An Improved Multi-objective Evolutionary Algorithm Based on Gaussian Distribution Estimation

Li-yang HOU¹, Xiao-yong LI¹,* , Yan-rong LI¹, Wen-ping KONG¹ and Hai-feng CHANG²

¹Key Laboratory of Trustworthy Distributed Computing and Service, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China

²National Engineering Laboratory for Integrated Command and Dispatch Technology

*Corresponding author

Keywords: Potential disaster sites, Resource scheduling, Multi-objective optimization, NSGA-III-GD.

Abstract. When an emergency occurs, how to specify a reasonable resource scheduling scheme significantly affects disaster relief efficiency. However, most actual existing schemes lack considering satisfaction of potential disaster sites, and lack a scheduling model with 3 or more optimization goals, which makes it difficult to apply to complex scenarios. In this paper, we propose a four-objective resource scheduling optimization model that additionally considers potential disaster sites satisfaction. And we have designed an improved NSGA-III-GD algorithm to optimize this model. First, we introduce NSGA-III, an algorithm that has a great advantage in multi-objective optimization problems. And more importantly, we use Gaussian estimation distribution instead of traditional cross mutation operators to extract the overall characteristics of the population, which improves the search accuracy of the optimal solution and greatly improves the convergence speed. The experimental results clearly show that the algorithm proposed in this paper has achieved very good performance.

Introduction

In recent years, emergencies such as natural disasters, traffic accidents, and fires have frequently occurred, which have caused serious harm to human production and life. Therefore, it is very important to designate an effective and reasonable emergency resource scheduling scheme to reduce various losses at the first time of these sudden events[1]. However, the time, place and degree of disasters have strong randomness and linkage. How to mobilize existing resources to the disaster sites in the shortest time and at the lowest cost requires emergency resource scheduling technology, which is based on system resources sharing and dynamic dispatching model of region linkage[2]. As the optimal allocation and scheduling of emergency resources become more and more important, more and more social attention has been paid to it, and research results in this area have continued to emerge. However, among the existing results, there is a lack of research on resource scheduling models with 3 or more optimization goals, and the possibility of potential disaster sites is also ignored. Obviously, these issues also need urgent consideration.

Combined with the actual resource scheduling cases, in this paper we propose a four-objective optimization model that minimizes time costs, minimizes scheduling costs, minimizes loss of disaster sites and maximizes satisfaction of potential disaster sites to build a resource scheduling model, which meets more complex emergency scenarios and solves the issues mentioned above; and we introduce the NSGA-III[3] algorithm, which has a great advantage in optimization of multiple objectives. Not only that, we also use the Gaussian estimation distribution to improve the original
algorithm. Therefore, a new non-dominated sorting algorithm NSGA-III-GD is designed, which greatly increases the iteration speed of the algorithm to give the optimal scheduling scheme.

**Relation Work**

At present, in solving the problem of constructing an emergency resource scheduling scheme for an emergency, it is mainly using optimization modeling methods. First, set multiple optimization goals for a specific event, and then select the appropriate multi-objective optimization algorithm to build a model. And the following are existing popular optimization modeling methods:

---

**Genetic Algorithm.** Solving optimization problems based on genetic algorithm. [4] is the simplest, feasible and most widely used technical solution. Darwin's theory of population evolution and survival of the fittest was used to simulate the evolution of living things and find the best solution in the solution space. However, the implementation of genetic algorithm is relatively complicated, the search speed of the solution space is relatively slow, and it is easy to cause the problem of too fast convergence speed. [5] used an adaptively mutate genetic algorithm to optimize emergency resources scheduling model, which is adjusting to a practical situation. [6] presented a genetic algorithm by establishing an optimization model using the random walk method, which is to minimize delays in emergency response.

**Memetic Algorithm.** Memetic genetic algorithm [7] is a combination of population-based global search and individual-based local heuristic search. [8] created a taboo search method to solve multi-objective combinatorial optimization problems, thereby eliminating two main problems of this multi-objective method. [9] presented emergency resource scheduling optimization model with uncertainty and used simulated annealing algorithm to speed up convergence.

