mathematics

Article

An Improved Intelligent Auction Mechanism for Emergency Material Delivery

Jie Zhang , Yifan Zhu, Tao Wang , Weiping Wang, Rui Wang and Xiaobo Li *

School of Systems Engineering, National University of Defense Technology, Changsha 410000, China; zhangjie@nudt.edu.cn (J.Z.); zhuyf@nudt.edu.cn (Y.Z.); wangtao1976@nudt.edu.cn (T.W.); wangwp@nudt.edu.cn (W.W.); rui_wang@nudt.edu.cn (R.W.)

* Correspondence: lixiaobo@nudt.edu.cn

Abstract: Emergency material delivery is vital to disaster emergency rescue. Herein, the framework of the emergency material delivery system (EMDS) with the unmanned aerial vehicle (UAV) as the vehicle is proposed, and the problem is modeled into a multi-trip time-dependent dynamic vehicle routing problem with split-delivery (MTTDDVRP-SD) in combination with the rescue reality, which provides decision support for planning disaster relief material. Due to the universality of dynamic interference in the process of material delivery, an optimization algorithm based on the traditional intelligent auction mechanism is proposed to avoid system performance degradation or even collapse. The algorithm adds pre-authorization and sequential auction mechanisms to the traditional auction mechanism, where the pre-authorization mechanism improves the capability performance of the system when there is no interference during the rescue process and the sequential auction mechanism improves the resilience performance of the system when it faces interferences. Finally, considering three types of interference comprehensively, which includes *new task generations*, *task unexpected changes* and *UAV's number decreases*, the proposed algorithm is compared with DTAP (DTA based on sequential single item auctions) and CBBA-PR (consensus-based bundle algorithms-partial replanning) algorithms under different dynamic interference intensity scenarios for simulation experimental from two perspectives of the capability performance and resilience performance. The results of Friedman's test with 99% confidence interval indicate that the proposed algorithm can effectively improve the capability performance and resilience performance of EMDS.

Keywords: emergency material delivery; MTVRP; DVRP; SDVRP; pre-authorization mechanism; sequential auction mechanism; resilience

MSC: 90C39

check for updates

Citation: Zhang, J.; Zhu, Y.F.; Wang, T.; Wang, W.; Wang, R.; Li, X. An Improved Intelligent Auction Mechanism for Emergency Material Delivery. *Mathematics* **2022**, *10*, 2184. https://doi.org/10.3390/math10132184

Academic Editor: Aleksandr Rakhmangulov

Received: 3 May 2022
Accepted: 21 June 2022
Published: 23 June 2022

1. Introduction

Earthquakes, floods and other large-scale natural disasters occur frequently, causing serious damage and far-reaching impacts on modern society. How to deal with natural disasters and their consequences can benefit the maintenance of social stability, and this is an emerging topic that has garnered increasing attention from both practitioners and academics in recent years. According to relevant research reports [1], the Wenchuan earthquake in 2008 caused 69,277 deaths, 374,643 injuries, 17,923 people missing, and direct economic losses of more than 800 billion US dollars and the 2015 earthquake in Nepal caused at least 8786 deaths, 22,303 injuries, and economic losses of about 113 billion yuan. The way to reduce the impact of natural disasters on human life, the economy and the environment requires not only to be prepared for detection and early warning before disasters occur [2], but also rapid post-disaster relief supplies and post-disaster recovery work, which led to the emergence of an EMDS. Due to the destruction of ground communication facilities, road traffic, and increased environmental hazards, the selection of vehicles in

EMDS presents challenges. While the use of UAVs as vehicles in EMDS can weaken the impact of ground traffic on rescue work more than manned aircraft and unmanned vehicles and avoid the loss of rescue equipment and personnel, UAVs as temporary communication relay technology are also very mature, which can replace communication base stations to restore certain communication capabilities in disaster areas [3].

The impact of the complex post-disaster environment is the urgent demand for materials by the disaster victims. Moreover, with the delay of time, the deterioration of the health of the disaster victims will further aggravate the urgency of the relief materials in the disaster area, resulting in greater casualties [4]. Song et al. [5] proposed an emergency material scheduling model with multiple disaster points and multiple rescue points, but the model is mainly aimed at the emergency material scheduling of disaster points with the same emergency indicator of material demand. Based on the weight-TOPSIS method, Guan et al. [6] determined the emergency indicator of the tasks in the disaster area, considered the penalty cost of unmet needs in the disaster area, and they established a collaborative disaster relief material scheduling model, but did not consider the change of the emergency indicator of the tasks. At the same time, due to the aftershocks after the earthquake and the crash of the UAVs, some other situations have brought great dynamics to the post-disaster rescue. On the other hand, considering the loading capacity constraints of UAVs and battery life constraints, UAVs are often required to be reused, and the material needs of a single disaster area may not be satisfied by a single UAV at one time. These are all issues that EMDS should consider when completing disaster relief, so we modeled it as a multi-trip time-dependent dynamic vehicle routing problem with split-delivery (MTTDDVRP-SD), based on the classic multi-trip vehicle routing problem (MTVRP) [7], which also takes into account the following characteristics: multi-trip per UAV, time-dependent emergency indicator, dynamics of the environment, split-delivery, and trip duration limit.

MTTDDVRP-SD takes into account various practical situations in the delivery of emergency materials, which makes it unique and difficult. From the perspective of models, MTTDDVRP-SD can be seen as a combination of MTTDVRP, DVRP and SDVRP, although these three models are more in line with the real problem, but the relevant research is not sufficient, especially the combination of the three is less.

- In the delivery of emergency materials, it is common to allow vehicles to make multiple trips, and it is also widely used in logistics [8], public transportation scheduling [9] and other fields. Cattaruzza et al. [10] emphasized that allowing vehicles to make multiple trips can significantly improve vehicle utilization and reduce the number of vehicles required, thereby further reducing fixed costs and overall costs, and the application in life is more realistic. Moreover, they theoretically proved that the MTVRP problem was NP-hard. Dorling et al. [11] argued that existing VRPs are insufficient for planning UAV deliveries, that is because multiple trips to the depot are not permitted, resulting in too many UAV solutions, so they proposed two multi-trip VRPs for UAV deliveries to address both problems. Sun et al. [12] proposed a time-dependent multi-trip vehicle routing problem (TDMTVRP) with an improved travel speed model, arguing that speed is time-dependent. Moreover, the experimental results indicated that compared with the capacitated vehicle routing problem with the time windows model (CVRPTW), the TDMTVRP model proposed can both decrease the vehicle utilized times dramatically and shorten the vehicle travel distances slightly in dealing with the vehicle routing problem (VRP).
- Dynamic VRP (DVRP) [13] emerged as a new variant of VRP and attracted more and more attention. Li et al. [14] investigated the emergency material dispatching problem under uncertainty and built a multi-objective optimization model to minimize the total cost and maximize the affected rate of demand, but without considering the case of multiple trips of vehicles. Nan et al. [15] investigated the dynamic demand VRP with time windows for the model of mixed distribution of electric and fuel logistics vehicles, i.e., the demand of the task changes randomly or a new task is

generated randomly during the execution of the task. Sabar et al. [16] considered the DVRP problem under vehicle speed uncertainty caused by traffic congestion, but this uncertainty has regularity. Ouaddi et al. [17] proposed the MTDVRP problem with a combination of MTVRP and DVRP and recognized the complexity of such problems. Their work considered the problem of vehicle overloadable transportation and distribution, with dynamics reflected in the cancellation and modification of user orders and changes in vehicle speed due to congestion, but without considering the problem of delivery splitting at a task.

- The split-delivery vehicle routing problem (SDVRP) is a variant of the Capacitated Vehicle Routing Problem (CVRP) in which a customer can be served by more than one vehicle, i.e., the requirement of a customer can be split among vehicles. Casazza et al. [18] addressed a single commodity Pickup and Delivery Vehicle Routing Problem and assumed that the requirements of both pickup and delivery nodes can be split, and each node can be visited more than once. An extensive experimental analysis showed their method to offer simultaneously more modeling flexibility and more computational effectiveness. Hartomo et al. [19] investigated the problem of distribution of relief materials from a logistics center to a network of demand points, and their assumptions included that the requirement for materials at a node would be served by several trucks or over multiple trips. However, their work does not take into account the dynamic changes in the environment and emergency indicator over time. Zhang et al. [20] considered the case of a customer requiring multiple vehicles for service, but without considering the change in capability due to the change in external information during the working process, and the case of multiple trips.

Currently, to solve these kinds of problems, the exact algorithm [21] and memetic algorithm [17] are commonly used, but they are only suitable for solving small-scale arithmetic cases. On the other hand, most of the processing methods of DVRP are dividing time slices and decomposing into a series of static VRP problems [22,23], and then using heuristic algorithms to solve them, but under the conditions of multi-trip and delivery splittable, the solution search space is huge and the time complexity is very high, so much so that it is not suitable for emergency material delivery scenarios under environmental uncertainty. For this purpose, this paper considers an intelligent auction algorithm based on the market mechanism, such as Guo et al. [24] who proposed an auction algorithm that introduces the idea of "blockchain" to realize the task allocation problem in the "last mile" logistics distribution. Auction algorithms have been improved in the literature, Choi et al. [25] proposed a consensus-based auction algorithm (CBBA), which uses both auction and consensus mechanisms for task selection and conflict resolution, thus reducing the computational resource consumption of the algorithm and accelerating the convergence. The performance improvement of CBBA algorithm over the initial auction algorithm was experimentally verified, but the research does not address the dynamic change of task information. In a later study, Buckman et al. [26] proposed the CBBA-PR algorithm for partial replanning based on the CBBA algorithm in order to cope with dynamic changes in the environment by mixing a subset of unfinished tasks with new tasks to achieve fast reallocation. CBBA-PR ensures optimality in dynamic assignment and improves the fast response of the algorithm to dynamic situations, but the algorithm is not effective in some cases where real-time performance is required or the problem size is large. Chen et al. [27] argued for the better applicability of online algorithms for the uncertain arrival of dynamic tasks and proposed that tasks should be assigned one by one rather than in batches to ensure the efficiency of UAV execution. Farinelli et al. [28] converted multi-robot patrol into a task assignment problem, considered online decision making techniques, proposed the DTAP algorithm based on sequential single auction, and verified the performance of the algorithm on a real robot platform. However, they relied only on the task information at the current moment, without considering more global known information. Of course, centralized algorithms such as the genetic algorithm (GA) and particle swarm optimization (PSO) can also solve VRP, but they are less adaptable to the dynamic situation of changing environment in the disaster area. For example, Luo et al. [29] pointed out that when the global information of the task changes dynamically over time, the centralized algorithm tends to decompose and re-iterate the constraint solution, which is less flexible than the auction algorithm.

When EMDS is implemented to start the emergency material delivery task, it is still necessary to pay attention to two aspects, i.e., not only to ensure the capability performance to avoid secondary damage to the disaster area due to untimely rescue, but also to ensure the resilience performance to have the ability to continue the task. In fact, accurate and timely information about the disaster area can be obtained through efficient reconnaissance detection [30]. However, the uncertainty of the environment in a disaster is common and may cause the system to collapse, thus requiring the EMDS to adapt effectively to interferences, i.e., to have a good resilience performance. While resilience was first introduced to engineering systems in 1995 [31], defined as, "Resilience is the probability of a system's recovery from failures", in 2009, Woods [32] argued that the focus should be on the adaptability of a system facing more complex situations, thus giving the definition, "Resilience, as a form of adaptive capacity, is a system's potential for adaptive action in the future when information varies, conditions change, or new kinds of events occur, any of which challenge the viability of previous adaptations, models, or assumptions". As to how to quantitatively evaluate resilience, summarizing the research results over the years, it can be broadly divided into two categories: time-dependent and time-independent models.

The core of time-dependent models is that the performance of systems (e.g., power transmission systems [33] and communication systems [34]) is time-dependent and can be expressed quantitatively, and such models usually describe the resilience of the system in terms of performance change curves. In contrast, time-independent models are mostly used to describe the system capability performance only calculated after the completion of the task, such as air defense systems [35]. The integral resilience model is commonly used for quantitative description of resilience, which describes the system resilience by comparing the cumulative distributional impact of system performance over time with and without interference, and can be found in the literature [36]. Dessavre et al. [37] studied the resilience of offensive and defensive countermeasures based on the integral resilience metric, using the number of OODA cycles as a performance indicator, but without considering the multi-interference situation, so some improvements are needed in emergency material delivery.

Motivated both by the adaption of new technologies and delivering vehicle (UAV) in practical transportation industry and the theoretical gap existing in the current literature, we investigate a problem model (MTTDDVRP-SD) that is more adapted to the emergency material delivery scenario in disaster areas. On this basis, due to the existence of two layers of optimization of UAV routing and delivery quantity, which leads to a huge solution search space of the traditional evolutionary algorithm and does not adapt to the objective demand of the dynamics of the disaster area. So, we try to design a new algorithm based on the intelligent auction mechanism and propose a resilience evaluation indicator under dynamic multiple interferences. Specifically, the main contributions of this paper can be summarized as follows.

- To meet the performance requirements of emergency material delivery, an overall framework of EMDS with UAV as the vehicle is proposed, and the emergency material delivery problem is described as a new variant of VRP (MTTDDVRP-SD);
- Based on the traditional intelligent auction algorithm, pre-authorization and sequential auction mechanisms are added, and sequential auction algorithm based on pre-authorization (SAABoPA) is proposed to solve MTTDDVRP-SD to optimize the system performance.
- Based on the integral resilience model, a multi-interference resilience evaluation indicator suitable for emergency material delivery scenarios is proposed;
- The experiments are carried out under different dynamic interference intensities through the post-earthquake emergency material delivery scenario and compared

with the comparative algorithms from two perspectives of capability performance and resilience performance to verify the effectiveness of the algorithm in this paper.

The problem MTTDDVRP-SD will be deeply analyzed and defined in the next section, while in Section 3, the proposed method of SAABoPA for solving it is presented. The experimental results are reported in Section 4. Section 5 concludes this study and discusses future research directions.

