since they can be only 0 or 1, thus, both $P(T_p | X)$ and $P(T_f | X)$ must be 1. The following four cases exist:

- (i) If the test point t_j passes the test, for the faulty component $x_i = 1$, the test t_j must be independent with the component x_i ($d_{ij} = 0$)
- (ii) If the test point t_j passes the test, for the component without failure $x_i = 0$, the test t_j must be independent with the component x_i ($d_{ij} = 0$ or $d_{ij} = 1$)
- (iii) If the test point t_k fails to pass the test, for the component without failure $x_i = 0$, the test t_k must be independent with the component x_i ($d_{ik} = 0$ or $d_{ik} = 1$)
- (iv) If the test point t_k fails to pass the test, there is at least one faulty component $x_i = 1$ which makes the test t_k dependent on the fault component x_i ($d_{ik} = 1$)

According to the aforementioned analysis, a lemma can be summarized as follows [19].

Lemma 1. *The maximum value of $P(T_p | X)P(T_f | X)$ can be calculated on the condition that the following conditions are satisfied:*

- (1) If $t_j \in T_p$ and $x_i = 1$, then $d_{ij} = 0$
(2) If $t_k \in T_f$, there is at least one faulty component ($x_i = 1$) which makes the test t_k dependent on the fault component x_i , subsequently, $d_{ik} = 1$

To fulfil the aforementioned condition 1, delete the fault set that has been confirmed not to happen from the fault set C and get the fault set C^- . In other words, the value of the confirmed fault x_i that will not happen is 0, and the value of others is still unknown. Subsequently, condition 1) can be converted into the following constraints: for $t_k \in T_f$, $Ax \geq e$, where A can be calculated as the transposed matrix of D in which the test column without failure has been deleted. The size of A is $|T_f| \times m$. e is a unit vector of column.

Taking the negative natural logarithm of formula(3) and ignoring its constant, the problem can be converted as follows [19]:

$$\begin{cases} Z_{IP} = \min_{x \in C^-} \sum_{c_i \in C^-} b_i x_i \\ \text{s.t.} & Ax \geq e, \quad x_i \in \{0, 1\}, i = 1, \dots, m, \end{cases} \quad (5)$$

where $b_i = -\ln(p_i/1 - p_i), i = 1, \dots, m$.

3. Lagrangian Relaxation Model

The Lagrangian relaxation algorithm can be adopted to solve the aforementioned multiple fault diagnosis problem. The original multiple fault diagnosis model is shown as follows [19]:

$$\begin{cases} Z_{IP} = \min_{x \in S^-} \sum_{s_j \in S^-} c_j x_j \\ \text{s.t.} & Ax \geq e, \quad x_j \in \{0, 1\}, j = 1, \dots, m, \end{cases} \quad (6)$$

where S^- is the set of fault sources in which the fault from the set S that has been confirmed to never occur is removed. A is the matrix that is transposed from matrix D where the test column corresponding to the fault that did not occur has been removed. c_j depends on the prior probability $p(s_j)$, and c_j can be presented as follows:

$$c_j = -\ln \frac{p(s_j)}{1 - p(s_j)}, \quad j = 1, \dots, m. \quad (7)$$

Lagrangian relaxation algorithm is applied to the model. The improved optimization model is proposed as follows [19]:

$$\begin{cases} Z_{LR}(\lambda) = \min \sum_{j=1}^m l_j x_j + \sum_{i=1}^{|T_f|} \lambda_i \\ \text{s.t.} & x_j \in \{0, 1\}, j = 1, \dots, m, \end{cases} \quad (8)$$

where $l_j = c_j - \sum_{i=1}^{|T_f|} \lambda_i a_{ij}$, $\lambda_i \geq 0$, ($i = 1, \dots, |T_f|$) is the Lagrangian multiplier. Subsequently, the Lagrangian dual model can be represented as follows:

$$Z_{LD} = \max Z_{LR}(\lambda). \quad (9)$$

Finally, the fault source x can be obtained by solving the Lagrangian dual model Z_{LD} .