**Multi-objective Optimization Algorithm.** When building a resource scheduling model, there are often multiple goals to optimize. For such problems, there are several best solutions, but they need to be grouped. This group of optimal solutions is usually referred to as one of a non-dominant solution set or a Pareto solution set. Now, how to solve high-dimensional multi-objective optimization (MOO) problems (generally refers to multi-objective optimization problems with more than 3 targets) has become a problem faced in the evolutionary multi-objective field for a period of time. Multi-objective optimization algorithms can be divided into the following categories:

1. **MOO based on particle swarm optimization.**

   [10] created an optimized model for particle heat by linearly adjusting the learning coefficients and inertial weights, increasing local perturbations and minimizing both transport costs and time delays. [11] considered the rescue cost from emergency system and the losses caused by not be rescued in time to solve a multi-objective emergency-resource scheduling problem on particle swarm algorithm. [12] introduced a two-way ant mechanism to speed up the search of optimal solutions.

2. **MOO based on distribution estimation.**

   [13] proposed a multi-objective optimization model based on a hybrid Gaussian model, combining supervised classification and lightweight proxy models to achieve individual selection of offspring.

3. **MOO based on decomposition.**

   [14] created a multi-objective model of the multi-period dynamic emergency resource scheduling problem based on decomposition (MOEA/D). [15] proposed to decompose the multi-objective optimization problem into multiple scalar optimization sub-problems to reduce the computational complexity, but the algorithm does not have a normalization mechanism for the target space. [16] in
order to solve the problem that non-dominated schemes cannot be evenly distributed and the solution search area is reduced, a bidirectional decomposition-based method is proposed for multi-objective optimization.

The Problem Description and the Model

Problem Description. The construction of emergency resource scheduling schemes is a very complicated process. It is necessary to construct a reasonable and effective scheduling model for specific events. Due to the uncertainty of emergencies, these events may occur in multiple sites. At the same time, due to the linkage of emergencies, many potential locations may be affected by the current disaster sites. This requires consideration of the common needs of the disaster sites and the potential disaster sites. It also leads to that the optimization goals of the resource scheduling model may reach 3 or more. Therefore, we need to fully consider the optimization goals of the model when constructing the resource scheduling model. In this paper, we aim to build a common demand that can satisfy both the disaster sites and potential disaster sites, with the goal of minimizing scheduling time, minimizing scheduling cost, minimizing disaster sites loss, and maximizing potential disaster sites satisfaction. So the resource scheduling model can meet the needs of emergencies in real life to the greatest extent. Figure 1 shows the scenario.

![Figure 1. Rescue point to multiple disaster sites and multiple potential disaster sites scenario.](image)

Model Assumption. Before starting to introduce the model building, let's make the following assumptions:

- Consider the scenario from a single rescue site to multiple disaster sites and multiple potential disaster sites;
- The distance from the rescue site to each disaster site and the speed of resource scheduling are known;
- The demand for different resources at each disaster site is known;
- The scheduling cost and loss of each resource at the disaster site are known during the scheduling process.

Model Construction. Our variable settings for this model are as follows:

Set S as the rescue point; R\_1, R\_2, \ldots, R\_r are R resources; N\_1, N\_2, \ldots, N\_n are N disaster sites; NR\_ik represents the demand for k-th resource at i-th disaster site; D\_i represents the distance from rescue point to i-th disaster site; v\_i represents the resource scheduling speed from rescue point to i-th disaster
site; \( v_{\text{min}} \) represents the minimum speed of resource scheduling, and \( v_{\text{max}} \) represents the maximum speed of resource scheduling; \( v_j \) is a random value between \( v_{\text{min}} \) and \( v_{\text{max}} \); \( P_{N1}, P_{N2}, ..., P_{Nm} \) are M potential disaster sites; \( P_1, P_2, ..., P_m \) are the probability of emergencies of M potential disaster sites; \( PNR_{jk} \) represents the demand for \( k \)-th resource at \( j \)-th potential disaster site; \( c_k \) represents the unit scheduling cost of \( k \)-th resource; \( t_k \) represents the unit scheduling time of \( k \)-th resource; And the scheduling result is expressed by a matrix \( X \), \( X_{ik} \) represents the \( k \)-th resource quantity that rescue point dispatches to \( i \)-th disaster site.