2. Problem Definition and Formulation

2.1. Problem Definition

UAVs with loading capacity were considered to deliver relief materials to the earthquake-affected areas, and information about the danger in the affected areas was obtained by satellite remote sensing technology, UAV reconnaissance technology, and ground rescue teams [38]. The relief materials are concentrated in the distribution center at the border of the disaster area, and the UAV makes task assignment according to the disaster area danger information, determines the task target, loads the materials at the distribution center, and then carries out the delivery to the disaster area target. The specific EMDS framework is shown in Figure 1, which describes the interaction between the system and the outside. The EMDS includes *information processing subsystem (IPSys)*, *command and control subsystem (CCSys)*, and *loading and transportation subsystem (LTSys)*.

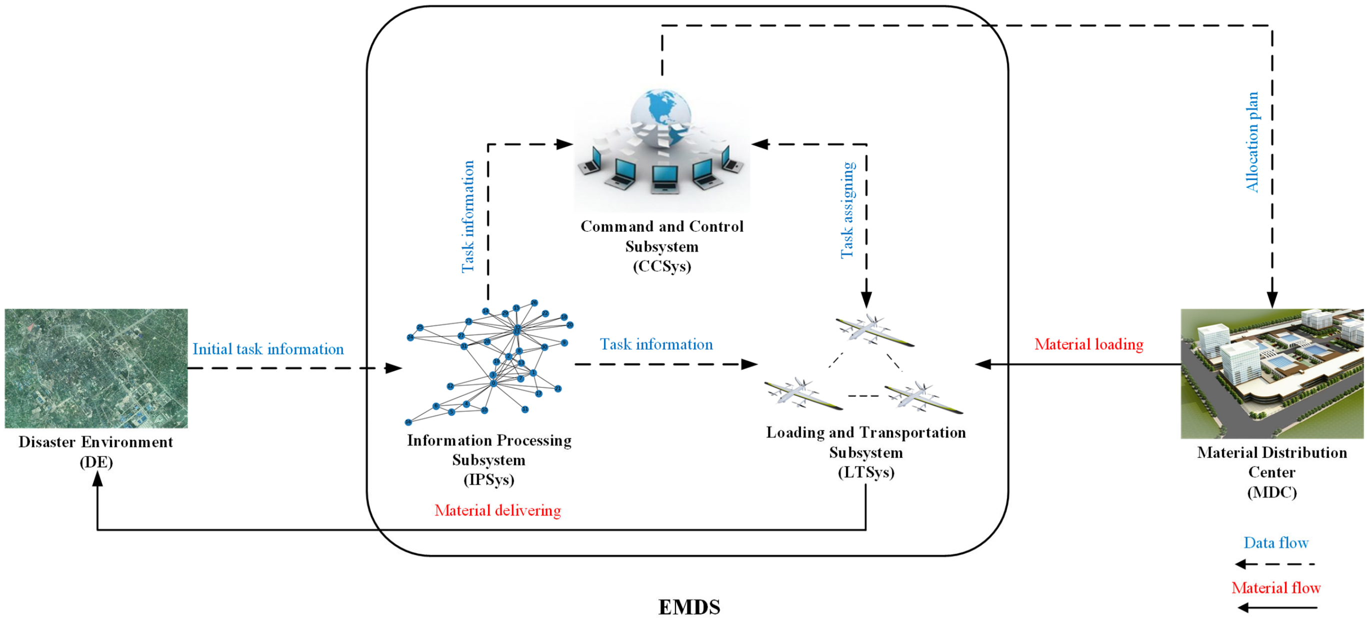


Figure 1. EMDS interaction framework chart.

The subsystem functions can be described as follows:

1. *IPSys*: Responsible for filtering, standardizing and sharing task information. Task information provided by the *disaster environment (DE)* is filtered, standardized, and shared with *CCSys* and *LTSys*. The sharing of task information is discontinuous and will only be re-shared when there is a sudden change in the information.
2. *CCSys*: Responsible for task assignment and solution sharing. Based on the task information shared by *IPSys*, the task assignment is executed by combining the status and capability of each *LTSys* component. Due to the dynamic change of task information, task assignment is not executed at once. After the allocation plan is formed, it is shared to the *material distribution center (MDC)*.

3. *LTSys*: Responsible for task assignment and material delivery. Based on the task information shared by *IPSys*, the task assignment is completed with the unified scheduling of *CCSys*. After the assignment plan is determined, the UAVs in *LTSys* load the materials stored in the *MDC* and execute the material delivery process.

The relevant assumptions of this paper are as follows:

1. The material requirements of the task may not be met by one UAV at a time;
2. A single deployment of a UAV may execute multiple tasks;
3. When the UAV is empty, it needs to be returned to the *MDC* and reloaded to be able to execute the next task;
4. Obstacle avoidance among UAVs and against obstacles (e.g., mountains and forests) is not considered. Obstacle avoidance can be accomplished by adjusting the height of the UAV under certain conditions [39], and the increase in range due to height change can be ignored because the height difference is small relative to the range;
5. In the case of emergency rescue, the UAV always maintains the maximum capacity, i.e., the UAV always maintains the maximum power during the task. The UAV endurance can be considered as fixed for a certain battery capacity;
6. Ignoring the deceleration of the UAV during the delivery of the material, as well as the change in direction, the time variation from this is insignificant in comparison. Theoretically, this variation could be integrated into the velocity and loading relationship, but we do not consider it in order to simplify the problem. Moreover, the kinematic model of the UAV is not investigated in depth in this paper;
7. Consider using a single base, i.e., an *MDC*. For a general multi-base VRP, it can be divided into sub-blocks by clustering methods, and within the sub-blocks still remains a single base VRP;
8. The material in the MDC meets the material needs of all affected areas and only one type of material is considered. The current national reserve of emergency resources is sufficient, and the amount of emergency resources available can be considered adequate in the face of a catastrophic event in a local area. In fact, many types of materials are needed in disaster relief situations, including food, water, medicine, etc. However, in order to facilitate the emergency and effective implementation of relief, in many cases we do not count the requirements of various types of materials too much and make a precise delivery, and the more general practice is to synthesize various materials into a rescue kit [40].

2.2. Problem Formulation

In this section, the mathematical model proposed to solve the above problem is introduced. However, before that we need a formal description of the disaster area environment and dynamic interferences.

2.2.1. *DE* Formulation

Here, we first formalize spatio-temporal characteristics and dynamic properties of *DE* based on task information of the disaster area.

Definition 1. *Task environment G. The task environment is modeled as an undirected graph $G = (V, E)$, where V denotes the set of vertices in Euclidean space, E denotes the set of edges.*

In G, V includes the task node T and the *MDC* node V_B. The edge $E \subset V \times V$ indicates that two neighboring vertices are reachable between them and are paths for the UAV to move. A task node represents a delivery task, and the set of all tasks is $T = \{T_1, T_2, \ldots, T_j, \ldots, T_{|T|}\}$.

Definition 2. *Time t. Time is represented by a set of discrete sequences, $t \in \{0, 1, 2, \ldots\}$.*

Take the time of the first UAV taking off for the task as the base time 0.

Definition 3. *Coordinates* (x, y)*. The node position is represented by the two-dimensional coordinates* (x, y)*.*

The node coordinates considered in this paper include the task coordinates and *MDC* coordinates, and the task coordinates are used to characterize the nearby affected areas that need relief materials.

Definition 4. *Quantity required* $R(t)$*. At t moments, the quantity of material required by the task* $T_j \in T$ *is denoted by* $R_j(t)$*.*

To simplify the problem, the quantity required is unitized, e.g., $R_j(t) = 3$ means that the quantity of material required for the task T_j at time t is 3 units.

Definition 5. *Emergency indicator* $E(t)$*. At t moments, the emergency of the task* $T_j \in T$ *on the material is denoted by* $E_j(t)$*.*

The emergency indicator [6] indicates the urgency of the requirement for materials at the task, which varies with time and the delivery of materials. When the upper threshold is reached, it will cause secondary damage to the disaster area due to the lack of replenishment of materials for a long time [41]. This is similar to a time window constraint, but the difference is that the emergency indicator can describe the range of time constraints, the value of the task is varying, i.e., the sooner the task is completed the higher the value. The formula is expressed as follows.

$$E_j(t) = \begin{cases} 0, & R_j(t) \leq 0, \\ 1, & E_j(t) \geq 1, \\ E_j(t_0) + a \cdot (t - t_0) - \frac{E_j(t_0)}{R_j(t_0)} \cdot MD_{ij}(t), & \text{others}. \end{cases} \tag{1}$$

where t_0 denotes the moment of the last *IPSys* shared information. Equation (1) indicates that the emergency indicator $E_j(t)$ of a task gradually increases with time t, but the material replenishment decreases the value of $E_j(t)$. In fact, the task emergency indicator formula is a fit to the actual indicator, and *IPSys* shares the information every time the task information changes, i.e., the task emergency indicator can be corrected at the t_0 moment.

2.2.2. Dynamic Interference Formulation

The dynamic scenarios in this paper consider three possible types of interferences in which their occurrence is unpredictable, and are therefore described in terms of stochastic processes and defined as follows.

Definition 6. *Dynamic interference type 1: new task generations. This means that during the delivery of emergency materials, new tasks will emerge as the disaster continues.*

Taking the stochastic process $\left\{\widetilde{\mathbf{X}}(n^1), n^1 \in \{1, 2, \ldots, \left|DIS^1\right|\}\right\}$ denotes the occurrence of dynamic interference events of the type *new task generations*, where $\widetilde{\mathbf{X}}(n^1) = (\widetilde{t}(n^1), \widetilde{x}(n^1),$ $\widetilde{y}(n^1), \widetilde{R}(n^1), \widetilde{E}(n^1))$, and each random variable $\widetilde{t}(n^1), \widetilde{x}(n^1), \widetilde{y}(n^1), \widetilde{R}(n^1), \widetilde{E}(n^1)$ satisfies a certain probability distribution, respectively, $\widetilde{t}(n^1) \sim DiscreteU(0, T_{end})$, $\widetilde{x}(n^1) \sim U(\underline{x}, \overline{x})$, $\widetilde{y}(n^1) \sim U(\underline{y}, \overline{y})$, $\widetilde{R}(n^1) \sim DiscreteU(\underline{R}, \overline{R})$, $\widetilde{E}(n^1) \sim U(\underline{E}, \overline{E})$. The result of the occurrence of dynamic interference events of this type causes the task T to be expanded, and the information of the new task is all generated randomly in the random vector $\widetilde{\mathbf{X}}(n^1)$, which is represented by the following equation.

$$T = T \cup \{T_{|T|+n^1}\}, \ \ t = \widetilde{t}(n^1) \tag{2}$$

Definition 7. *Dynamic interference type 2: task unexpected changes. This means that during the delivery of emergency materials, information on the material requirements and emergency indicator of currently known tasks will deteriorate as the disaster continues and disaster information is reevaluated.*

Taking the stochastic process $\left\{\widetilde{\mathbf{Y}}(n^2), n^2 \in \{1, 2, \ldots, |DIS^2|\}\right\}$ denotes the occurrence of dynamic interference events of the type *task unexpected changes*, where $\widetilde{\mathbf{Y}}(n^2) = (\widetilde{t}(n^2), \widetilde{j}(n^2), \widetilde{\Delta R}(n^2), \widetilde{\Delta E}(n^2))$, and each random variable $\widetilde{t}(n^2), \widetilde{j}(n^2), \widetilde{\Delta R}(n^2)$, $\widetilde{\Delta E}(n^2)$ satisfies a certain probability distribution, respectively, $\widetilde{t}(n^2) \sim DiscreteU(0, T_{end})$, $\widetilde{j}(n^2) \sim DiscreteU(1, |T|)$, $\widetilde{\Delta R}(n^2) \sim DiscreteU(0, \overline{R}/2)$, $\widetilde{\Delta E}(n^2) \sim U(0, \overline{E}/2)$. The result of the occurrence of dynamic interference events of this type causes changes in the information about the material requirements and emergency indicator in the current task T, and the amount of changes in the information are generated randomly in the random vector $\widetilde{\mathbf{Y}}(n^2)$, which is represented by the following equation.

$$R_{\widetilde{j}(n^2)}(t) = R_{\widetilde{j}(n^2)}(t) + \widetilde{R}(n^2), \ \ t = \widetilde{t}(n^2) \tag{3}$$

$$E_{\widetilde{j}(n^2)}(t) = E_{\widetilde{j}(n^2)}(t) + \widetilde{E}(n^2), \ \ t = \widetilde{t}(n^2) \tag{4}$$

Definition 8. *Dynamic interference type 3: UAV's number decreases. This means that during the delivery of emergency materials, the number of available UAVs will be reduced due to failure and other circumstances.*

Taking the stochastic process $\left\{\widetilde{\mathbf{Z}}(n^3), n^3 \in \{1, ..., |DIS^3|\}\right\}$ denotes the occurrence of dynamic interference events of the type *UAV's number decreases*, where $\widetilde{\mathbf{Z}}(n^3) = (\widetilde{t}(n^3), \widetilde{i}(n^3))$, and each random variable $\widetilde{t}(n^1) \sim DiscreteU(0, T_{end})$, $\widetilde{i}(n^3) \sim DiscreteU(1, |U|)$. The result of the occurrence of dynamic interference events of this type causes the number of currently available drones is reduced, which is represented by the following equation.

$$U = U - \{U_{\widetilde{i}(n^3)}\}, \ \ t = \widetilde{t}(n^3) \tag{5}$$

2.2.3. MTTDDVRP-SD

In summary, The MTTDDVRP-SD is formulated as follows.

$$f = \min \textstyle\sum_1^{N_T} E_j(T_{end}) \tag{6}$$

$$0 \leq L_i^k(t) < L_i^{\max} \tag{7}$$

$$L_i^k(t) = L_i^k(sT_i^k) - \textstyle\sum_{j=1}^{|T|} x_{ij}^k \times MD_{ij}(t) \tag{8}$$

$$L_i^k(sT_i^k) > 0 \tag{9}$$

$$0 \leq MD_{ij}(t) \leq L_i^k(t) \tag{10}$$

$$aT_i^k - sT_i^k \leq T_i^{duration} \tag{11}$$

$$v_i(t) = v_i^{unload} - \delta_i \times L_i^k(t) \tag{12}$$

$$1 \leq Ex_i \tag{13}$$

$$\sum_{k=1}^{Ex_i} x_{ij}^k \in \{0,1\} \quad (14)$$

$$\sum_{j=1}^{|T|} x_{ij}^k \geq 1 \quad (15)$$

$$\sum_{i=1}^{|U|} \sum_{k=1}^{Ex_i} x_{ij}^k \geq 1 \quad (16)$$

$$0 \leq MD_{ij}(\mathrm{t}) \leq R_j(t) \quad (17)$$

$$x_{ij}^k \in \{0,1\} \quad (18)$$

Equation (6) aims to minimize the number of tasks whose emergency indicator reaches the threshold 1 at the end of the emergency material delivery. Different from the general objective function of the VRP and its variants problem, which considers minimizing transportation and inventory costs or maximizing transportation and inventory rewards [42,43]. In the disaster area emergency material delivery problem, this paper considers, based on the rescue work after the earthquake disaster, to ensure the safety of victims' lives and avoid more secondary injuries. Equation (7) indicates that the loading of any UAV at any time is less than the maximum loading and is non-negative. Equation (8) represents the change in the current UAV loading. Equation (9) represents the loading capacity of the UAV U_i when it is deployed from the *MDC* is strictly large 0. Equation (10) indicates that the quantity delivered by the UAV U_i to the task T_j is less than the current loading capacity of the UAV i. Equation (11) represents the flight time constraint for a single UAV trip. Equation (12) represents the relationship between the UAV speed and the loading capacity [44]. Equation (13) indicates the constraint on the number of trips of a single UAV. Equation (14) indicates the entire emergency material delivery process, the UAV U_i to the task T_j at most once. Equation (15) indicates that a single UAV executes at least one task in a single trip. Equation (16) indicates that the entire emergency material delivery process, task j is executed at least once. Equation (17) indicates that the quantity of UAV U_i to task T_j delivery is less than the current requirement of task T_j. Equation (18) represents the decision variables.