4. Lagrange-Particle Swarm Optimization Algorithm

The $Z_{LR}(\lambda)$ gradually approaches its optimal solution in the original problem by improving the value of Lagrangian multiplier λ during the process of solving the Lagrangian dual model Z_{LD} . The optimal solution of the problem can be calculated once the optimized value of Lagrangian multiplier λ is found out. In general, the method of updating multiplier λ in Lagrange relaxation algorithm is with the subgradient method, which means complex mathematical tool and theoretical analysis. Thus, the PSO algorithm is adopted to update the multiplier λ because it is not that complex and can solve the problem of optimization solution. Consequently, the PSO algorithm is combined with the Lagrangian relaxation model. In Lagrange-particle swarm optimization algorithm, the PSO algorithm is adopted to find the optimized value of Lagrangian multiplier λ . That is to say, the subgradient method is replaced by the PSO algorithm for calculating the Lagrangian multiplier λ .

The problem of defining a fitness function is involved in PSO algorithm. Here, the absolute value of the difference between upper and lower bounds J is chosen as the reference standard. The upper bound of the original multiple fault model (Z_{IP}) is introduced as a parameter Z_{UP} ; it can be replaced by the value of a feasible solution. Z_{DW} is a lower bound that is the nearest value to Z_{IP} for the function $Z_{LR}(\lambda)$. The absolute value of the difference between upper and lower bounds is $J = |Z_{UP} - Z_{DW}|$. The fitness function is $F = 1/1 + KJ$, where K is a scaling coefficient. If the particle λ reaches the optimal solution, then, the distance calculated by the particle between the upper bound Z_{UP} and lower bound Z_{DW} decreases. A smaller difference value between the upper and lower bounds means a bigger value of the fitness function, which shows the rationality of the designed fitness function F . In addition, the absolute value of the difference between upper and lower bounds J is a positive value; thus, the interval for the value of the fitness function F is $[0, 1]$. The normalization method is beneficial for the position updating of the particle λ and the searching for the global optimal particle. Consequently, the fitness function meets the requirement of the algorithm.

The flow chart of the algorithm is shown in Figure 1. It should be noted that the particle λ should be a nonnegative value for updating [19]. Consequently, the speed and position of the particle shall be limited during the updating process to slow down the speed of moving to the negative direction.

Defining Z_{DQ} as the optimal value for function $Z_{LR}(\lambda)$ for the current situation of λ . Above all, the Lagrange-particle swarm optimization algorithm can be summarized as follows:

- (i) Step 1. Initialization of parameters.

The initial population is $\lambda(0) = \{\lambda_1^0, \lambda_2^0, \dots, \lambda_g^0\}$, the size of the population is g , the best position for initial particle is $P_b(0) = \lambda(0) = \{\lambda_1^0, \lambda_2^0, \dots, \lambda_g^0\}$ and