\[
\min \ F(X) = (f_1(X), f_2(X), f_3(X), f_4(X))^T
\]

\[
f_1(X) = \sum_{i=1}^{N} \sum_{j=1}^{R} t_i * X_{ik} * \frac{D_i}{v_i}
\]

\[
f_2(x) = \sum_{i=1}^{N} \sum_{k=1}^{R} c_t * X_{jk}
\]

\[
f_3(x) = \sum_{i=1}^{N} \sum_{k=1}^{R} L_k * (NR_{ik} - X_{ik})
\]

\[
f_4(x) = \sum_{k=1}^{R} \frac{R_k - \sum_{i=1}^{N} X_{ik}}{\sum_{j=1}^{M} PNR_{jk} * P_j}
\]

s.t.

\[
0 \leq X_{ik} \leq NR_{ik}, i = 1,2, ..., N; k = 1,2, ..., R
\]

\[
\sum_{j=1}^{M} PNR_{jk} \leq PR_k, k = 1,2, ..., R
\]

Equation (1) shows the overall optimization goal of emergency resource scheduling model.

Equation (2), (3), (4), (5) shows the four optimization goals proposed in this article: scheduling time, scheduling cost, disaster sites loss, and potential disaster sites satisfaction.

Equation (6) shows the constraints on the amount of resources allocated to disaster sites by rescue point, which are non-negative and no higher than the demand of the disaster needs.

Equation (7) shows the amount of resources allocated to disaster sites by rescue point must also meet the constraint of not being higher than the total resources of rescue point.

**Model Optimization Algorithm**

This article proposes the following improvements in existing resource scheduling schemes:

1) Consider the linkage of emergencies, and consider the needs of potential disaster sites in the scheduling model;

2) Consider the uncertainty of actual emergencies, and proposed a solution with more than 3 optimization goals in the scheduling model;

3) When existing multi-objective optimization algorithms deal with more than three objective functions, there are many problems, which are mainly reflected in the following aspects:

a) The increase in the number of optimization goals leads to an increase in the proportion of non-dominated solutions in the population, and the process of searching for solutions will slow down;
b) In the case of high-dimensional target spaces, the computational complexity of maintaining the diversity index is too high to find the solution's peripheral elements;

c) In the case of a high-dimensional object space, the search function of recombination operators is inefficient.

4) Based on NSGA-III algorithm, the Gaussian estimation distribution method is used to solve the problem because the original algorithm ignores the global characteristics of the entire population, and the newly proposed algorithm improves the optimal solution convergence speed and solution space search accuracy.

![Figure 2. Left: Hyperplane composed of widely distributed reference points; Right: Selection of population using reference points and ideal point.](image)

In this new model, we utilize the advantages of NSGA-III algorithm in high-dimensional target space. We use widely distributed reference points (Figure 2: Left) to maintain the diversity of individuals in the population; and use the connection between ideal points and reference points when selecting individuals in the hyperplane constructed by the optimization target (objective function), that is, choose the nearest point(individual) from the line consisting of the ideal point and the reference points (Figure 2: Right), which solves the problem of low optimal solution search ability[3, 17]. In the use of this algorithm, we found that the algorithm uses the traditional cross mutation operator to randomly select individuals in the parent population. This operation obviously only considers the local characteristics of the individual. Therefore, we have designed a new algorithm NSGA-III-GD based on this. Instead, we select non-dominant individuals in each population, count overall characteristics, and use Gaussian estimation methods to generate a new generation of population. This can greatly improve the search efficiency and accuracy of optimal solutions.

