Annals of Operations Research (2023) 321:717–753
https://doi.org/10.1007/s10479-022-05094-4

ORIGINAL RESEARCH

Scheduling operating rooms of multiple hospitals considering transportation and deterioration in mass-casualty incidents

Shuwan Zhu[1,2] · Wenjuan Fan[1,2] · Shanlin Yang[1,2] · Panos M. Pardalos[3]

Accepted: 17 November 2022 / Published online: 5 December 2022

Abstract

In mass casualty incidents, patients need to be evacuated to nearby hospitals as soon as possible, and a surge in demand for emergency medical services then occurs. It would result in ambulance offload delays, i.e., no emergency operating room is available when the ambulance arrives at a hospital, and thus the patients cannot be treated immediately. In this paper, we aim to solve a combinatorial problem of patient-to-hospital assignment and patient surgery sequence considering patient deterioration and ambulance offload delay during a mass casualty incident. A mixed-integer programming model is proposed. The objective is to minimize the completion time of all patients' surgeries. For solving such a problem, some structural properties of our studied problem are derived, and a heuristic is developed to solve the single operating room scheduling problem considering ambulance offload delay and patient deterioration based on these structural properties. A hybrid Firefly Algorithm-Variable Neighborhood Search algorithm incorporating the heuristic method is proposed to solve it. Our proposed algorithm can solve the problem within a short computation time, and the computational results demonstrate the superiority of our proposed algorithm over the compared algorithms.

Keywords Operating room · Scheduling · Mass casualty incident · Ambulance dispatching · Heuristics · FA-VNS

1 Introduction

A mass casualty incident (MCI), also called a multiple-casualty incident or multiple-casualty situation, means an incident in which the emergency medical service resources (e.g., human and facility) are overwhelmed by the number and severity of casualties. MCIs include disasters

✉ Wenjuan Fan
fanwenjuan@hfut.edu.cn

[1] School of Management, Hefei University of Technology, Hefei, China

[2] Key Laboratory of Process Optimization and Intelligent Decision-Making of Ministry of Education, Hefei, China

[3] Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL, USA

caused by natural catastrophes (such as earthquakes, hurricane, or volcanic activity) or man-made calamities (such as traffic accidents, terrorist attacks, or public health emergencies). Even an aircraft running off the end of a runway when landing at the airport (Dean & Nair, 2014), a bus fire, or an industrial explosion could almost instantly generate 10–125 severely injured casualties which is far more than the existing medical service resources can manage (Melton & Riner, 1981). A recent scientific framework has been developed using expert opinions, which defines an MCI as any event that results in more than 5 casualties (Kim et al., 2014). For example, under the circumstance when a two-person staff is responding to a motor accident with three critically injured casualties could be regarded as an MCI.

In China, due to the imbalanced development among rural areas and urban areas, there is usually few ambulances available in some rural areas (Yan et al., 2017). Some researchers point out that the number of negative pressure isolation ambulances is only 0.15 per health service institution on average (Ye, 2018). Especially for the mass casualty incidents, the available hospital emergency medical resources are limited and the ambulance utilization is very high (Repoussis et al., 2016). The injured must be rescued, triaged and dispersed to nearby hospitals The information available to medical resource owners, e.g., local hospitals, is often incomplete. In an emergency response time, ambulances and emergency operating rooms are in short supply beyond question. Usually, each hospital has its own ambulance site, while their ambulances are directed by the ambulance dispatching center. The decisions made by different medical resource owners/managers are based on partial and fragmented information, and thus affect the overall efficiency (Besiou et al., 2018). It would experience a high influx of patients and the hospitals would soon be overburdened. Investigators from Australia, Spain, and the United States find that patients experience higher mortality rates during periods of emergency operating room crowding (Bernstein et al., 2009).

It has been proved that having an effective real-time response in the aftermath of a disaster requires several inter-dependent decisions to be made and numerous coordinated operations to be arranged quickly (Farahani et al., 2020). Patients are supposed to receive surgeries as soon as possible, but their allocation is a complex problem. For example, one ambulance site may send their ambulances to the casualty collection area without the knowledge of other sites' operations and the capabilities of hospitals' admission, etc., which may cause rescue chaos and unbalanced load of resources. While one of the prerequisites for improving rescue efficiency is centralized coordination that can match limited medical resource capacity and urgent surgery demand. In a more efficient situation, the emergency medical resources are arranged in a united system by a centralized decision-maker, e.g., the government, which has the completed information about all the available emergency medical resources' owners (Rachel Lu et al. 2011), with the knowledge of the patients' situation, such as the rescue demand, severity, etc. The knowledge of the patients' injury severity comes from diagnostics/physician assessments at the casualty collection site (Laan et al., 2016).