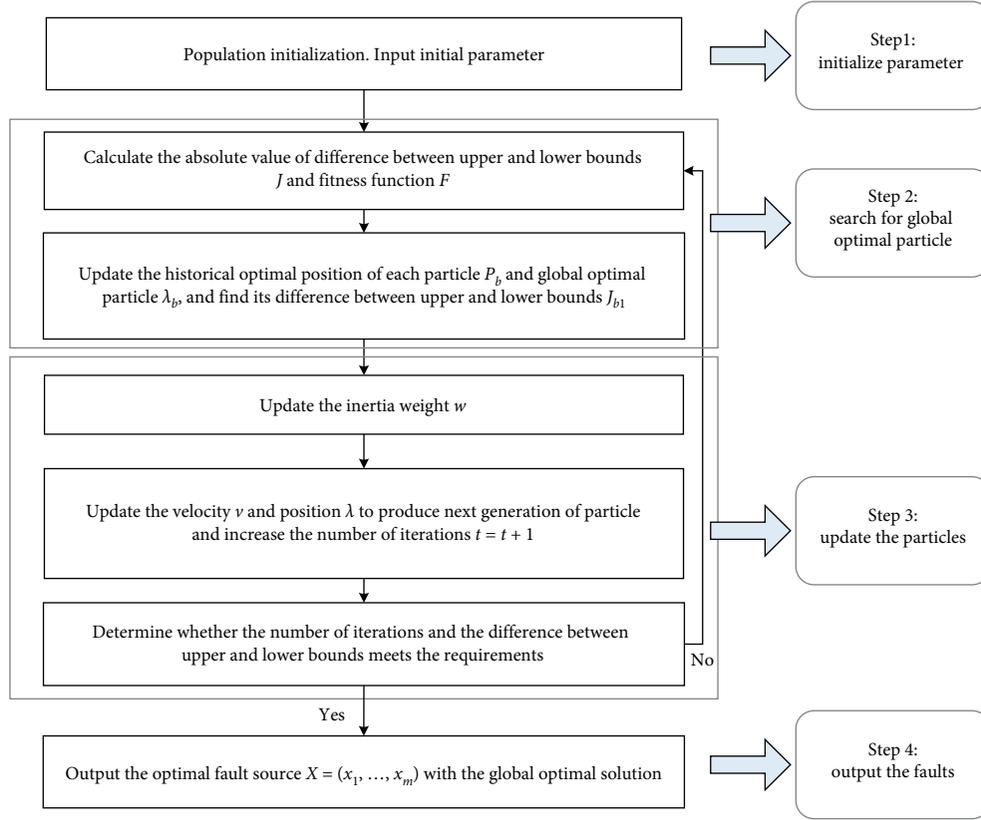


FIGURE 1: The flow chart of the Lagrange-particle swarm optimization algorithm.

the initial velocity vector is $v(0) = \{v_1^0, v_2^0, \dots, v_g^0\}$. The initial velocity is required to be between v_{\max} and v_{\min} . The optimal fitness function value for each initial particle is set as $F_b^0 = \{0, 0, \dots, 0\}$, where the dimension is g . The difference of the absolute value between the minimum upper and lower bounds for each initial particle is $J_b(0) = \{\infty, \infty, \dots, \infty\}$, where the dimension is g . The difference of the absolute value between the optimal upper and lower bounds for the overall initial situation is $J_{b1}^t = \infty$. Define $Z_{UP}(0) = \{\infty, \infty, \dots, \infty\}$ and $Z_{DW}(0) = \{-\infty, -\infty, \dots, -\infty\}$, where the dimension is g .

(iii) Step 2. Searching the global optimal particle.

Step 2.1. Do the following five steps for each element λ_i^t , ($i = 1, 2, \dots, g$) in the t th iteration of the particle denoted as $\lambda(t) = \{\lambda_1^t, \lambda_2^t, \dots, \lambda_m^t\}$.

- (1) Solving equation (8) with λ_i^t . The optimal value calculated by equation (8) is $Z_{DQ,i}^t$. Then, updating $Z_{DQ,i}^t = \max(Z_{DW,i}^t, Z_{DQ,i}^t)$.
- (2) Constructing a feasible solution vectors C_i^t for the original problem in equation (6) with the following four steps:

(a) Defining $C_i^t = [j | x_j = 1, j = 1, \dots, m]$, where the vector x is the calculation result of equation (8) by using λ_i^t . The value of C_i^t is chosen as the subscript j of the element in x corresponding to $x_j = 1$.

(b) For the row i uncovered in C_i^t (namely, $\sum_{j=1}^m a_{ij}x_j = 0$), add the corresponding value of j to C_i^t regarding $\min[j | a_{ij} = 1, c_j < \infty, j = 1, \dots, m]$.

(c) Checking all the subscript j satisfying $j \in C_i^t$. If $C_i^t - j$ is still the feasible solution of Z_{IP} , define $C_i^t = C_i^t - j$. This strategy is used for (1) removing unnecessary fault sources in the following application, (2) reducing the gap between $Z_{UP,i}^t$ and $Z_{DW,i}^t$, and (3) terminating the algorithm as soon as possible.

(d) Updating the $Z_{UP,i}^t = \min(Z_{UP,i}^t, \sum_{j \in C_i^t} c_j)$.

(3) Calculating the absolute value of the difference between the upper and lower bounds denoted as $J_i^t = |Z_{UP,i}^t - Z_{DW,i}^t|$.

(4) Calculating the value of the fitness function denoted as $F_i^t = 1/1 + KJ_i^t$.

(5) Basing on the fitness value of the global optimal value $F_{b,i}^t$ and the current generation's value F_i^t to determine whether to update the optimal position of the particle $P_{b,i}^t$. If $F_{b,i}^t < F_i^t$, update the position as $P_{b,i}^t = \lambda_i^t$, the value of fitness function as $F_{b,i}^t = F_i^t$, and the difference between upper and lower bounds as $J_{b,i}^t = J_i^t$; otherwise, update no parameter.

Step 2.2. Basing on the value of the fitness function $F_b(t) = \{F_{b,1}^t, F_{b,2}^t, \dots, F_{b,g}^t\}$, selecting

the global optimal particle λ_b^t in the optimal position $P_b(t) = \{P_{b,1}^t, P_{b,2}^t, \dots, P_{b,g}^t\}$ among all the particles. In addition, calculating the corresponding difference between the upper and lower bounds as $J_{b,1}^t$, which is also the global smallest difference value between the upper and lower bounds.