**NSGA-III-GD Algorithm**

Let us consider t-th generation of this algorithm. Assuming that the parent population is \( P_t \), the first step is to calculate the fitness \( F_t = [f_1(P_t), f_2(P_t), f_3(P_t), f_4(P_t)] \) of the population and perform non-dominated sorting. Then, use the information of highest-ranking individuals to calculate the mean and covariance of Gaussian distribution, and the offspring individuals are generated by this distribution estimation method; Next, use polynomial mutation[18] to mutate newly generated individuals; Next, calculate the minimum fitness \( Z_{\min} = (Z_{1\min}, Z_{2\min}, \ldots, Z_{n\min}) \) using the parents \( P_t \) and offspring individuals \( G_t \), where \( Z_{n\min} \) is the minimum fitness in n-th objective function dimension, and non-dominantly sort \( G_t \) and \( F_t \); Finally, use the generated reference points and the minimum fitness \( Z_{\min} \) for individual selection to get child population \( P_{t+1} \). The specific algorithm flow is shown in the table below:
Algorithm 1 NSGA-III-GD Algorithm.

Input: Population Size S, Generation g
Parameter: Objective-Function Nums n, X dimension d
Output: Optimal Solution P<sub>g</sub>
1. Reference Points rp = uniformpoint(S, n)
2. Population P<sub>0</sub> = initialSolutions(rp.size, d)
3. Fitness F<sub>0</sub> = [f<sub>1</sub>(P<sub>0</sub>), f<sub>2</sub>(P<sub>0</sub>), f<sub>3</sub>(P<sub>0</sub>), f<sub>4</sub>(P<sub>0</sub>)]
4. For t: 0 → g do
5. ShuffleSample(P<sub>t</sub>)
6. G<sub>t</sub> = GaussianEDASample(P<sub>t</sub>)
7. PolyMutatiohn(G<sub>t</sub>)
8. Fitness GF<sub>t</sub> = [f<sub>1</sub>(G<sub>t</sub>), f<sub>2</sub>(G<sub>t</sub>), f<sub>3</sub>(G<sub>t</sub>), f<sub>4</sub>(G<sub>t</sub>)]
9. SET Z<sub>min</sub> = (Z<sub>1</sub><sub>min</sub>, Z<sub>2</sub><sub>min</sub>, ..., Z<sub>n</sub><sub>min</sub>) in (GF<sub>t</sub>, F<sub>t</sub>)
10. (L<sub>1</sub>, L<sub>2</sub>, ..., L<sub>i</sub>) = NonDominatedSort(P<sub>t</sub> + G<sub>t</sub>)
11. P<sub>t+1</sub>: Using rp choose from (L<sub>1</sub>, L<sub>2</sub>, ..., L<sub>i</sub>)
12. Fitness F<sub>t+1</sub> = [f<sub>1</sub>(P<sub>t+1</sub>), f<sub>2</sub>(P<sub>t+1</sub>), f<sub>3</sub>(P<sub>t+1</sub>), f<sub>4</sub>(P<sub>t+1</sub>)]
13. End for
14. Return P<sub>g</sub>

Algorithm 2 GaussianEDASample Algorithm.