3. Method: SAABoPA

3.1. Overall Framework

In the EMDS framework, *CCSys* and *LTSys* use the SAABoPA algorithm for task assignment based on the task information shared by *IPSys*. The tasks shared by *IPSys* are equivalent to *auction items*. *CCSys* acts as the *auctioneer* and sends the task information to the UAVs in the corresponding *LTSys*. The UAV acts as a *bidder*, calculates the net benefit for each task with the task price stated by *CCSys*, and sends the bid price of the task back to *CCSys*. *CCSys* decides the pre-authorization and authorization of the task based on the bid price, and sends the task to the authorized UAV and the overall framework of SAABoPA is shown in Figure 2.

The SAABoPA algorithm requires the UAV to obtain at most one task authorization and one task pre-authorization in one auction. The algorithm is based on pre-authorization to ensure the system capability performance, and based on sequential auctions to effectively improve the system resilience performance and avoid the frequent occurrence of interference that causes the system performance degradation by reallocation or adjustment of the assignment results. It should be noted that since the published task is executed in the future time and has no effect on the current execution process, the UAV can participate in the auction during the execution of the task [27], but it can only obtain the pre-authorization of the task instead of the authorization.

There are two possibilities for starting an auction:

1. *CCSys* active start: in case of interference.
   - In EMDS, when the task information changes abruptly (due to new task T_{new} generation, unexpected changes in task information and completion of one delivery), *IPSys* shares the current task information with *CCSys*. Based on the shared information and the set of previous authorization information Y, *CCSys* can determine the cause of the task information mutation. If a task is completed, the auction is not started; if a new task T_{new} is created and the task information changes unexpectedly, the auction is started and the previous set of preauthorization information X is deleted and the task price vector $\mathbf{P}(t)$ is updated;
   - In the case that the UAV U_i is lost, the UAV U_i task completion time is known from the elements in the previous set of authorization information from *CCSys* (expected arrival time t_i^1). If *IPSys* shares the information with *CCSys* at this point and it matches the expected change, the task is successfully executed and the auction is not started; otherwise, it means that the UAV is malfunctioning and there is one less bidder in the next auction, so ask *IPSys* to share the task information at this moment, start a new auction, and delete the previous set of pre-authorization information X and reset the task price vector $\mathbf{P}(t)$.
2. CCSys passive start: in case of no interference.
   After the UAV U_i arrives at the *MDC* V_0, it sends an auction asking to *CCSys*. If U_i has been pre-authorized, it is directly converted to authorization and sent to U_i. Meanwhile, what needs to be achieved is to add the pre-authorization quintuple relation related to U_i to the authorization set Y and remove it from the pre-authorization set X; conversely, start the auction and *CCSys* sends the task information set *TASK* to all UAVs that are not pre-authorized.

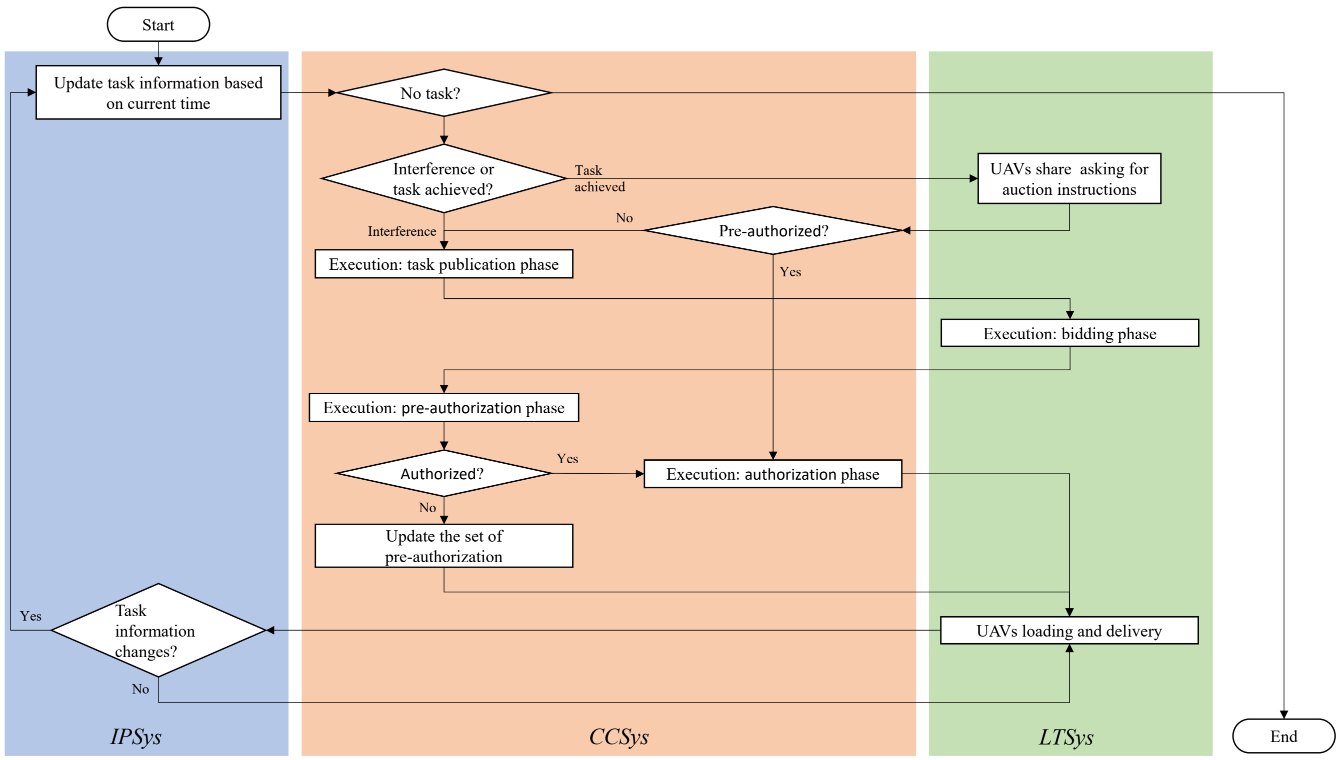


Figure 2. Overall framework of SAABoPA.

3.2. Mechanism Design

In MTTDDVRP-SD, after a task is executed once, if it is not completely completed, it is considered as a new task and the gain obtained from the execution is recalculated. In a dynamic environment, interference *new tasks generations* and *task unexpected changes* occur without burden to the proposed algorithm. This is because newly generated tasks or tasks with unintended changes in information are treated the same as tasks after one execution. Moreover, in the auction mechanism, UAVs lose their capabilities due to failures, which is equivalent to a reduction in the number of *auctioneers*. Due to the loose coupling between UAVs, no additional burden is imposed on the algorithm of this paper.

The SAABoPA mechanism consists of four phases: (1) Task Publication Phase, (2) Bidding Phase, (3) Pre-authorization Phase and (4) Authorization Phase. The *CCSys* performs the task publication, pre-authorization and authorization phases, and the UAV in the *LTSys* performs the bidding phase.

3.2.1. Task Publication Phase

At the beginning of an auction, *CCSys* releases to *LTSys* the estimated set of task information $TASK = \{T, (x, y), R', E', \mathbf{P}(\tau)\}$, where T denotes the set of uncompleted tasks at the current τ moment. When task T_j is partially completed, T_j will not be deleted from T, but the information of material requirements and emergency indicator will be updated; when task T_j is fully completed, T_j will be deleted, $T = T - \{T_j\}$; when a new task T_{new} appears, the set of tasks T is expanded, $T = T \bigcup \{T_{new}\}$. N_T denotes the number of elements in the set T. $(x, y) = \{(x_j, y_j)|T_j \in T\}$ denotes the position information of the uncompleted tasks. $R' = \{R_j^{'}(\tau)|T_j \in T\}$ denotes the estimated material requirement of the uncompleted tasks. $E' = \{E_j^{'}(\tau)|T_j \in T\}$ denotes the estimated value of the emergency indicator of the uncompleted tasks. $\mathbf{P}(t)$ is the price vector consisting of the prices of all current outstanding tasks, and the initial moment price vector $\mathbf{P}(0) = 0$.

In the r-th auction, there are some tasks that have been authorized but not executed in the $r-1$-th auction. In order to accurately calculate the rewards of the UAV for the tasks in the r-th auction, we introduce the material requirement information prediction $R_j^{'}(\tau)$ and the emergency indicator information prediction $E_j^{'}(\tau)$. The $R_j^{'}(\tau)$ and $E_j^{'}(\tau)$ of the task T_j are calculated based on the $R_j(\tau)$ and $E_j(\tau)$ true values and the authorized UAV U_{i_j} in the $r-1$ auction delivery $MD_{i_j j}(\tau)$, expressed as follows.

$$R_j^{'}(\tau) = R_j(\tau) - MD_{i_j j}(\tau) \tag{19}$$

$$E_j^{'}(\tau) = \begin{cases} E_j(\tau) - \frac{E_j(\tau)}{R_j(\tau)} \cdot MD_{i_j j}(\tau), & R_j^{'}(\tau) \neq 0, \\ 0, & \text{others.} \end{cases} \tag{20}$$

3.2.2. Bidding Phase

Different loadings of UAVs lead to different speeds, so that UAVs may obtain different rewards for the same task. Accordingly, while completing the task assignment in EMDS, the loading and delivery of UAVs can be determined. The pseudo-code is shown in Algorithm 1.

Before bidding, the UAVs in *LTSys* calculate the income values based on their own computational resources based on the set of task information shared by *CCSys*, and form the income matrix:

$$INCOME_{i,T}(t) = \begin{bmatrix} Income_{i,1}^{1}(t) & \cdots & Income_{i,1}^{L_i}(t) \\ \vdots & \ddots & \vdots \\ Income_{i,|T|}^{1}(t) & \cdots & Income_{i,|T|}^{L_i}(t) \end{bmatrix}_{|T| \times L_i} \tag{21}$$

where, $INCOME_{i,T}(t)$ denotes the income matrix of the UAV U_i for the current set of outstanding tasks T at t moments. The matrix elements are calculated as:

$$Income_{ij}^{L_i}(t) = Value_{ij}^{L_i}(t) - Cost_{ij}^{L_i}(t) \tag{22}$$

$$Value_{ij}^{L_i}(t) = \begin{cases} E_j(t_0) + a \cdot (\frac{D_{ij}}{v_i(t)} + t - t_0) + \frac{E_j(t_0)}{R_j(t_0)} \cdot \max(MD_{ij}, L_i), & E_j(t) + \frac{a \cdot D_{ij}}{v_i(t)} < 1, \\ 0, & \text{others}. \end{cases} \tag{23}$$

$$Cost_{ij}^{L_i}(t) = \begin{cases} \frac{s \cdot (t_i^1 - t_i^0 + -\frac{d_{j,V_0}}{v_i(t_i^1)})}{T_i^{duration} - T_i^{fly}}, & t_i^1 - t_i^0 + -\frac{d_{j,V_0}}{v_i(t_i^1)} + T_i^{fly} \le T_i^{duration}, \\ \infty, & \text{others}. \end{cases} \tag{24}$$

where, $Income_{ij}^{L_i}(t)$ denotes the reward value that a UAV U_i with loading L_i of materials can obtain by executing a task T_j at time t, which is determined by the value of the task $Value_{ij}^{L_i}(t)$ and cost $Cost_{ij}^{L_i}(t)$; s is the scaling factor; t_i^0 denotes the expected time to start the task T_i; t_i^1 denotes the expected time to complete the task T_j, satisfying $D_{ij} = \int_t^{t_i^1} v_i(\tau) d\tau$; T_i^{fly} denotes the time that the UAV U_i has flown; MD_{ij} denotes the time that the UAV U_i has performed the task T_j of material delivery, satisfying $MD_{ij} = \min\{L_i, R_j(t)\}$; D_{ij} denotes the actual flight distance of the UAV U_i from the current position to the task T_j.

Algorithm 1: Bidding phase (Part of *LTSys*, take U_i as an example).