(iv) Step 3. Updating the particles.

Step 3.1. Updating the inertia weight $w^t = w_{\max} - t(w_{\max} - w_{\min})/N_{\max}$.

Step 3.2. Updating the velocity and position of the particles. The velocity of the i th particle in the d dimension for the $t+1$ times iteration is $v_{i,d}^{t+1} = w^t v_{i,d}^t + c_1 r_1^t (P_{b,i,d}^t - \lambda_{i,d}^t) + c_2 r_2^t (\lambda_{b,d}^t - \lambda_{i,d}^t)$. If the updated velocity is negative, defining $v = -K_1 e^{-(v+v_{\max})}$ to slow down the movement speed in the negative direction. This is because the Lagrangian multiplier is required to be a nonnegative value. The updated position is $\lambda_{i,d}^{t+1} = \lambda_{i,d}^t + v_{i,d}^{t+1}$. If $\lambda_{i,d}^{t+1} < 0$, setting $\lambda_{i,d}^{t+1} = 0$. Updating the time of the iteration as $t = t + 1$.

Step 3.3. Determining whether the maximum number of the iterations N_{\max} has been reached. If $t > N_{\max}$ or the absolute value ($J_{b,1}^t$) of the difference between the upper and lower bounds obtained by calculating the global optimal particle λ_b^t is less than the preset threshold, then, continue and turn to Step 4; else, turn to Step 2.

(v) Step 4. Solving equation (8) with the global optimal solution λ_b^t . Outputting the fault source $X = (x_1, \dots, x_m)$.

The proposed Lagrange-particle swarm updating algorithm inherits the advantages in PSO algorithm and Lagrangian relaxation model. On one hand, the objective function of the proposed algorithm integrates the complex constraint conditions, which makes it easier for solving the multiple fault diagnosis. On the other hand, the subgradient method is replaced by the PSO algorithm for calculating the Lagrangian multiplier λ , which simplifies the theoretical analysis and solving process. It should be noted that the proposed algorithm has more than one feasible solution, which shows priority over the original Lagrange relaxation algorithm. While using PSO to update the Lagrangian multiplier λ , there are many alternative particles except for the optimal ones. This contributes to more alternative fault diagnosis strategies in solving multiple fault diagnosis problem, which makes the proposed method more suitable for locating the multiple fault types in fault diagnosis.

5. Experiment and Discussion

To test the performance of Lagrange-particle swarm updating algorithm in multiple fault diagnosis problem, an example of system state diagnosis before the launch of

Apollo spacecraft was adopted from [36, 37]. In the case study, 10 fault states and 15 sensor test points are known and the prior probabilities among all the fault states are assumed to be equal. The dependency matrix of the system is shown in Table 1.

Establishing the corresponding Lagrangian dual model, the proposed Lagrange-particle swarm optimization algorithm is adopted to solve the model. The solving of the model is by programming in MATLAB R2014a. The parameters in the algorithm are chosen as follows. The maximum iteration number $N_{\max} = 200$, population size $\text{popsize} = 50$, acceleration factor $c_1 = 1.3$ and $c_2 = 1.5$, maximum of inertia weight $w_{\max} = 1.2$, minimum inertia of weight $w_{\min} = 0.4$, maximum of velocity $v_{\max} = 2$, minimum of velocity $v_{\min} = -2$, proportional coefficient $K = 5$ and $K_1 = 0.2$, and the absolute value of the minimum difference between upper and lower bounds $J_{\min} = 0.01$. J_{\min} affects the quality of solution and the calculating time significantly. If J_{\min} is too small, it may lead to a failure of the solution to be less than the threshold even after the solution reached the maximum number of iterations, which means a high but unnecessary requirement on the quality of solution. On the contrary, if J_{\min} is too big, the number of iterations may be too less, which may lead to a bad quality of solution. After many times of trial, J_{\min} is finally to be chosen as 0.01.

Take the double-fault diagnosis problem as an example, the results are shown in Table 2. The corresponding fault isolation rate is 91.11%. The proposed method has a right diagnosis result on almost all the double-fault types. As for the fault No. (4 and 6), the proposed method gives a solution as (4, 5, and 6), which includes all the potential fault elements 4, 5, and/or 6. In this case, the fault diagnosis with the proposed method eliminates all possible faults without the missed one. Similar situation also exists in the fault No. (5 and 6), (5 and 8), and (5 and 10) according to Table 2.