Input: P<sub>t</sub>
Parameter: Population Size S
Output: G<sub>t</sub>
1. (L<sub>1</sub>, L<sub>2</sub>, ..., L<sub>i</sub>) = NonDominatedSort(P<sub>t</sub>)
2. GET L<sub>i</sub>{X<sub>1</sub>, X<sub>2</sub>, ..., X<sub>m</sub>}
3. μ = Σ<sub|i=1</sub><sup>m</sup> X<sub>i</sub>
4. Σ = 1<sub>m-1</sub> Σ<sub|i=1</sub><sup>m</sup> (X<sub>i</sub> - μ)(X<sub>i</sub> - μ)<sup>T</sup>
5. G<sub>t</sub>{ X<sub>1</sub>, X<sub>2</sub>, ..., X<sub>S</sub>} = GaussianSample(μ, Σ)
6. return G<sub>t</sub>
Experiment

This experiment is based on the scenario of a single resource supply point (rescue point) to multiple disaster sites and multiple potential disaster sites, and establishes a four-objective resource scheduling optimization that minimizes scheduling time, minimizes scheduling cost, minimizes disaster sites loss, and maximizes potential disaster sites satisfaction. The model was solved using the improved NSGA-III-GD algorithm based on Gaussian distribution estimation. We analyzed some specific emergency resource scheduling scenarios, extracted some data characteristics from them, and combined with the proposed new algorithm, so we simulated some input datas to perform the following experiment. The specific datas of this experiment are shown in the following tables:

Table 1. Resource Information.

| Resource | R1  | R2  | R3  |
|----------|-----|-----|-----|
| Total    | 500 | 600 | 400 |
| Price cost| 38  | 31  | 39  |
| Time cost| 0.2 | 0.5 | 0.3 |
| Loss     | 10  | 30  | 60  |

Table 2. Disaster Sites Information.

| Disaster Sites | N1  | N2  | N3  | N4  |
|----------------|-----|-----|-----|-----|
| Distance       | 453 | 411 | 107 | 111 |

Table 3. Potential Disaster Sites Information.

| Potential Disaster Sites | PN1 | PN2 | PN3 |
|--------------------------|-----|-----|-----|
| Probability              | 0.65| 0.23| 0.38|

Table 4. Resource requirements for disaster sites.

|       | R1  | R2  | R3  |
|-------|-----|-----|-----|
| N1    | 30  | 15  | 34  |
| N2    | 33  | 43  | 73  |
| N3    | 53  | 94  | 37  |
| N4    | 9   | 25  | 12  |

Table 5. Resource requirements for potential disaster sites.

|       | R1  | R2  | R3  |
|-------|-----|-----|-----|
| PN1   | 36  | 68  | 31  |
| PN2   | 35  | 13  | 4   |
| PN3   | 45  | 79  | 19  |

Table 6. Algorithm Parameters.

| G     | S  | pm | vmin | vmax | η  |
|-------|----|----|------|------|----|
| 700   | 100| 1/12| 50   | 80   | 20 |
Table 1 shows the storage capacity of rescue point for three resources, the unit scheduling cost of each resource, the unit scheduling time of each resource, and the unit loss of each resource missing the disaster sites.

Table 2 is the distance from rescue site to four disaster sites.

Table 3 is the probability of occurrence of three potential disaster sites.

Table 4 shows the demand for three resources at the disaster sites.

Table 5 shows the demand for three resources at potential disaster sites.

Table 6 is some hyperparameters of NSGA-III-GD algorithm, genetic generation, population size, mutation probability, minimum scheduling speed, maximum scheduling speed, and mutation parameter.

Figure 3. shows the convergence of the four optimization goals (Resource Scheduling Time, Resource Scheduling Cost, Disaster Sites Loss, Potential Disaster Sites Unsatisfaction) of the resource scheduling model. In the figure, we can see that the original NSGA-III algorithm will converge around 400-th generation, and the improved algorithm will roughly converge to the optimal solution within 200 generations. It can be seen that the improved NSGA-III-GD algorithm based on Gaussian distribution estimation has greatly improved compared with the original algorithm, and the convergence speed of the optimal solution is greatly accelerated.

**Conclusion and Future Work**

In this paper, we analyze the limitations of existing resource scheduling schemes. We propose a four-objective resource scheduling optimization model, which additionally considers potential disaster sites satisfaction, and we design an improved NSGA-III-GD algorithm based on Gaussian
distribution estimation, which makes good use of the overall characteristics of the population. Empirically, it greatly improves the search accuracy and convergence speed of the optimal solution.