We consider such a rescue system under the centralized decision-making mechanism after an MCI. Integrating patient assignment and operating room scheduling presuppose centralized coordination that can match limited capacity and urgent demand. First, the patients' information is collected from the casualty collection area to the government, which further gets the information of available medical resources from the hospitals and the ambulance dispatching center under govern. Then, the government makes decisions on ambulance transportation and operating room scheduling in an integrated way. The respective decisions will be transmitted to the corresponding hospitals and ambulance dispatching center, and they will act correspondingly, i.e., transport the patients, and treat them in the hospital following the decisions.

Under the setting of the above-centralized decision-making system, this paper considers two kinds of limited and reusable resources, i.e., ambulances and emergency operating rooms. Reusable resources mean that each ambulance is restored to available status after the previous patient has been transported to the target hospital and can continue to transport the next patient, and each emergency operating room is also available for the next patient after completing the previous patient's surgery. The arrival times of patients at the hospitals and the trauma level will dictate the surgery order of the patients at the hospitals and the time required to treat all patients. We study the collaborative problem of ambulance scheduling and multi-hospital operating room scheduling in MCIs.

Although some models and decision support systems have been proposed for medical resource scheduling in the MCI response problem, there are still no generally accepted rules or principles to guide scheduling personnel on basic problems, such as which hospital ambulance should transport each patient to and how many patients should be transported to each hospital. Repoussis et al. (2016) formulate a mixed-integer model to integrate ambulance dispatching, transportation of patients and surgery order scheduling. It is assumed that one patient is assigned to an ambulance at a time. The impact of capacity-based bottlenecks on ambulances and hospital beds is quantified. Sung and Lee (2016) use a MIP formulation to model the problem as an ambulance routing problem, and the order and destination hospitals for patient evacuation are determined. The number of ambulances is not taken into consideration. Mills et al. (2018) propose two heuristic policies to determine how to allocate ambulances to patient and which hospital should be the destination for those ambulances. Dean and Nair (2014) determine which hospital ambulance should transport each patient to. A SAVE model is introduced to maximize the number of expected survivors from the MCI. While our paper solves the patient assignment problem and sequence of surgeries in hospital under the limitation of ambulances' number and hospital capacity considering ambulance offload delay and patient deterioration.

Given the limited resources of ambulance, hospitals and operating rooms, the aim of our study is to determine the assignment of patients to multiple hospitals and exact surgery start times. The objective is to minimize surgery completion time of all patients. Figure 1 demonstrates the methods for our integrated problem in two stages.

The contributions of this paper are as follows: (1) We explicitly consider the combinatorial optimization problem of patient assignment and patient surgery sequence in MCIs, taking into account ambulance offload delay and the deteriorating condition of patients. Capacity limitation of operating rooms and ambulances are specially considered. An operating room

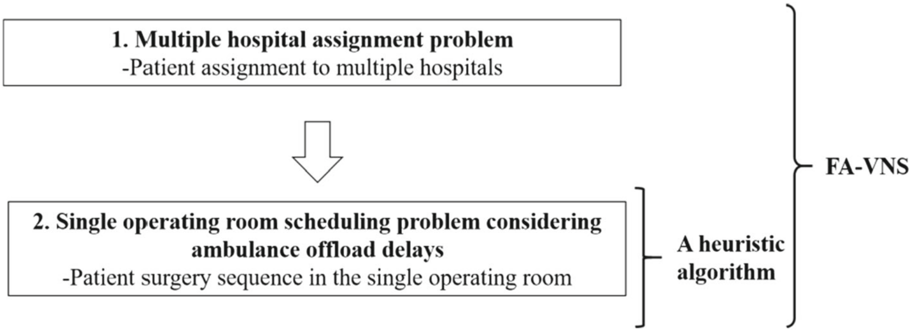

1. Multiple hospital assignment problem
-Patient assignment to multiple hospitals
2. Single operating room scheduling problem considering ambulance offload delays
-Patient surgery sequence in the single operating room
A heuristic algorithm
FA-VNS

Fig. 1 The methods for our integrated problems

would be released after one surgery. (2) Some structural properties of the studied problem are proposed, and a heuristic is developed to solve the single operating room scheduling problem considering ambulance offload delay based on these structural properties. (3) We also develop an effective novel hybrid Firefly Algorithm (FA)-Variable Neighborhood Search (VNS) algorithm incorporating the heuristic to solve our problem.