Adopting the case study in [19] where Lagrange relaxation algorithm is proposed for the diagnosis of double-fault types, the failure states are not recognized with the fault No.: (1) 3 and 5, (2) 6 and 8, (3) 8 and 9, (4) 4 and 6, (5) 5 and 6, (6) 5 and 10, and (7) 5 and 8. The experiment results with the original Lagrange relaxation algorithm and the improved algorithm are shown in Table 3.

According to Table 3, the fault status exists in fault No. (3 and 5), and the diagnosis result with the original Lagrange relaxation algorithm is fault No. (1, 3 and 4), while the result with the proposed method is fault No. 3 and 5. The proposed method can successfully find the right fault types. In addition, if the fault status exists in fault No. (4 and 6), (5 and 6), the improved method can locate all the potential faults denoted as No. (4, 5 and 6), which can ensure that all potential fault statuses can be located during the fault diagnosis process. For the fault No. (3 and 5), (6 and 8), and (8 and 9), the proposed method gets accurate and right solution, while the method with only Lagrange relaxation algorithm gets wrong diagnosis results where some elements with no fault are also included in the fault diagnosis reports (e.g., element No. 1 and 2). As for the fault No. (4 and 6), (5 and 6), (5 and 8), and (5 and 10), the original Lagrange relaxation algorithm cannot distinguish the fault status with the same alarm

TABLE 1: The dependency matrix for fault status and test in the system of Apollo spacecraft.

Fault no.	Test point															Prior probability
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
1	0	0	0	1	0	0	0	1	0	1	1	1	1	0	0	0.1
2	0	0	1	0	1	0	0	0	0	0	1	1	0	1	0	0.1
3	0	0	0	0	1	1	1	0	1	1	1	1	0	0	1	0.1
4	0	1	0	0	0	1	1	0	0	0	0	0	0	1	1	0.1
5	0	1	0	1	0	1	1	1	1	1	0	0	1	1	0	0.1
6	0	0	0	1	1	0	0	1	1	1	0	1	1	1	1	0.1
7	1	0	0	1	1	0	0	0	0	0	0	1	0	1	1	0.1
8	1	1	1	0	0	1	1	0	1	1	1	0	0	1	0	0.1
9	1	0	0	1	1	0	0	0	0	0	0	1	0	1	1	0.1
10	1	1	1	1	0	0	1	0	0	0	1	0	1	1	0	0.1

TABLE 2: The results of double-fault diagnosis with Lagrange relaxation-particle swarm optimization algorithm.

Fault no.	Sensor report no.	Diagnosis results
1, 2	3, 4, 5, 8, 10, 11, 12, 13, 14	1, 2
1, 3	4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15	1, 3
1, 4	2, 4, 6, 7, 8, 10, 11, 12, 13, 14, 15	1, 4
1, 5	2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14	1, 5
1, 6	4, 5, 8, 9, 10, 11, 12, 13, 14, 15	1, 6
1, 7	1, 4, 5, 8, 10, 11, 12, 13, 15	1, 7
1, 8	1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14	1, 8
1, 9	1, 4, 5, 8, 11, 12, 13, 14, 15	1, 9
1, 10	1, 2, 3, 4, 7, 8, 10, 11, 12, 13, 14	1, 10
2, 3	3, 5, 6, 7, 9, 10, 11, 12, 14, 15	2, 3
2, 4	2, 3, 5, 6, 7, 11, 12, 14, 15	2, 4
2, 5	2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14	2, 5
2, 6	3, 4, 5, 8, 9, 10, 11, 12, 13, 14, 15	2, 6
2, 7	1, 3, 4, 5, 8, 11, 12, 13, 14, 15	2, 7
2, 8	1, 2, 3, 5, 6, 7, 9, 10, 11, 12, 14	2, 8
2, 9	1, 3, 4, 5, 11, 12, 14, 15	2, 9
2, 10	1, 2, 3, 4, 5, 7, 11, 12, 13, 14	2, 10
3, 4	2, 5, 6, 7, 9, 10, 11, 12, 14, 15	3, 4
3, 5	2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	3, 5
3, 6	4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	3, 6
3, 7	1, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15	3, 7
3, 8	1, 2, 3, 5, 6, 7, 9, 10, 11, 12, 14, 15	3, 8
3, 9	1, 4, 5, 6, 7, 9, 10, 11, 12, 14, 15	3, 9
3, 10	1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15	3, 10
4, 5	2, 4, 6, 7, 8, 9, 10, 13, 14, 15	4, 5
4, 6	2, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15	4, 5, 6
4, 7	1, 2, 4, 5, 6, 7, 8, 11, 13, 14, 15	4, 7
4, 8	1, 2, 3, 6, 7, 9, 10, 11, 14, 15	4, 8
4, 9	1, 2, 4, 5, 6, 7, 12, 14, 15	4, 9
4, 10	1, 2, 3, 4, 6, 7, 11, 13, 14, 15	4, 10
5, 6	2, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15	4, 5, 6
5, 7	1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15	5, 7
5, 8	1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 13, 14	5, 8, 10
5, 9	1, 2, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15	5, 9
5, 10	1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 13, 14	5, 8, 10
6, 7	1, 4, 5, 8, 9, 10, 11, 12, 13, 14, 15	6, 7
6, 8	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	6, 8
6, 9	1, 4, 5, 8, 9, 10, 12, 13, 14, 15	6, 9
6, 10	1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15	6, 10
7, 8	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15	7, 8
7, 9	1, 4, 5, 8, 11, 12, 13, 14, 15	7, 9
7, 10	1, 2, 3, 4, 5, 7, 8, 11, 13, 14, 15	7, 10
8, 9	1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 14, 15	8, 9
8, 10	1, 2, 3, 4, 6, 7, 9, 10, 11, 13, 14	8, 10
9, 10	1, 2, 3, 4, 5, 7, 11, 12, 13, 14, 15	9, 10
Fault isolation rate		91.11%