Input : set of task information $TASK$;
Output: Set of bidding BID_i.

```
calculate INCOME_{i,T}(t);
calculate NET_{i,T}(t);
if all(NET_{i,T}(t)(:) < 0) then
    P'_i(t) ← (−∞, −∞, ..., −∞)^T;
    BID_i ← {U_i, ∅, P'_i, 0, 0, 0};
    NODE_i ← NODE_i ∪ {V_0} and go to V_0;
else
    net^max_{i,T}(t) ← max NET_{i,T}(t);
    (T_{j^i}, L_i) ← index(net^max_{i,T}(t));
    MD_{ij^i} ← min{L_i, R_{j^i}(t)};
    t^1_i ← f(D_{ij^i}, t, v_i(τ));
    net^max_{i,T−{T_{j^i}}}(t) ← max NET_{i,T−{T_{j^i}}}(t);
    calculate P'_i(t);
send BID_i ← {U_i, T_{j^i}, P'_i, L_i, MD_{ij}, t^1_i} to CCSys;
```

In Equation (23), under the constraint of $\frac{E_j(t)}{a} + \frac{D_{ij}}{v_i(t)} < 1$, the first two terms of the right-hand side of the $Value_{ij}^{L_i}(t)$ calculation equation indicate the emergency indicator at the moment before the task is executed, which reflects the basic value of the UAV to execute the task; the third term of the equation reflecting the additional value of the UAV U_i executing task T_j, i.e., the UAV loads as much materials as possible under the condition that the material is obtained before the emergency indicator of the task T_j reaches 1.

The UAV U_i based on the price vector $\mathbf{P}(t)$ of the uncompleted tasks, combined with the income matrix can be calculated to obtain the net income matrix $NET_{i,T}(t)$ as:

$$NET_{i,T}(t) = INCOME_{i,T}(t) - [\boldsymbol{P}_T(t); \boldsymbol{P}_T(t); ...; \boldsymbol{P}_T(t)]_{N_T \times L_i} \tag{25}$$

The maximum net income $net_{i,T}^{\max}(t)$ is:

$$net_{i,T}^{\max}(t) = \max NET_{i,T}(t) \tag{26}$$

Then, the tasks T_{j^i} bid by the UAV U_i, the material load L_i and the delivery MD_{ij^i} to the task T_{j^i} are:

$$(m, n) = index(net_{i,T}^{\max}(t)) \tag{27}$$

$$T_{j^i} = T(m) \tag{28}$$

$$L_i = n \tag{29}$$

$$MD_{ij^i} = \min\{L_i, R_{j^i}(t)\} \tag{30}$$

where the function $(m, n) = index(a)$ in Equation (27) means to obtain the values of the ranks of a in the matrix A, and if there are multiple values of a in the matrix A, then a randomly selected rank value.

To further determine the bidding of the UAV U_i to the task T_{j^i}, the maximum net income value $net_{i,T-\{T_{j^i}\}}^{\max}(t)$ in the net income matrix of the UAV U_i other than the task T_{j^i} is also needed, expressed as follows:

$$net_{i,T-\{T_{j^i}\}}^{\max}(t) = \max NET_{i,T-\{T_{j^i}\}}(t) \tag{31}$$

where $T - \{T_{j^i}\}$ denotes the deletion of elements of the set T_{j^i} from the set of tasks T.

Combining Equations (21)~(31), the bidding vector $\mathbf{P_i}'(t)$ of the UAV U_i for all tasks is determined as follows.

$$\boldsymbol{P}_i'(t) = (p_{i,1}'(t), p_{i,2}'(t), ..., p_{ij}'(t), ..., p_{i,N_T}'(t))^T \tag{32}$$

$$p_{ij}'(t) = \begin{cases} Income_{i,j^i}^{L_i}(t) - net_{i,T-\{T_{j^i}\}}^{\max}(t) + \varepsilon, & j = j^i, \\ -\infty, & j \neq j^i. \end{cases} \tag{33}$$

where $\varepsilon > 0$ is a slack parameter [42] that ensures that the algorithm does not enter a dead loop when multiple UAVs compete for the same task because the bids do not improve; $P_{ij}'(t)$ denotes the bidding of the UAV U_i for the task T_j at time t.

If the elements of the net income matrix $NET_{i,T}(t)$ are all non-positive, which means that the UAV U_i will not profit from bidding on any task at the current price, then the UAV U_i abandons further bidding in this auction and sets $\mathbf{P}_i'(t) = (-\infty, -\infty, \ldots, -\infty)^T$, as shown in lines 3~6 of Algorithm 1; lines 7~13 show that if there are positive values of the elements in the net income matrix, the UAV U_i participates in the bid, and the task, loading and bid price of the bid are determined by Equations (26)~(33).

After the UAV U_i in *LTSys* completes the bidding phase, it sends the set of bid information $BID_i = \{U_i, T_{j^i}, \mathbf{P}_i', L_i, MD_{ij}, t_i^1\}$. The elements in the set BID_i denote the UAV order number, the bid task, the bid vector to the task, the material loading, the delivery to the task, and the expected arrival time, respectively.

3.2.3. Pre-Authorization Phase

The *CCSys* receives the bid information from the UAV and starts the pre-authorization phase, the pseudo-code is shown in Algorithm 2.

As in lines 1~3 of Algorithm 2, *CCSys* waits to receive the bid information BID_i and then updates its own stored bidding matrix $P'(t)$ based on the bidding vector in the bid information, expressed as follows.

$$P^{'}(t) = \begin{bmatrix} P^{'}_{1,1}(t) & \cdots & P^{'}_{|U|,1}(t) \\ \vdots & \ddots & \vdots \\ P^{'}_{1,|T|}(t) & \cdots & P^{'}_{|U|,|T|}(t) \end{bmatrix}_{|T|\times|U|} \tag{34}$$

Pre-authorize the task T_j to the UAV with the highest bid $U_{i'j}$,

$$U_{i'j} = \arg\max_{U_i \in U} \left\{ P^{'}_{1,j}(t), P^{'}_{2,j}(t), ..., P^{'}_{|U|,j}(t), \right\} \tag{35}$$

We consider removing the quintet of relations $(U_i, L_i, MD_{ij}, t_i^1, T_j)$ associated with the task T_j from the set of pre-authorized information X, and adding new quintet of relations $(U_{i'j}, L_{i'j}, MD_{i'j\,j}, t_{i'j}^1, T_j)$, as shown in lines 4∼8.

Where line 7 indicates that once the task T_j is pre-authorized to the UAV $U_{i'j}$ at moment t, the price vector $\mathbf{P}(t)$ of all uncompleted tasks is updated, i.e., the price of the task T_j is updated to the bidding price of the UAV $U_{i'j}$,

$$p_j(t) = P^{'}_{i'j,j}(t) \tag{36}$$

As in lines 9∼13 of Algorithm 2, at the end of an auction round, *CCSys* sends a rebid command to the UAVs U_i that have not abandoned their bids and are not pre-authorized, and sends a set of current information about all tasks $TASK$. The UAVs that receive the rebidding command start a new round of the auction process, while the other UAVs enter the waiting phase. Lines 14∼15 indicate that the iterative process terminates with the condition that all UAVs that have not abandoned their bids during the auction process are pre-authorized for the task.

Algorithm 2: Pre-authorization phase (*CCSys* part).

Input : Bid vector BID_i;
Output: Set of task $TASK$.

1 $BID \leftarrow BID_i$;
2 wait;
3 $P^{'}(t) \leftarrow BID$;
4 **for** $T_j \leftarrow T_1$ **to** $T_{|T|}$ **do**
5     $U_{i'j} \leftarrow \arg\max_{U_i \in U}\{P^{'}_{1,j}(t), P^{'}_{2,j}(t), ..., P^{'}_{|U|,j}(t)\}$;
6   **if** $U_{i'j} \neq 0$ **then**
7         $p_j(t) \leftarrow P^{'}_{i'j,j}(t)$;
8         $X \leftarrow X - \{x_i\}$ and $X \leftarrow X \cup \{x_{i'j}\}$; `// Update the set of pre-authorization`
9 $flag \leftarrow 0$;
10 **for** $U_i \leftarrow U_1$ **to** $U_{|U|}$ **do**
11   **if** $\boldsymbol{P}_i^{'}(t) \neq (-\infty, -\infty, ..., -\infty)^T$ *and* $x_i == \emptyset$ **then** `// Indicate that` U_i `has not abandoned the bid and is not pre-authorized`
12   send $TASK$ to U_i and $flag \leftarrow 1$;
13   Execution: bidding phase;
14 **if** $flag \neq 1$ **then** `// Indicates the end of the pre-authorization phase of an auction`
15   Execution: authorization phase;

3.2.4. Authorization Phase

CCSys updates the set of authorization information Y based on the set of pre-authorization information X, adds the pre-authorization quintet relation to Y for UAVs U_i that are pre-authorized but not authorized, and removes it in X, while resetting the task price vector. $CCSys$ sends the authorization quintet y_i to the UAVs authorized in this auction, sends the loading information U_i, L_i to the MDC, and saves the set of pre-authorization information and the set of authorization information Y as the initial state for the next auction.

3.3. Algorithm Analysis

The algorithm involves internal system interaction and inevitably has communication problems. On the one hand, in the disaster area environment, ground communication base stations are damaged, and it is necessary to restore communication, and UAVs are mature and widely used as temporary communication relays [3]; on the other hand, the bidding process is performed by UAVs, which greatly reduces the state information interacted between $CCSys$ and UAVs, and can be used in a limited bandwidth environment [27].

The computation time of one auction in the SAABoPA algorithm is much less than that of the traditional auction algorithm [25], although the former requires multiple executions of the algorithm during the completion of all tasks, resulting in a longer cumulative execution time. However, considering the case of excessive interference, the traditional auction algorithm needs to reset or partially reset the previous allocation results and reallocate them. If we consider the case of poor communication, the conventional auction algorithm may cause excessive waiting time due to waiting for communication in a single execution, which may affect the timely departure of UAVs [28], and thus affect the capability performance and resilience performance of EMDS. In the EMDS framework, a single auction execution can be performed during the UAV's return trip or by using another gap time. As argued in the literature [45], real-time does not mean that the faster the computation time is, the more real-time it is, but that the solution to the problem is found within a limited time.

Specifically, we analyze the time complexity and space complexity of one auction round in a single auction of SAABoPA. As mentioned above, an auction is divided into the task publication phase, bidding phase, pre-authorization phase, and authorization phase, and the main complexity is concentrated in the bidding phase and pre-authorization phase. However, in the task publication phase and authorization phase, the main work is to complete the packaging, sending, and assignment of data processing operations. In the bidding phase, see (Algorithm 1), the main work is to complete the computation of the $INCOME_{i,T}(t)$ matrix, and the sorting operation to find the maximum element of the matrix. Assuming that the time complexity of computing a single matrix element is I and the space complexity is J, then the average time complexity is $O(|T| * L_i * I)$, and the space complexity is $O(|T| * L_i * J)$. On the other hand, the sorting algorithms are heap sorting, bubble sorting, and shell sorting, whose time complexity is $O(\log(|T| * L_i))$, $O((|T| * L_i)^2)$, and $O((|T| * L_i)^{1.3})$, respectively, and the space complexity is is $O(1)$. In the Pre-authorization Phase (Algorithm 2), the main work is to complete the sorting operation to find the maximum element of the bidding vector BID for each task, similarly using heap sorting, bubble sorting, shell sorting, their time complexity is $O(|T| * \log(L_i))$, $O(|T| * (L_i)^2)$ and $O(|T| * (L_i)^{1.3})$, respectively, and the space complexity is $O(1)$.

4. Computational Experiment

In this section, the effectiveness of the proposed method is verified by experiments. The simulation experiments were performed using Visual Studio 2022 platform, C++ programming language, CPU is Intel(R) Core(TM) i9-10885H CPU @ 2.40GHz 2.40GHz, OS is Windows 10.

4.1. Experimental Settings

4.1.1. Case Scenario

The relevant data for the computational case were chosen using a combination of real and hypothetical data, since some disaster data is not published or could not be obtained through official reports [46,47]. The data of the UAVs are derived from some public sources and scaled accordingly to fit the case scenario. Meanwhile, in order to verify the performance of the algorithm, large sample experiments are required in this section, and the data for the experimental scenarios are randomly generated on the basis of known partial data.

The earthquake disaster in a city in southwest China is simulated as a scenario, and the affected area C is a rectangular area of 4000 × 4000 m, with $|T|$ tasks uniformly distributed in the area, and the quantity of material required for the tasks $R(0)$ and the emergency indicator $E(0)$ also obey uniform distribution. The outside of the rectangular area is secure, and a well-stocked *MDC* is deployed on the boundary of the disaster area with a total of $|U|$ UAVs involved in the task execution. The number of the three dynamic interference types considered during the task execution is determined in advance as $|DIS^1|$, $|DIS^2|$, and $|DIS^3|$, but the time of occurrence of the dynamic interference types is random and uncertain, and the values of the changed information obey a uniform distribution. The specific parameters are shown in Table 1.

Table 1. Scenario parameters instructions.

Parameters	Value	Parameters	Value
x, y	$U(0, 4000)$	(x_0, y_0)	Boundary of C
$R(0)$	$DiscreteU(6, 10)$	$L^{\max}$	$U(11, 15)$
$E(0)$	$U(0.1, 0.8)$	V^{unload}	$U(15, 20)$
ΔR	$DiscreteU(0, 5)$	$T^{duration}$	$U(400, 500)$
ΔE	$U(0, 0.4)$	δ	0.5

In order to investigate the effects of different dynamic interference intensities on the performance of the algorithm, this paper designs experiments with seven different control variables at different scenario scales, as shown in Table 2.

Table 2. Scenario scale instructions.

Name of Scenario	$\|T\| \times \|U\| \times \|DIS^1\| \times \|DIS^2\| \times \|DIS^3\|$ [1]
Scenario 1	50 × 5 × 10 × 10 × 2
Scenario 2	50 × 5 × 20 × 10 × 2
Scenario 3	50 × 5 × 30 × 10 × 2
Scenario 4	50 × 5 × 20 × 20 × 2
Scenario 5	50 × 5 × 20 × 30 × 2
Scenario 6	50 × 5 × 20 × 10 × 3
Scenario 7	50 × 5 × 20 × 10 × 4

[1] Taking Scenario 2 as an example, 50 × 5 × 20 × 10 × 2 means that there are 50 tasks in the scenario, 5 UAVs, and the number of three dynamic disturbances *new task generations*, *task unexpected changes* and *UAV 's number decreases* are 20, 10 and 2, respectively.