There are several more promising improvements for future work. (1) Apply similar improvements to other evolutionary algorithms, such as MOEA/D; (2) Increase the number of optimization goals of the model to meet more complex actual needs; (3) Explore more sophisticated EDA methods to improve the original optimization algorithm, in order to make the search of the optimal solution be more stable.

Acknowledgement
This work is supported by the National Key Research and Development Program of China (No. 2018YFC0831501).

References
[1] Xiaofang X, Ningning L, Zejiang W, et al. Study on Deterministic Emergency Resources Dispatching Mode under Circumstances with Multiple Disaster Outbreaks[J]. Logistics Technology, 2015.
[2] Wei-Hua Z, Xiao-Lin X, Zi-Ying M, et al. Method of scheduling expressway emergency rescue resource under condition of resources sharing[J]. China Safety Science Journal, 2014.
[3] Deb K, Jain H. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints[J]. IEEE Transactions on Evolutionary Computation, 2014, 18(4):577-601.
[4] Mitsuo Gen, Lin Lin. Genetic Algorithms[M]. John Wiley & Sons, Inc. 2008.
[5] Zhang L, Lin Y, Yang G, et al. Emergency resources scheduling based on adaptively mutate genetic algorithm[J]. Computers in Human Behavior, 2011, 27(5):1493-1498.
[6] Niu Y. A Novel Model and GA-Based Solution for Resource Scheduling in Highway Emergency Rescue[C]//2018 7th International Conference on Energy, Environment and Sustainable Development (ICEESD 2018). Atlantis Press, 2018.
[7] Pablo Moscato, Carlos Cotta. A Gentle Introduction to Memetic Algorithms[M]. Springer US, 2003.
[8] Kulturel-Konak S, Smith A E, Norman B A. Multi-objective tabu search using a multinomial probability mass function[J]. European Journal of Operational Research, 2006, 169(3):918-931.
[9] Li D, Liu G, Gao Y. Uncertainty Optimization Model for Emergency Resource Scheduling[C]// Knowledge Acquisition and Modeling, 2009. KAM 09. Second International Symposium on. IEEE, 2010.
[10] Yu X B. Multi-objective Emergency supplies scheduling based on improved particle swarm optimization[J]. Industrial Engineering Journal, 2014, 17(3): 18-21.
[11] Yu P, Jia Y U, Qing-Li D A. Emergency resources scheduling on continuous consumption system based on particle swarm optimization[J]. Journal of Systems Engineering, 2007, 22(5):556-560.
[12] Yong L, Xin-Hua W, Chang-Ming X, et al. Resources Scheduling Strategy Based on Ant Colony Optimization Algorithms in Cloud Computing[J]. Computer Technology and Development, 2011.
[13] Zhou A, Zhang Q, Zhang G. Multiobjective evolutionary algorithm based on mixture gaussian models[J]. J Softw, 2014, 25(5): 913-928.
[14] Zhou Y, Liu J, Zhang Y, et al. A multi-objective evolutionary algorithm for multi-period dynamic emergency resource scheduling problems[J]. Transportation Research Part E: Logistics and Transportation Review, 2017, 99: 77-95.

[15] Zhang Q, Li H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition[J]. IEEE Transactions on Evolutionary Computation, 2008, 11(6):712-731.

[16] Chengzhong L Y U, Weimin L I. Evolutionary many objective optimization based on bidirectional decomposition[J]. Journal of Systems Engineering and Electronics, 2019, 30(2): 319-326.

[17] Jain H, Deb K. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: handling constraints and extending to an adaptive approach[J]. IEEE Transactions on Evolutionary Computation, 2013, 18(4): 602-622.

[18] Zeng G Q, Chen J, Li L M , et al. An improved multi-objective population-based extremal optimization algorithm with polynomial mutation[J]. Information Sciences, 2016, 330(C):49-73.