TABLE 3: The experiment results with the Lagrange relaxation algorithm and the proposed method.

Fault	Sensor report no.	Lagrange algorithm	Proposed method
3, 5	2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	1, 3, 4	3, 5
6, 8	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	2, 3, 4, 7	6, 8
8, 9	1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 14, 15	2, 3, 4, 9	8, 9
4, 6	2, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15	4, 6	4, 5, 6
5, 6	2, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15	4, 6	4, 5, 6
5, 10	1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 13, 14	5, 8	5, 8, 10
5, 8	1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 13, 14	5, 8	5, 8, 10

sensor report numbers, while the proposed method reports the diagnosis results including all the fault elements without missing fault type or including normal element. The fault isolation rate of the improved algorithm for double-fault diagnosis reaches 91.11% which is higher than that of the literature [19] (where the fault isolation rate is 88.89%). In addition, the troubleshooting rate with the proposed method reaches 100% in this case. Above all, the proposed method has a better performance on the fault isolation rate and the troubleshooting rate which is helpful for solving the problem of multiple fault diagnosis in complex equipment.

Simulation experiment result shows that the fault isolation rate of the proposed method is higher than that of the original Lagrange relaxation algorithm. The proposed method has a higher accuracy in fault diagnosis. This is because the Lagrangian multiplier in the original Lagrange relaxation algorithm has been improved. In the proposed method, (1) the PSO algorithm is adopted to update the Lagrangian multiplier, and (2) the fitness function is constructed by using the difference between upper and lower bounds. With PSO, the initial particle and its iterative updating process have the feature of a random process; consequently, a global search can be achieved and contribute to more choices for selecting particles.

6. Conclusions

In this paper, the Lagrange-particle swarm optimization algorithm is proposed to solve the multiple fault diagnosis model. Being different from the original subgradient ascending method, the Lagrangian multiplier in the original Lagrangian relaxation algorithm is updated by the PSO algorithm, which makes it easier for theoretical analysis and algorithm implementation. The fitness function is constructed based on the absolute value of the difference between the upper and lower bounds. The new fitness function can be a main adjusted index that is responsible for the effectiveness of the multiplier. By controlling the speed of updating speed to the negative direction, we can improve the search performance in the proposed approach. The effectiveness of the new method is verified according to the simulation experiment, and the superiority is also discussed in comparison with the original algorithm.

In the following work, the reliability of the test can be taken into consideration. The multiple fault diagnosis model may include the occurrence of failure in sensors. Currently, it is assumed that the test of fault detection is reliable. If a fault occurs, the fault can be detected by the test safely and

reliably. However, in practical system, the fault detection rate may not reach 100%, because the test is composed of several sensors, while the sensor itself may fail.