4.1.2. Comparison Algorithms

The experiments with 1000 random samples are conducted for each scenario, respectively, considering all three interference types, and the SAABoPA algorithm under the EMDS framework is compared with the DTAP (DTA based on sequential single item auctions) [28] and CBBA-PR (consensus-based bundle algorithms-partial replanning) [26] algorithms to verify and compare the capability performance and resilience performance of EMDS under the three algorithms.

Numbered lists can be added as follows:

1. DTAP: Dynamic task assignment algorithm based on sequential single-item auctions is an online algorithm that differs from the algorithm in this paper in that there is no pre-authorization phase, and the DTAP algorithm starts once the auction timing is the current target task completion.
2. CBBA-PR: Dynamic task assignment based on the consistent consent bundle algorithm is an offline algorithm where one auction UAV gets one task sequence, i.e., one task

bundle. When interference occurs, a certain percentage of unexecuted tasks with low revenue are selected for authorization reset and the auction is restarted.

4.2. Performance Indicators

To verify the performance of EMDS based on the SAABoPA algorithm, the performance metrics fully consider the trade-off between efficiency and resilience [48]. In this paper, two main types of performance metrics are used: (a) capability performance, i.e., the completion of the rescue mission; and (b) resilience performance, i.e., the change of the system capability performance in the case of dynamic interference.

4.2.1. Capability Performance (CP)

In Definition 5, it is considered that when the emergency indicator reaches 1, the task will not be completed for a long time and the actual rescue will not be timely and cause secondary injuries. In this paper, the system capability performance is described as the proportion of tasks that are not subject to secondary injuries, i.e., the proportion of tasks whose emergency indicator fails to reach 1, which, combined with Equation (6), is expressed as follows.

$$CP = \frac{N - f}{N} \times 100\% \tag{37}$$

where N denotes the total number of tasks that occur during the entire EMDS emergency material delivery.

4.2.2. Resilience Performance (RP)

The EMDS capability performance metric proposed in this paper is time-invariant, not computable at any moment, and is obtained only at the end of the whole working process. In order to describe the calculation of the resilience performance metric, this paper considers the system capability performance as time-invariant and represents the in-process capability performance by the calculated value CP. When an interference situation occurs, for example, DIS^1 occurs at t_1^s, the system capability performance after t_1^s moment is denoted by the computed value CP; the computed value CP in the same initial situation, under the condition that DIS^1 does not occur at t_1^s moment denotes the system capacity performance before t_1^s moments. The same is true for the case of multiple interferences. The resulting EMDS capability performance metric is converted to time-dependent and the system capability performance changes as shown in Figure 3.

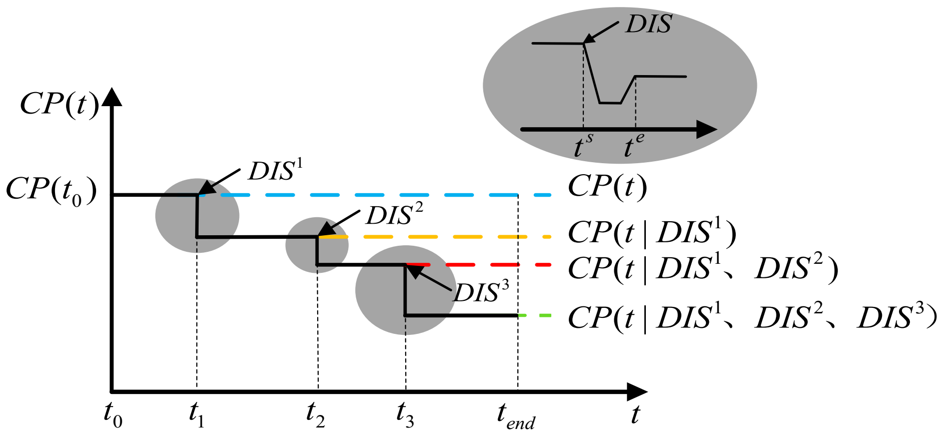


Figure 3. The change of system capability performance.

Since the capability performance at any moment of the whole working process is expressed by the calculated value CP, this paper does not describe too much the change of the system capability performance in a short time after the interference occurs, as Figure 3 incorporates the capability degradation, maintenance and recovery phases of the system after suffering from the interference [49].

Combined with the integral elasticity model, this paper defines the resilience performance of the system expressed as follows.

$$RP(\tau|DIS) = \frac{\int_{t_0}^{t_{end}} CP(\tau|DIS)d\tau}{\int_{t_0}^{t_{end}} CP(\tau)d\tau} \tag{38}$$

where $RP(\tau|DIS)$ denotes the system resilience performance with interference DIS occurring, $CP(\tau)$ denotes the system baseline capability performance when τ without interference, and $CP(\tau|DIS)$ denotes the system capability performance when τ with interference.

4.3. Algorithm Results of Hypothesis Testing: Friedman Test

Based on the experimental design of 1000 random instances of 7 scenarios in the previous paper, the three algorithms can obtain 7 sets of experimental results. We compare the computational results of the three algorithms pair-wisely for each group of results, from three perspectives of no interference capability performance, interference capability performance, and resilience performance, to test the performance of the SAABoPA algorithm proposed in this paper by whether there are significant differences. Current experiments on comparing the average performance of multiple algorithms on multiple problem instances are widely and well-known by methods such as Complete Block Design (CBD) ANOVA or Friedman's test [50]. In this paper, Friedman's test is used, but the Friedman's test can only test whether the average performance indicator value of at least one algorithm has a significant difference from the average performance indicator value of at least one algorithm, and cannot specifically determine which two algorithms have a significant difference between them, so Multcompare is also required. The results are shown in Figure A1. As an additional note, the confidence interval of Friedman's test in this paper is 99%.

From the results, in the seven scenarios, there are no significant differences in the capability performance of the system without interference under the SAABoPA and CBBA-PR algorithms, but there are significant differences with the DTAP algorithm. However, there are significant differences in the capability performance of the system with interferences under the three algorithms. It indicates that the SAABoPA and CBBA-PR algorithms have similar capability performance in the no interference case and are significantly better than the DTAP algorithm. However, in the case of interferences, the capability performance of the system under the CBBA-PR algorithm degrades more and is significantly worse than SAABoPA and DTAP algorithms. Analyzing Figure A1c,f,i,l,o,r,u together, it can be found that there are significant differences in the resilience performance of the system under the three algorithms. The exception is that in Figure A1f,i (scenarios 2 and 3), there is no significant difference between SAABoPA and DTAP algorithms regarding the system resilience performance, which may be because the resilience performance of them gradually approaches as the number of new tasks generation and the intensity of interference faced by the system increases. However, compared to scenarios 4 and 5, 6 and 7, the system is not yet facing the maximum intensity of interference, but the type of interference has changed, which reflects the greater adaptability of the SAABoPA algorithm to *new task generations* type of dynamic interferences. On the other hand, the specific performance gap also needs to be analyzed in terms of average, standard deviation, CDF, etc.

4.4. Algorithm Comparison Analysis: Scenario 2 as an Example

At the scale of Scenario 2, 1000 random sample experiments were conducted. The results of the simulation experiments are studied from the mean and variance of the metrics, as shown in Tables 3 and 4. The SAABoPA has a slightly lower capability performance than the CBBA-PR and roughly 6% higher than the DTAP in the static case where no interference occurs, because the DTAP, as an online algorithm, considers only the current deterministic information when making decisions, while the SAABoPA adds a pre-authorization process that considers the current information along with the next task estimation information, and achieved better results. However, it is obvious that the CBBA-PR, as an offline algorithm, considers all the information of the whole system working process, so the derived allocation scheme is better. It should be noticed that this algorithm requires higher accuracy of information, and in the presence of errors in the model, the errors in the offline solution will gradually accumulate causing the degradation of the solution quality.The resilience performance of SAABoPA and DTAP is similar, which is more than 5% higher than that of CBBA-PR, reflecting that the former two have better adaptability to the dynamic environment and resistance to interference situations are stronger. After analysis, the reason is that when there is an interference situation, the CBBA-PR needs to reallocate some of the uncompleted tasks with the interference in a comprehensive consideration, and considering the trade-off between algorithm execution time and optimality, it often cannot handle the interference situation in time, and it causes a significant decrease in the system capability performance, which reflects the lack of system resilience performance. Analyzing the mean value of system capability performance and the mean value of resilience performance under the three algorithms, although the mean value of resilience performance is above 0.85, the resilience indicator integrally considers the time of occurrence of multiple interferences, so the change of the calculated value of the indicator is relatively insignificant, but the capability performance after corresponding to multiple interferences will have a large reduction, for example, the highest resilient performance DTAP with multiple interferences and the ability to perform without interference differ by about 14%.

Analyzing the system performance variance, it is found that the variance of both metrics is around 0.01, which reflects the stability of the capacity performance and resilience performance of EMDS under the three algorithms in the scale of Scenario 2, and the experimental results are effective.

Table 3. Average of system performance with different algorithms for 1000 sample experiments.

Algorithm	$CP(\tau)$	$CP(\tau\|DIS)$	$RP(t\|DIS)$
SAABoPA	0.9491	0.7992	0.9443
DTAP	0.8867	0.7506	0.9459
CBBA-PR	0.9497	0.6663	0.8916

Table 4. Variance of system performance with different algorithms for 1000 sample experiments.

Algorithm	$CP(\tau)$	$CP(\tau\|DIS)$	$RP(t\|DIS)$
SAABoPA	0.0032	0.0102	0.0009
DTAP	0.0055	0.0115	0.0005
CBBA-PR	0.0031	0.0132	0.0013

A comparative analysis of the cumulative distribution functions (CDF) of the capability performance and resilience performance of SAABoPA, DTAP and CBBA-PR (Figure 4) shows that: (a) the CDF curves of the capability performance of the SAABoPA and CBBA-PR almost overlap in the initial case where no interference occurs, and the percentage of capability performance greater than 0.9 is 80% much higher than the 44% of the DTAP; (b) the capability performance of all three algorithms decreases in the presence of multiple disturbances. However, the SAABoPA maintains the best capability performance with 50% of capability performance greater than 0.8 higher than 31% of DTAP and 10% of CBBA-PR; (c) the CDF curves of the resilience performance of SAABoPA and DTAP are close, and the percentage of resilience performance greater than 0.9 is over 90% much higher than 41% of CBBA-PR algorithm. Moreover, the lower limit of the resilience performance of SAABoPA and DTAP still exceeds 0.8, which indicates that SAABoPA and DTAP have stronger resilience performance and better adaptability in the face of interference situations.

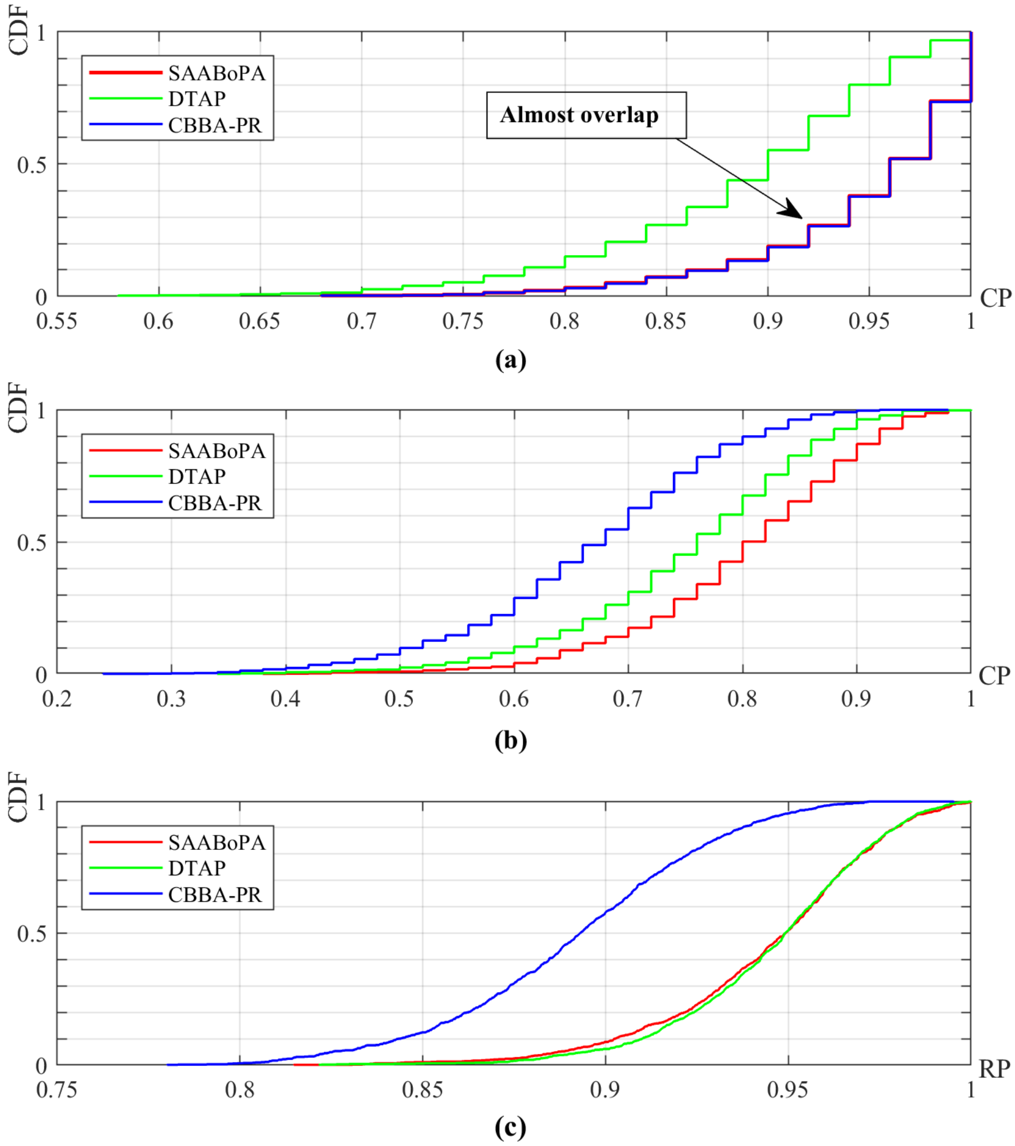


Figure 4. Comparison of CDF of system performance under different algorithms: (**a**) Capability performance without dynamic interference; (**b**) Capability performance with dynamic interference; (**c**) Resilient performance.