The remainder of the paper is organized as follows: Sect. 2 presents the literature review of ambulance scheduling and operating room scheduling problems. Section 3 describes our problem by a mixed-integer programming model. Section 4 develops a heuristic algorithm based on some structural properties and scheduling rules to solve the single operating room scheduling problem considering ambulance offload delays. In Sect. 5, a hybrid FA-VNS algorithm incorporating the proposed heuristic is developed to solve the combinatorial problem. Computational experiments are conducted in Sect. 6 to verify the correctness and rationality of the proposed model and evaluate the effectiveness of the proposed algorithm. The conclusions are presented in Sect. 7, and some future research directions are put forward.

2 Literature review

2.1 Ambulance dispatching and offload delay problem

A number of related papers determine transportation order to the hospital according to patient prioritization based on field triage in MCIs. A prevailing protocol of mass casualty triage is START protocol (Fiedrich et al., 2000; Frykberg, 2005; Jenkins et al., 2008; Mills et al., 2013; Sacco et al., 2005). Patients are triaged into four classes (e.g., minor, delayed, immediate, and expectant) by emergency medical technicians at the casualty collection location, and then evacuated following the priority. Patients with higher priority should receive surgeries ahead of those with lower priority, which is the standard practice. Wang et al. (2015) address the single operating room scheduling problem for one day. They consider two priority levels of patients: high-priority patients and low-priority patients. Patients with low priority can only be operated after the operations of high priority have been completed. They determine the surgery sequence in each level of patients by striking a balance between patient satisfaction and operating costs. Mills et al. (2013) present a model of patient triage and design prioritization policies considering multiple casualty locations and multiple receiving hospitals. Nevertheless, Garner (2003) mention that the only recorded incidents for which triage tags were considered useful are small incidents. In large incidents, triage tags were either not used, caused problems, or incidents were managed efficiently because triage tags were not used intentionally. It is noteworthy that little attention has been paid to preventing surges (e.g., avoiding over-triage) by better guiding patient distribution, and how to determine the minimum hospital capacity required to treat all MCI casualties. In this paper, these complex issues are incorporated, and thus we propose an optimal method that can be used to generate and assess emergency preparedness plans.

The transfer of patients from emergency medical service (EMS) providers to emergency department (ED) staff is an important bottleneck (Carter et al., 2014). Ambulance offload delays have recently emerged as one of the most significant challenges for EMS managers, which would increase both risks and surgery costs for patients. When the ambulance delivers the patient to the emergency department, the patient often cannot be admitted in time, resulting in the ambulance staying in the emergency department for a long time. It is called “ambulance offload delay”. Centers for Medicare & Medicaid Services (CMS) stated that it could be reasonable to require the EMS provider to accompany the individual until such time as there where ED resource available to provide care for that individual (Cooney et al., 2011). It takes much time to get ambulances back into service, severely affects the effective turnover of pre-hospital emergency resources, and thus leads to prolonged time on surgery and even compromise safety of casualties. Cooney et al. (2011) point out that it is important to monitor offload delays in evaluating inefficiency of EMS system. While recording this delay presents a serious challenge, most emergency medical services systems only measure the complete time at the hospital (Carter et al., 2014). Until now, there are many research papers that discuss on ambulance offload delay problem in medical services (Almehdawe et al., 2013, 2016; Carter et al., 2014; Cooney et al., 2011, 2013; Li et al., 2019; Majedi, 2008). Cone et al. (2012) found that 12.5% patients experience ambulance offload delay of 30–60 min, and 5% a delay of more than 60 min. In the report of (Crilly et al., 2015), ambulance offload time delay of more than 30 min was experienced by 15% of the 40,783 analyzable ambulance presentations. Carter et al. (2014) study how well the turnaround can act as a proxy for offload delay time, and verify the good correlation between turnaround and actual offload delay time. Although the health and family planning department has organized various hospitals to analyze and solve the problem of ambulance offload delay on many occasions, it has not been fundamentally alleviated because no effective operating mechanism has been established. There is still a lack of formal quantitative models for analyzing ambulance delay problem. Almehdawe et al. (2016) present a queueing network model to investigate the impact of patient routing decisions on ambulance offload delays. In their earlier research (Almehdawe et al., 2013), a queuing model is introduced, in which ambulance utilization is assumed to be not too high and thus travel durations of ambulances are negligible. While in our assumptions, the ambulance utilization is very high under the circumstance of MCIs.