Data Availability

All relevant data are provided within the paper.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] H. Liu and J. Xiang, "Autoregressive model-enhanced variational mode decomposition for mechanical fault detection," *IET Science, Measurement & Technology*, vol. 13, no. 6, pp. 843–851, 2019.
- [2] S. Wang, J. Xiang, H. Tang, X. Liu, and Y. Zhong, "Minimum entropy deconvolution based on simulation-determined band pass filter to detect faults in axial piston pump bearings," *ISA Transactions*, vol. 88, pp. 186–198, 2019.
- [3] D. Wu and Y. Tang, "An improved failure mode and effects analysis method based on uncertainty measure in the evidence theory," *Quality and Reliability Engineering International*, pp. 1–22, 2020.
- [4] L. Wang, J. Xiang, and Y. Liu, "A time-frequency-based maximum correlated kurtosis deconvolution approach for detecting bearing faults under variable speed conditions," *Measurement Science and Technology*, vol. 30, no. 12, p. 125005, 2019.
- [5] Y. Wang, L. Yang, J. Xiang, J. Yang, and S. He, "A hybrid approach to fault diagnosis of roller bearings under variable speed conditions," *Measurement Science and Technology*, vol. 28, no. 12, 2017.
- [6] X. Lv, D. Zhou, L. Ma, and Y. Tang, "Dependency model-based multiple fault diagnosis using knowledge of test result and fault prior probability," *Applied Sciences*, vol. 9, no. 2, p. 311, 2019.
- [7] B. Jiang, J. Xiang, and Y. Wang, "Rolling bearing fault diagnosis approach using probabilistic principal component analysis denoising and cyclic bispectrum," *Journal of Vibration and Control*, vol. 22, no. 10, pp. 2420–2433, 2016.
- [8] S. Wang, J. Xiang, Y. Zhong, and Y. Zhou, "Convolutional neural network-based hidden Markov models for rolling element bearing fault identification," *Knowledge-Based Systems*, vol. 144, no. 65–76, 2018.
- [9] D. Kayrbekova, A. Barabadi, and T. Markeset, "Maintenance cost evaluation of a system to be used in arctic conditions: a case study," *Journal of Quality in Maintenance Engineering*, vol. 17, no. 4, pp. 320–336, 2011.