For further analysis, the performance metrics of each sample were compared as shown in Figure 5. To facilitate the observation of the experimental results, we adjusted the sample serial numbers and reordered the samples in the lowest to highest order of SAABoPA's metrics. We found that although the experimental parameters are the same, the disaster area environment and UAV parameters (e.g., mission coordinates, UAV speed, etc.) are different for different samples, resulting in different performance metrics for different samples by different algorithms, which we believe is reasonable, and there is a certain applicability of different algorithms to different scenarios themselves, which is why we consider large sample experiments.

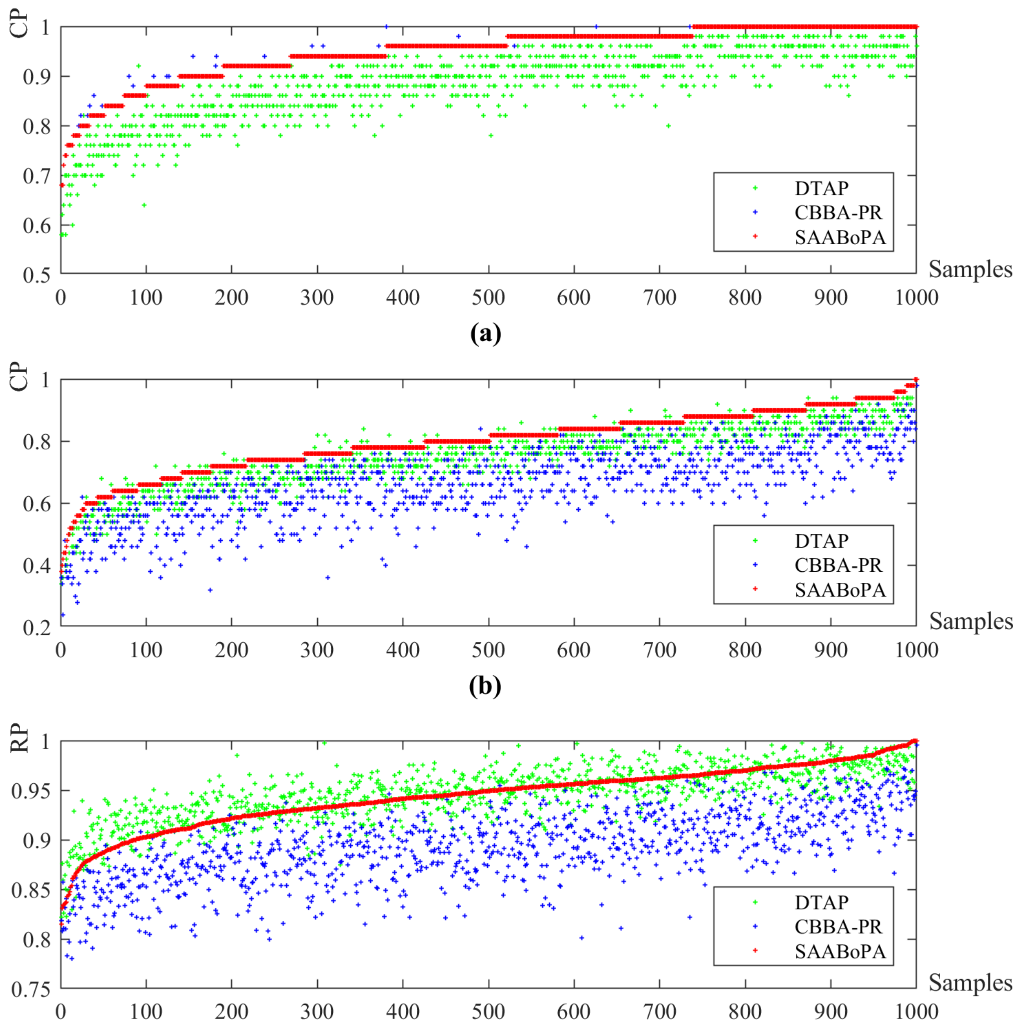


Figure 5. Comparison of system performance under different algorithms in each sample scenario: (**a**) Capability performance without dynamic interference; (**b**) Capability performance with dynamic interference; (**c**) Resilient performance.

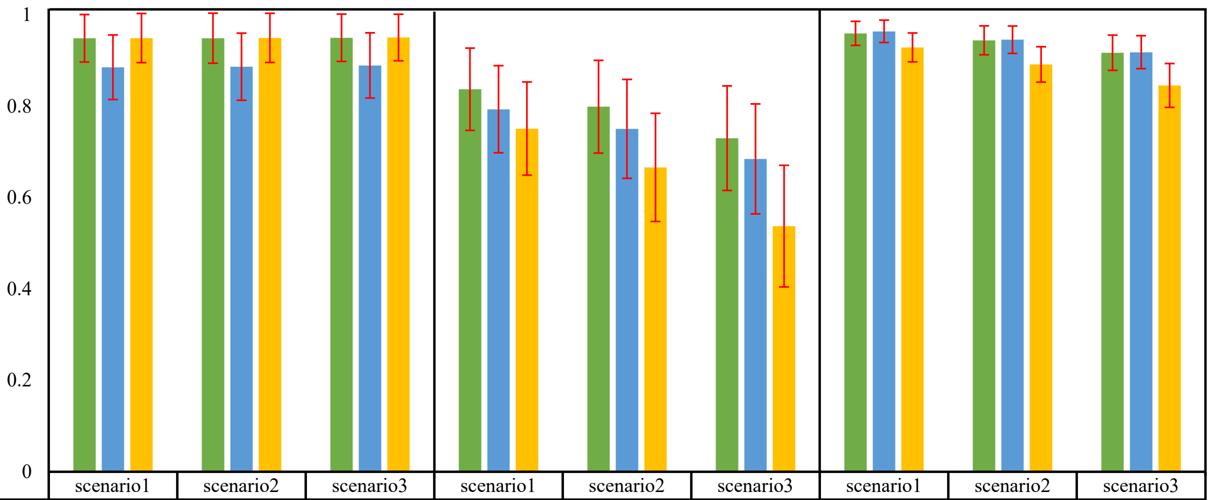
4.5. Analysis of the Results under Different Dynamic Interference Intensity

As for the seven scenarios shown in Table 2, they are divided into three groups (scenario 1, 2, 3, scenario 2, 4, 5 and scenario 2, 6, 7) according to the principle of controlled variable method in this paper to study the effects of *new task generations*, *task unexpected changes* and *UAV 's number decreases* at different intensities on the performance of EMDS, as shown in Figures 6–8. From the experimental results, the more *new task generations* type of dynamic interferences have a certain degradation on the capability performance and resilience performance of all three algorithms. The capability performance of the SAABoPA proposed in this paper is optimal under different intensities of dynamic interference. On the other hand, when the number of new tasks increases from 10 to 30 (from Scenario 1 to Scenario 3), the capability performance of SAABoPA and DTAP decreases by about 10% and the resilience performance decreases by about 4%, but the CBBA-PA decreases by about 22% and 8%, respectively, which reflects the importance of the resilience performance indicator in the scenario with dynamic interference, which can better describe the performance of EMDS. *task unexpected changes* and *UAV's number decreases* type of dynamic interference at different intensities can still obtain the same conclusion as *new task generations*.

In the vertical comparison, compared with Scenario 2, the number of *new tasks generations* and *task unexpected changes* increases by 10 in Scenario 3 and Scenario 4, respectively, but the capability performance of the three algorithms decreases by approximately 7%, 7% and 13% in Scenario 3, and the resilience performance decreases by 3%, 3% and 5%, respectively; then the capability performance of the three algorithms decreases by approximately 4%, 2% and 7% in Scenario 4, and the resilience performance decreases by 1%, 1% and 4%, respectively. In Scenario 4, the capability performance of the three algorithms decreases by about 4%, 2% and 7%, and the resilience performance decreases by 1%, 1% and 4%, respectively. This reflects the fact that dynamic interference of the *new tasks generations* type has a greater impact on the system capability performance and resilience performance than *task unexpected changes*, which is reasonable. However, since *UAV's number decreases* type of dynamic interference is the reduction in task execution vehicles, it is not comparable to *new tasks generations* and *task unexpected changes*.

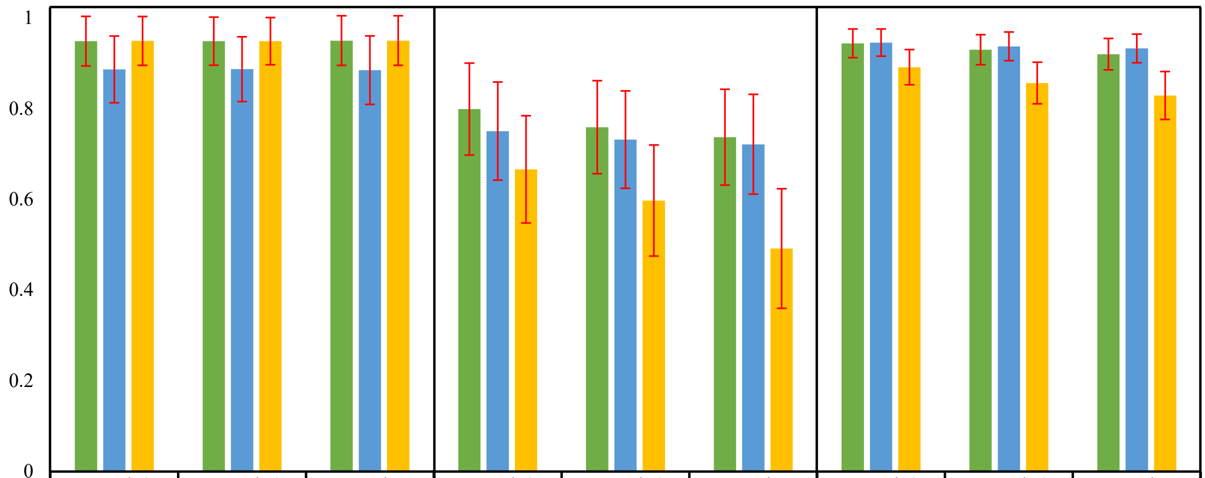
To analyze the dynamic interference of *UAV's number decreases* type individually, from scenario 2 to scenario 6 to scenario 7, the UAVs are reduced by 1 in turn, which means that the transport delivery capacity is reduced by 20% in turn (total number = 5), while the capability performance reductions of the three algorithms become larger from 8%, 7%, and 9%, respectively, with reductions of 13%, 19%, 12%; on the other hand, the reduction in resilience performance becomes 4%, 4%, 4% from 3%, 2%, 3%, respectively. This indicates that the smaller the number of available UAVs, the worse the system performance, and this gap is increasing as the number of UAVs decreases. It can also be found that the DTAP has a more drastic performance degradation compared to the SAABoPA and CBBA-PA algorithms in scenario 7 (loss of 4 UAVs), and the reason for this analysis is that the capability performance of EMDS under the DTAP is not high when no such interference occurs, and the more UAVs are lost to the DTAP brings about a further degradation of the task planning capability.

Comparing the standard deviations of the experimental results under each experimental scenario, it can be found that the resilience indicator RP is smaller compared to CP under no interference and CP under multiple interferences, reflecting that the resilience indicator of multiple interferences proposed in this paper is more stable under large sample experiments, and it is robust and convincing to describe the system resilience performance by RP.



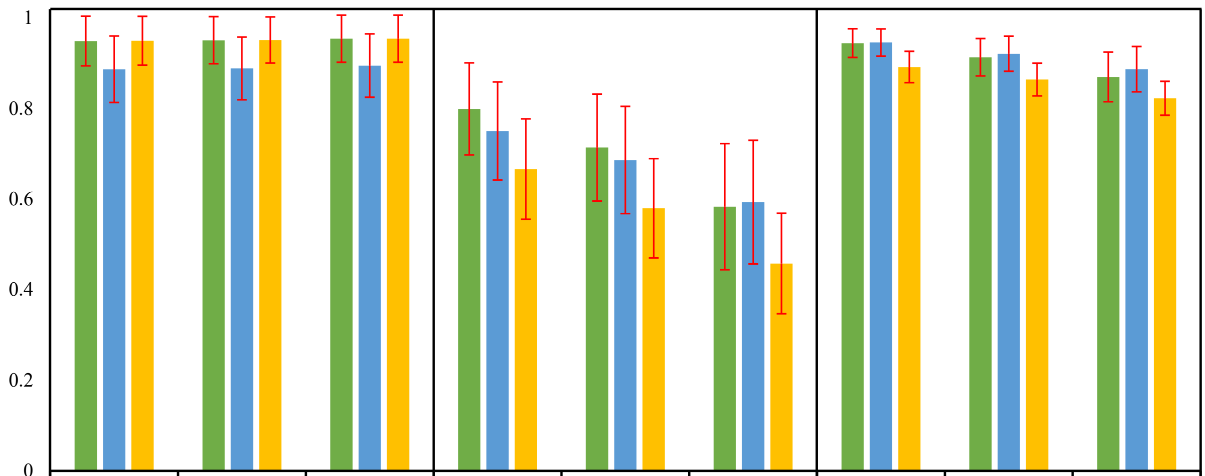
	$CP(\tau)$			$CP(\tau\mid DIS)$			$RP(\tau\mid DIS)$		
	scenario1	scenario2	scenario3	scenario1	scenario2	scenario3	scenario1	scenario2	scenario3
SAABoPA	0.9490	0.9491	0.9501	0.8373	0.7992	0.7301	0.9597	0.9443	0.9173
DTAP	0.8854	0.8867	0.8894	0.7935	0.7506	0.6848	0.9641	0.9459	0.9184
CBBA-PR	0.9495	0.9497	0.9507	0.7511	0.6663	0.5376	0.9288	0.8916	0.8456

Figure 6. Comparison of system performance of each algorithm at different dynamic strengths (different number of *new task generations*).



	scenario2	scenario4	scenario5	scenario2	scenario4	scenario5	scenario2	scenario4	scenario5
SAABoPA	0.9491	0.9490	0.9503	0.7992	0.7593	0.7373	0.9443	0.9302	0.9203
DTAP	0.8867	0.8874	0.8851	0.7506	0.7320	0.7216	0.9459	0.9375	0.9331
CBBA-PR	0.9497	0.9490	0.9503	0.6663	0.5976	0.4918	0.8916	0.8567	0.8292
	$CP(\tau)$			$CP(\tau \mid DIS)$			$RP(\tau \mid DIS)$		

Figure 7. Comparison of system performance of each algorithm at different dynamic strengths (different number of *task unexpected changes*).



	scenario2	scenario6	scenario7	scenario2	scenario6	scenario7	scenario2	scenario6	scenario7
SAABoPA	0.9491	0.9508	0.9542	0.7992	0.7139	0.5833	0.9443	0.9132	0.8699
DTAP	0.8867	0.8887	0.8948	0.7506	0.6863	0.5934	0.9459	0.9210	0.8869
CBBA-PR	0.9497	0.9514	0.9542	0.6663	0.5798	0.4579	0.8916	0.8641	0.8227
	$CP(\tau)$			$CP(\tau \mid DIS)$			$RP(\tau \mid DIS)$		

Figure 8. Comparison of system performance of each algorithm at different dynamic strengths (different number of *UAV's number decreases*).