- [10] P. Gupta, S. Gupta, and O. P. Gandhi, "Annual maintenance budget estimation for a plant system," *Journal of Quality in Maintenance Engineering*, vol. 20, no. 2, pp. 193–210, 2014.
- [11] C. He Chang and P. Yang Jiang, P. Ju Cao, "Research on testability evaluation method of complex equipment based on test data in development phase," in *Advanced Materials Research*, vol. 301–303, pp. 913–918, Trans Tech Publ, 2011.
- [12] X. Zhang, K. Huang, P. Yan, and G. Lian, "Hierarchical hybrid testability modeling and evaluation method based on information fusion," *Journal of Systems Engineering and Electronics*, vol. 26, no. 3, pp. 523–532, 2015.
- [13] T. Li, Y. Li, Y. Qian, and Y. Xu, "Optimizing reliability, maintainability and testability parameters of equipment based on gspn," *Journal of Systems Engineering and Electronics*, vol. 26, no. 3, pp. 633–643, 2015.
- [14] M. Efatmaneshnik, M. J. Ryan, and S. Shoval, "A framework for testability analysis from a systems architecture perspective," *Insight*, vol. 21, no. 3, pp. 72–79, 2018.
- [15] S. Wang, J. Xiang, Y. Zhong, and H. Tang, "A data indicator-based deep belief networks to detect multiple faults in axial piston pumps," *Mechanical Systems and Signal Processing*, vol. 112, pp. 154–170, 2018.
- [16] W. Song, J. Xiang, and Y. Zhong, "A simulation model based fault diagnosis method for bearings," *Journal of Intelligent & Fuzzy Systems*, vol. 34, no. 6, pp. 3857–3867, 2018.
- [17] X. Lv, D. Zhou, Y. Tang, and L. Ma, "An improved test selection optimization model based on fault ambiguity group isolation and chaotic discrete pso," *Complexity*, vol. 2018, Article ID 3942723, 10 pages, 2018.
- [18] M. Amit, S. Ghoshal, D. Haste et al., "An integrated support system for rotorcraft health management and maintenance," in *Proceedings of the 2000 IEEE Aerospace Conference Proceedings (Cat. No. 00TH8484)*, vol. 6, pp. 1–8, IEEE, Big Sky, MT, USA, March 2000.
- [19] L. Ke, B. Long, and H. J. Wang, "A fault diagnosis approach of the large complex system based on bayes theory," *Acta Armamentarii*, vol. 29, no. 3, pp. 352–356, 2008.
- [20] V. Neven, P. Kovacki, M. Vidovi'c, and A. T. Sari'c, "Scalable algorithm for the dynamic reconfiguration of the distribution network using the Lagrange relaxation approach," *International Journal of Electrical Power & Energy Systems*, vol. 94, pp. 188–202, 2018.
- [21] Z. Rafie-Majid, S. H. R. Pasandideh, and B. Naderi, "Modelling and solving the integrated inventory-location-routing problem in a multi-period and multi-perishable product supply chain with uncertainty: Lagrangian relaxation algorithm," *Computers & Chemical Engineering*, vol. 109, no. 9–22, 2018.
- [22] O. G. Czibula, H. Gu, and Y. Zinder, "Lagrangian relaxation versus genetic algorithm based metaheuristic for a large partitioning problem," *Theoretical Computer Science*, vol. 718, pp. 24–36, 2018.
- [23] S. Emami, M. Sabbagh, and G. Moslehi, "A Lagrangian relaxation algorithm for order acceptance and scheduling problem: a globalised robust optimisation approach," *International Journal of Computer Integrated Manufacturing*, vol. 29, no. 5, pp. 535–560, 2016.
- [24] D. L. Pereira and A. S. da Cunha, "Polyhedral results, branch-and-cut and Lagrangian relaxation algorithms for the adjacent only quadratic minimum spanning tree problem," *Networks*, vol. 71, no. 1, pp. 31–50, 2018.
- [25] M. Kheirabadi, B. Naderi, A. Arshadikhamseh, and V. Roshanaei, "A mixed-integer program and a Lagrangian-based decomposition algorithm for the supply chain network design with quantity discount and transportation modes," *Expert Systems with Applications*, vol. 137, pp. 504–516, 2019.
- [26] S. Yamada and A. Takeda, "Successive Lagrangian relaxation algorithm for nonconvex quadratic optimization," *Journal of Global Optimization*, vol. 71, no. 2, pp. 313–339, 2018.
- [27] L. Ping, K. Xiangrui, F. Chen, and Y. Zheng, "Novel distributed state estimation method for the ac-dc hybrid microgrid based on the Lagrangian relaxation method," *The Journal of Engineering*, vol. 2019, no. 18, pp. 4932–4936, 2019.
- [28] M.-C. Tsai, F. Y.-S. Lin, and Y.-F. Wen, "Lagrangian-relaxation-based self-repairing mechanism for wi-fi networks," *IEEE Access*, vol. 7, pp. 15868–15883, 2019.
- [29] J.-c. Huang, Y. Tao, L. You, and S. Jia-you, "Improved Lagrange relaxation algorithms and its application in multiple fault diagnostic strategy," *Computer Engineering and Design*, vol. 38, no. 4, pp. 1040–1044, 2017.
- [30] S. D. Chavan and N. P. Adgokar, "An overview on particle swarm optimization: basic concepts and modified variants," *International Journal of Science and Research*, vol. 4, no. 5, pp. 255–260, 2015.
- [31] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview," *Soft Computing*, vol. 22, no. 2, pp. 387–408, 2018.
- [32] Q.-b. Zhang, P. Wang, and Z.-h. Chen, "An improved particle filter for mobile robot localization based on particle swarm optimization," *Expert Systems with Applications*, vol. 135, 2019.
- [33] Z. Xiang, D. Ji, H. Zhang, H. Wu, and Y. Li, "A simple pid-based strategy for particle swarm optimization algorithm," *Information Sciences*, vol. 502, 2019.
- [34] M. Imran, R. Hashim, and N. Khalid, "An overview of particle swarm optimization variants," *Procedia Engineering*, vol. 53, pp. 491–496, 2013.
- [35] Y. Mao, C. Liu, X. Dong, J. Wang, and T. Ba, "Study of the magnetic properties of haematite based on spectroscopy and the ipso-elm neural network," *Journal of Sensors*, vol. 2018, Article ID 6357905, 9 pages, 2018.
- [36] J. G. Wohl, "Information automation and the apollo program: a retrospective," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 12, no. 4, pp. 469–478, 1982.
- [37] S. Deng, B. Jing, and H. Zhou, "Heuristic particle swarm optimization approach for test point selection with imperfect test," *Journal of Intelligent Manufacturing*, vol. 28, no. 1, pp. 37–50, 2017.