In summary, the capability performance of the SAABoPA proposed in this paper is similar to that of CBBA-PA under no interference and better than that of the DTAP; the capability performance of the SAABoPA under multiple interference is better than that of the DTAP and CBBA-PA; the resilience performance of the SAABoPA under multiple interference is similar to that of the DTAP and better than that of the CBBA-PA, and it is more adaptable to dynamic scenarios.

5. Conclusions

This paper presents a basic framework for applying UAVs as important components of EMDS to emergency material delivery scenarios, and proposes SAABoPA for capability performance and resilience performance optimization. Combined with the actual material delivery, the problem is modeled as MTTDDVRP-SD on the basis of the VRP problem, while several features are considered, including multi-trip per UAV, time-dependent emergency indicator, dynamics of the environment, split-delivery, and trip duration limit. Three types of dynamic interference scenarios are considered, including: *new task generation*, *task unexpected changes*, and *UAV's number decreases*. To solve this new problem, this paper proposes the SAABoPA algorithm, which adds a pre-authorization and sequential auction mechanism. Meanwhile, in order to describe the effect of dynamic disturbances on EMDS, we improve the integral resilience model under single interference and propose a resilience performance indicator adapted to multiple interference scenarios. Finally, through comparison experiments with DTAP and CBBA-PR in scenarios with different dynamic intensities, and using Friedman's test with 99% confidence interval, we verified the effectiveness of SAABoPA, which can improve the resilience performance of the system while maintaining a better capability performance, compensating for the shortcomings of CBBA-PR which only focuses on capability performance and DTAP algorithm which only focuses on resilience performance.

It should be noted that the scenario in this paper is material delivery under disaster area rescue, which mainly considers task survival measured by task emergency indicator as the optimization objective, and may need to trade-off more objectives if it is applied to other scenarios such as UAV logistics, combat equipment delivery, and cluster fire strike. In the future, we consider building a small hardware experiment bed to test the algorithm effectiveness and system performance, and try to solve more practical problems to extend to real applications.

Author Contributions: Conceptualization, J.Z. and Y.Z.; methodology, J.Z.; software, J.Z.; validation, J.Z., T.W. and W.W.; formal analysis, R.W.; investigation, X.L.; resources, X.L.; data curation, Y.Z.; writing—original draft preparation, J.Z.; writing—review and editing, T.W.; visualization, R.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported in part by the National Natural Science Foundation of China under Grant 72101263.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Notations

Some of the parameters involved in the model and their meanings are as follows.

Sets

U	Set of UAVs
V_B	Set of MDC, $V_B = \{V_0\}$
DIS^1	Set of dynamic interference types 1
DIS^2	Set of dynamic interference types 2
DIS^3	Set of dynamic interference types 3

Indices

i	Index of UAV serial numbers, $U_i \in U$
j	Index of task serial numbers, $T_j \in T$
k	Index of UAV trips, $k \in \{1, 2, ..., Ex_i\}$
n^1, n^2, n^3	Index of the serial numbers of dynamic interference types 1, 2 and 3, which are parameters of the stochastic process, $n^1 \in \{1, 2, ..., \lvert DIS^1 \rvert\}$, $n^2 \in \{1, 2, ..., \lvert DIS^2 \rvert\}$, $n^3 \in \{1, 2, ..., \lvert DIS^3 \rvert\}$

Operator	
$\lvert \bullet \rvert$	Number of elements in the set
$\tilde{\bullet}$	Random variable or random vector (bold font)
$\overline{\bullet}$	Upper limit of possible values of the random variable
$\underline{\bullet}$	Lower limit on the possible values of the random variable
$U(\bullet, \bullet)$	Continuous uniform distribution between two values
$DiscreteU(\bullet, \bullet)$	Discrete uniform distribution between two values
Parameters	
$T_i^{duration}$	Maximum duration of the UAV U_i
δ	Ratio coefficient of the speed of the UAV with the weight of the loading
a	Constant parameter
$L_i^{\max}$	Upper limit of the loading capacity of the UAV U_i
V_i^{unload}	Speed of the UAV U_i at no-load
(x_j, y_j)	Two-dimensional coordinates of the task T_j
(x_0, y_0)	Two-dimensional coordinates of the MDC V_0
Variables	
t	Discrete time series
T_{end}	Moment of completion of emergency material delivery task
Ex_i	Number of UAV U_i trips
sT_i^k	Departure time of the k-th trip of the UAV U_i
aT_i^k	Arrival time of the k-th trip of the UAV U_i
$v_i(t)$	Speed of the UAV U_i at t moments
$E_j(t)$	Emergency indicator of the task T_j at t moments
$R_j(t)$	Quantity of material Requirement of the task T_j at t moments
$L_i^k(t)$	Loading of the k-th trip of the UAV U_i at t moments
$MD_{ij}(t)$	Quantity of material delivered by the UAV U_i to the task T_j at t moments
$(x_i(t), y_i(t))$	Two-dimensional coordinates of the UAV U_i at t moments
ΔR	Quantity of change in the material requirements of the task
ΔE	Value of change in the material emergency indicators of the task
MD_{ij}^k	Decision variable, quantity of material delivered by the UAV U_i to the task T_j in k-th trip
x_{ij}^k	Binary decision variable, if the UAV U_i execute task T_j in k-th trip, $x_{ij}^k = 1$; otherwise $x_{ij}^k = 0$

Appendix A

The results of pair-wise comparisons of 1000 random instances for 7 scenarios are shown in Figure A1, which provide the confidence intervals (adjusted for a familywise error rate of $\alpha = 0.01$) for all comparisons between the SAABoPA algorithm and the comparison algorithm. Note that in each subplot of Figure A1, blue and red indicate that the results of the two algorithms for this scenario have significant differences at the 99% confidence intervals, while blue and black indicate that the results of the two algorithms for this scenario have no significant differences at the 99% confidence intervals.

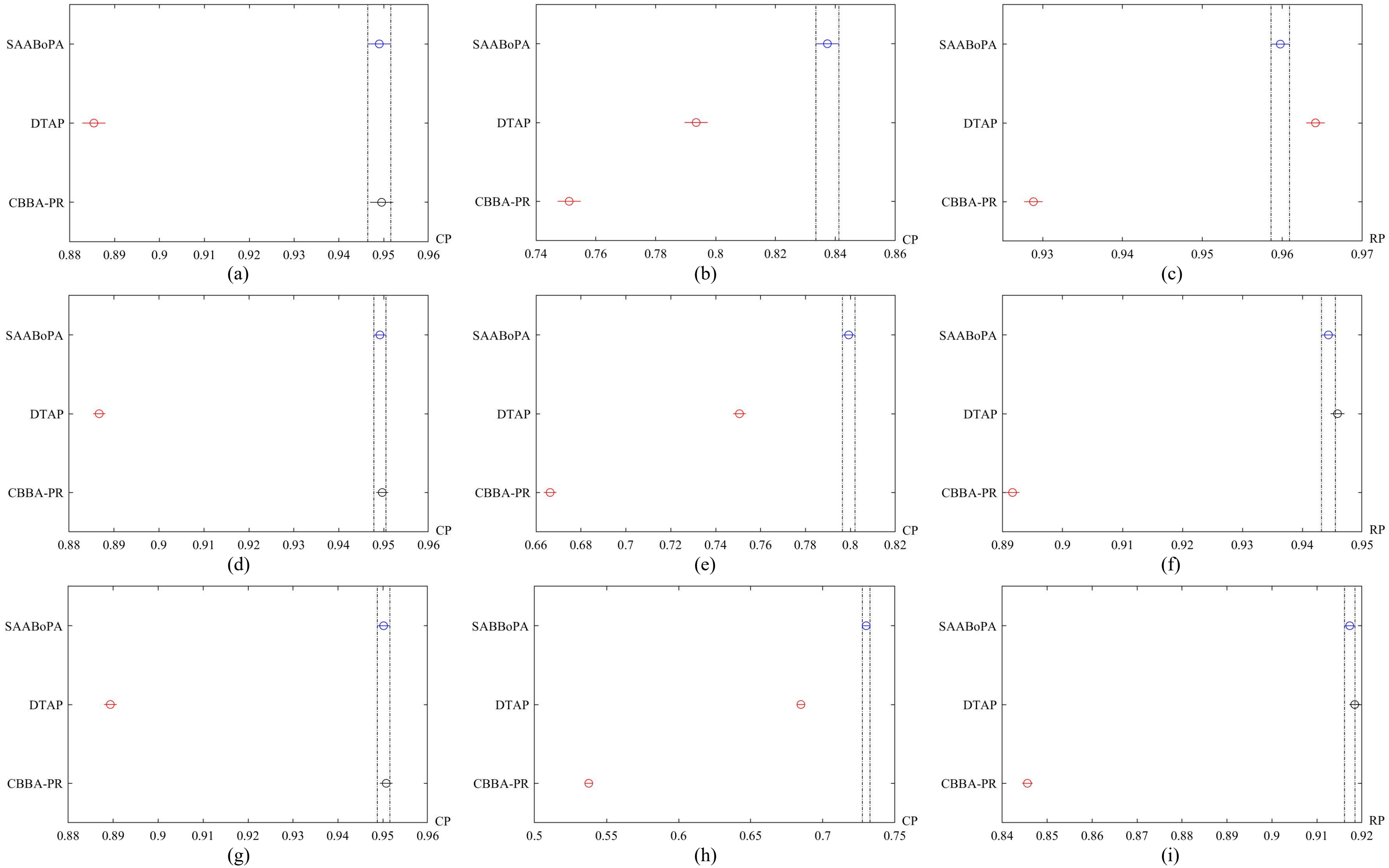


Figure A1. *Cont*.

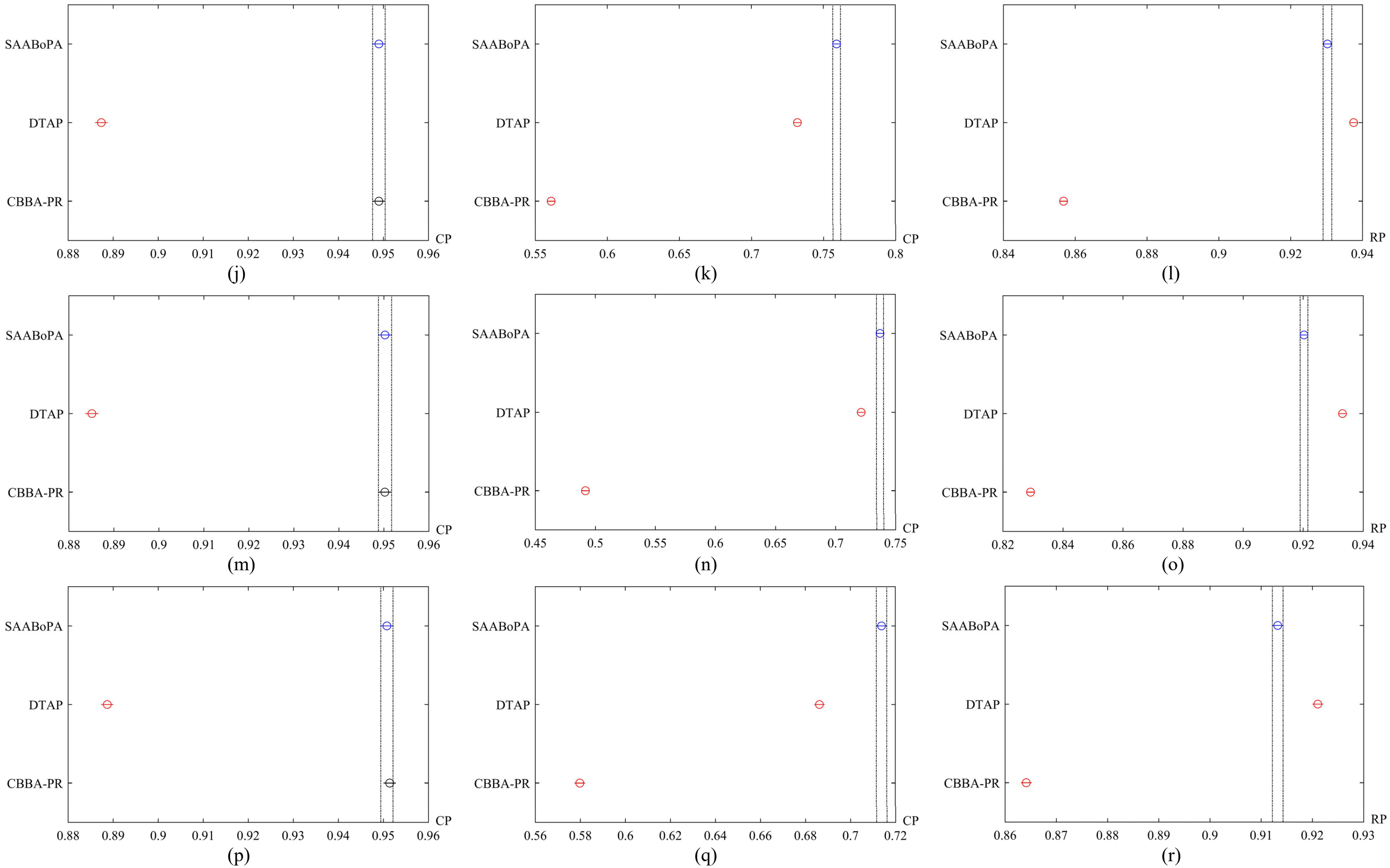
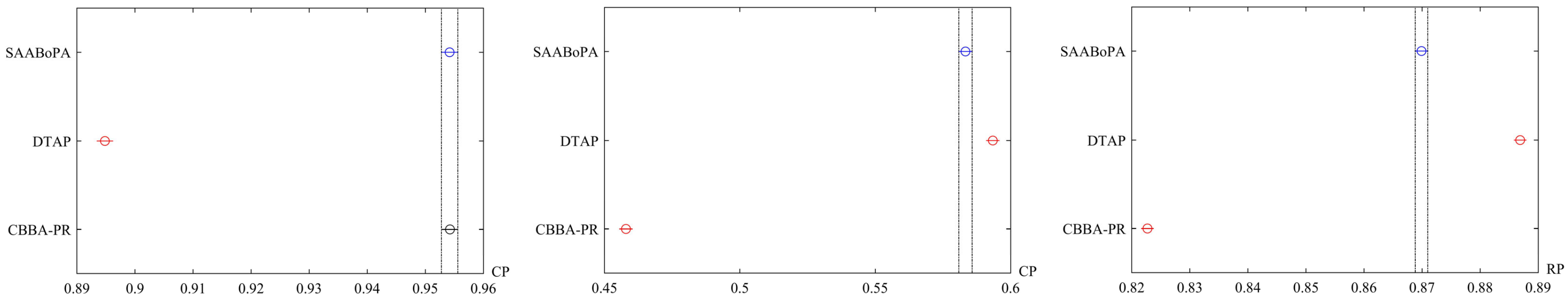


Figure A1. *Cont.*



(s) (t) (u)

Figure A1. Pair-wise comparison of system performance under different algorithms in 7 scenarios: (**a,d,g,j,m,p,s**) Capability performance without dynamic interference of scenarios 1 to 7, respectively; (**b,e,h,k,n,q,t**) Capability performance with dynamic interference of scenarios 1 to 7, respectively; (**c,f,i,l,o,r,u**) Resilient performance of scenarios 1 to 7, respectively.

References

1. Pérez-Rodríguez, N.; Holguín-Veras, J. Inventory-allocation distribution models for postdisaster humanitarian logistics with explicit consideration of deprivation costs. *Transp. Sci.* **2016**, *50*, 1261–1285. [CrossRef]
2. Saini, K.; Sood, S.K. Exploring the emerging ict trends in seismic hazard by scientometric analysis during 2010–2019. *Environ. Earth Sci.* **2021**, *80*, 1–25. [CrossRef]
3. Ng, J.S.; Lim, W.Y.B.; Dai, H.-N.; Xiong, Z.; Huang, J.; Niyato, D.; Hua, X.-S.; Leung, C.; Miao, C. Joint auction-coalition formation framework for communication-efficient federated learning in uav-enabled internet of vehicles. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 2326–2344. [CrossRef]
4. Hou, H.; Yang, J.; Chen, Y.; Yu, M. A material delivery system built on internet of things and the role of libraries. *Libr. Tech.* **2019**, *38*, 195–209. [CrossRef]
5. Song, X.; Zheng, Y.; Chang, C. An emergency dispatch model based on variable precision rough sets. *Inf. Control.* **2011**, *40*, 858–864.
6. Guan, G.; Lin, Z.; Gong, Y.; Jiang, Z. Modeling and simulation of collaborative dispatching of disaster relief materials based on urgency. *Math. Probl. Eng.* **2020**, *2020*, 4274106. [CrossRef]
7. Taillard, É.D.; Laporte, G.; Gendreau, M. Vehicle routeing with multiple use of vehicles. *J. Oper. Res. Soc.* **1996**, *47*, 1065–1070. [CrossRef]
8. Liu, Y.; Liu, Z.; Shi, J.; Wu, G.; Pedrycz, W. Two-echelon routing problem for parcel delivery by cooperated truck and drone. *IEEE Trans. Syst. Man Cybern. Syst.* **2020**, *51*, 7450–7465. [CrossRef]
9. Sadrani, M.; Tirachini, A.; Antoniou, C. Vehicle dispatching plan for minimizing passenger waiting time in a corridor with buses of different sizes: Model formulation and solution approaches. *Eur. J. Oper. Res.* **2022**, *299*, 263–282. [CrossRef]
10. Cattaruzza, D.; Absi, N.; Feillet, D. Vehicle routing problems with multiple trips. *Ann. Oper. Res.* **2018**, *271*, 127–159.
11. Dorling, K.; Heinrichs, J.; Messier, G.G.; Magierowski, S. Vehicle routing problems for drone delivery. *IEEE Trans. Syst. Man, Cybern. Syst.* **2016**, *47*, 70–85. [CrossRef]
12. Sun, Y.; Wang, D.; Lang, M.; Zhou, X. Solving the time-dependent multi-trip vehicle routing problem with time windows and an improved travel speed model by a hybrid solution algorithm. *Clust. Comput.* **2019**, *22*, 15459–15470. [CrossRef]
13. Rios, B.H.O.; Xavier, E.C.; Miyazawa, F.K.; Amorim, P.; Curcio, E.; Santos, M.J. Recent dynamic vehicle routing problems: A survey. *Comput. Ind. Eng.* **2021**, *160*, 107604. [CrossRef]
14. Li, J.; Xiong, Y.; Jiang, B. Distribution fairness in emergency material vehicle scheduling based on uncertain time and demand. *J. Uncertain Syst.* **2021**, *14*, 2150014. [CrossRef]
15. Nan, L.; Chen, Y.; Zhang, Z. An improved adaptive large-scale neighborhood search algorithm solves hybrid vehicle path problems with dynamic demand. *Comput. Appl. Res.* **2021**, *38*, 9.
16. Sabar, N.R.; Bhaskar, A.; Chung, E.; Turky, A.; Song, A. A self-adaptive evolutionary algorithm for dynamic vehicle routing problems with traffic congestion. *Swarm Evol. Comput.* **2019**, *44*, 1018–1027. [CrossRef]
17. Ouaddi, K.; Mhada, F.; Benadada, Y. Memetic algorithm for multi-tours dynamic vehicle routing problem with overtime (mdvrpot). *Int. J. Ind. Eng. Comput.* **2020**, *11*, 643–662. [CrossRef]
18. Casazza, M.; Ceselli, A.; Calvo, R.W. A route decomposition approach for the single commodity split pickup and split delivery vehicle routing problem. *Eur. J. Oper. Res.* **2021**, *289*, 897–911. [CrossRef]
19. Hartomo, K.D.; Purnomo, H.D.; Anwar, M.T. Routing model for medium disaster relief operations. In Proceedings of the 2017 International Conference on Innovative and Creative Information Technology (ICITech), Salatiga, Indonesia, 2–4 November 2017; pp. 1–6.
20. Zhang, P.; Xiao, K.; Fu, C.; Yang, K. A fast dynamic programming algorithm for solving large-scale vcvrp problems. *Syst. Eng. Theory Pract.* **2016**, *36*, 12.
21. Mingozzi, A.; Roberti, R.; Toth, P. An exact algorithm for the multitrip vehicle routing problem. *INFORMS J. Comput.* **2013**, *25*, 193–207. [CrossRef]
22. Kilby, P.; Prosser, P.; Shaw, P. Dynamic vrps: A study of scenarios. *Univ. Strathclyde Tech. Rep.* **1998**, *1*, 11.
23. Montemanni, R.; Gambardella, L.M.; Rizzoli, A.E.; Donati, A.V. Ant colony system for a dynamic vehicle routing problem. *J. Comb. Optim.* **2005**, *10*, 327–343. [CrossRef]
24. Guo, X.; Ji, M.; Wen, D.; Zhang, X.; Tian, S. Distributed multi-uav tasking and path planning for 'last-mile' delivery. *Syst. Eng. Theory Pract.* **2021**. [CrossRef]
25. Choi, H.-L.; Brunet, L.; How, J.P. Consensus-based decentralized auctions for robust task allocation. *IEEE Trans. Robot.* **2009**, *25*, 912–926. [CrossRef]
26. Buckman, N.; Choi, H.-L.; How, J.P. Partial replanning for decentralized dynamic task allocation. In *AIAA Scitech 2019 Forum*; American Institute of Aeronautics and Astronautics, Inc.: San Diego, CA, USA, 2019; p. 0915.
27. Chen, C.; Bao, W.; Men, T.; Zhu, X.; Wang, J.; Wang, R. Nectar-an agent-based dynamic task allocation algorithm in the uav swarm. *Complexity* **2020**, *2020*, 6747985. [CrossRef]
28. Farinelli, A.; Iocchi, L.; Nardi, D. Distributed on-line dynamic task assignment for multi-robot patrolling. *Auton. Robot.* **2017**, *41*, 1321–1345. [CrossRef]
29. Luo, L.; Chakraborty, N.; Sycara, K. Distributed algorithms for multirobot task assignment with task deadline constraints. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 876–888. [CrossRef]

30. Chung, K.; Park, R.C. P2p cloud network services for iot based disaster situations information. *Peer-to-Peer Netw. Appl.* **2016**, *9*, 566–577. [CrossRef]
31. Vogel, R.M.; Fennessey, N.M.; Bolognese, R.A. Storage-reliability-resilience-yield relations for northeastern united states. *J. Water Resour. Plan. Manag.* **1995**, *121*, 365–374. [CrossRef]
32. Woods, D. Escaping failures of foresight. *Saf. Sci.* **2009**, *4*, 498–501. [CrossRef]
33. Yodo, N.; Wang, P.; Zhou, Z. Predictive resilience analysis of complex systems using dynamic bayesian networks. *IEEE Trans. Reliab.* **2017**, *66*, 761–770. [CrossRef]
34. Pumpuni-Lenss, G.; Blackburn, T.; Garstenauer, A. Resilience in complex systems: An agent-based approach. *Syst. Eng.* **2017**, *20*, 158–172. [CrossRef]
35. Liaghati, H.R.; Mazzuchi, T.A.; Sarkani, S. Utilizing a maximin optimization approach to maximize system resiliency. *Syst. Eng.* **2021**, *24*, 144–157. [CrossRef]
36. Ouyang, M.; Duenas-Osorio, L. Time-dependent resilience assessment and improvement of urban infrastructure systems. *Chaos Interdiscip. J. Nonlinear Sci.* **2012**, *22*, 033122. [CrossRef] [PubMed]
37. Dessavre, D.G.; Ramirez-Marquez, J.E.; Barker, K. Multidimensional approach to complex system resilience analysis. *Reliab. Eng. Syst. Saf.* **2016**, *149*, 34–43. [CrossRef]
38. Jing, T.; Wang, T.; Wang, W.; Li, X.; Zhou, X. An adaptive regulation method for cluster size of continuous reconnaissance uavs. *J. Comput. Res. Dev.* **2018**, *55*, 9.
39. Arya, S.R.; Ashokkumar, C.R.; Arya, H. Gamma and velocity tracking for uav obstacle avoidance in pitch plane. In Proceedings of the 2016 Indian Control Conference (ICC), Hyderabad, India, 4–6 January 2016; pp. 362–368.
40. Song, J.M.; Chen, W.; Lei, L. Supply chain flexibility and operations optimisation under demand uncertainty: A case in disaster relief. *Int. J. Prod. Res.* **2018**, *56*, 3699–3713. [CrossRef]
41. Zhou, Y.; Chen, N.; Li, Z.; Gong, Y. Optimal design of emergency logistics network at the initial stage of post-earthquake relief considering facility disruption scenarios. *Oper. Res. Manag.* **2020**, *29*, 6.
42. Lv, G.; Peng, Z.; Wang, D.; Dou, W. Target task assignment for unmanned ship cluster formation reconfiguration. *Chinese Nav. Res.* **2018**, *13*, 6.
43. Vincent, F.Y.; Widjaja, A.T.; Gunawan, A.; Vansteenwegen, P. The multi-vehicle cyclic inventory routing problem: Formulation and a metaheuristic approach. *Comput. Ind. Eng.* **2021**, *157*, 107320.
44. Peng, Y.; Li, Y. Collaborative truck-drone delivery path optimization considering the impact of epidemic. *Chinese J. Highw.* **2020**, *33*, 73.
45. Zhao, M. Research and Simulation of Distributed Multi-Type UAV Cooperative Mission Assignment. Ph.D. Thesis, Nanjing University of Science and Technology, Nanjing, China, 2009.
46. Song, X.; Wang, J.; Chang, C. Nonlinear continuous consumption emergency scheduling model and algorithm based on demand urgency. *Inf. Control* **2014**, *43*, 9.
47. Wang, Y.; Sun, B. Multiperiod optimal emergency material allocation considering road network damage and risk under uncertain conditions. *Oper. Res.* **2021**, *22*, 2173–2208. [CrossRef]
48. Patterson, M.D.; Wears, R.L. Resilience and precarious success. *Reliab. Eng. Syst. Saf.* **2015**, *141*, 45–53. [CrossRef]
49. Francis, R.; Bekera, B. A metric and frameworks for resilience analysis of engineered and infrastructure systems. *Reliab. Eng. Syst. Saf.* **2014**, *121*, 90–103. [CrossRef]
50. Campelo, F.; Wanner, E.F. Sample size calculations for the experimental comparison of multiple algorithms on multiple problem instances. *J. Heuristics* **2020**, *26*, 851–883. [CrossRef]