2.2 Deterioration effect

One fundamental feature of trauma care after an MCI is the expectation that severely injured patients will deteriorate over time before receiving surgeries (Dean & Nair, 2014; Hupert et al., 2007; Kamali et al., 2017; L. Lei et al., 2015; Mills et al., 2013; Sung & Lee, 2016). The deteriorating condition of patients inevitably leads to increased care time for the casualties. Dean and Nair (2014) measure the care times by the number of periods for each patient class, and the number is incremental over time up to 7.5 h. Mills et al. (2013) build a fluid model in which patient deteriorate over time according to a survival probability function. The authors design an optimal policy to decrease critical mortality rate. Kamali et al. (2017) solve a optimal triage service order problem under a realistic assumption that patients would deteriorate and have decreasing survival probability. Eun et al. (2019) introduce a MIP model to optimize surgeries assignment considering patient health condition deterioration. They apply a tabu search (TS) to provide effective solutions. Different from the above papers, we consider the deteriorating condition of patients using time-dependent surgery durations. Incorporation of transportation times and deterioration over time makes our problem different from many well-studied problems. The comparison between existing studies and our study is shown in Table 1. We can see that many models and decision support systems have been proposed for medical resource scheduling in the MCI response problem. Most researchers are focused on the location and distribution of emergency response units (Fiedrich et al., 2000) and the supply and distribution of relief materials (Barbarosoğlu & Arda, 2004; Mete & Zabinsky, 2010), while few papers consider the flow of patients between different locations.

Table 1 Key comparisons of the related papers and our study

Papers	ED capacity	Multi-EDs	Ambulance offload delay	Deter-iorate	Ambu-lance	Vehicle assignment	Transp-ortation sequence	Hospital assignment	Patient surgery sequence	Method	Obje-ctive
Sun et al. (2021)	N-R	✓	×	✓	L	✓	✓	✓	×	ε - Con-straints	9,11
Gu et al. (2018)	N-R	✓	×	×	×	×	×	✓	×	Greedy algorithm	4
Kamali et al. (2017)	R	✓	×	✓	×	×	×	×	✓	LP	4
Repoussis et al. (2016)	N-R	✓	×	–	L	✓	✓	✓	×	Heuristic algorithm	1
Almehdawe et al. (2016)	R	✓	✓	–	L	×	×	✓	×	Markov chain	2
Laan et al. (2016)	R	×	✓	–	×	×	×	×	×	Markov chain	8
Sung and Lee (2016)	N-R	✓	×	✓	L	✓	✓	✓	✓	Column genera-tion	4
Leo et al. (2016)	R	✓	×	×	×	×	×	✓	×	MIP	3,10

Table 1 (continued)

Papers	ED capacity	Multi-EDs	Ambulance offload delay	Deter-iorate	Ambu-lance	Vehicle assignment	Transp-ortation sequence	Hospital assignment	Patient surgery sequence	Method	Obje-ctive
Lei et al. (2015)	N-R	✓	×	–	×	×	×	✓	✓	Heuristic algorithm	7
Dean and Nair (2014)	R	✓	×	✓	L	×	×	✓	×	SAVE model	4
Wilson et al. (2013)	N-R	✓	×	✓	×	×	×	×	×	Heuristic algorithm	1,5,9
Almehdawe et al. (2013)	R	✓	✓	–	L	×	×	×	×	Markovian queueing model	3
Ingolfsson et al. (2008)	–	✓	✓	–	L	×	×	×	×	Iterative algorithm	6
Our study	R	✓	✓	✓	L	✓	✓	✓	✓	MIP; TS-ALNS	1

In the above table, R and N-R denote recyclable and non-recyclable. M denotes multiple trips and 1 denotes one trip. Objective: 1-Maximize the makespan; 2- Minimize offload delays; 3-Minimize waiting time;4- Maximize no. of survivors; 5- Minimize no. of fatalities; 6- Maximize system coverage; 7- Minimize the total tardiness; 8- Minimize no. of waiting ambulances; 9-minimize suffering

2.3 Operating room scheduling problem

Operating room scheduling problems have also been widely discussed in the past decades year. Denton et al. (2007) propose a stochastic optimization model for a single operating room daily scheduling problem. They study the simultaneous effects of sequencing surgeries and scheduling start times with the goal of minimizing the weighted sum of surgeon waiting, operating room idling and tardiness. A simple sequencing rule is designed to solve the problem. Y. Sun and Li (2011) present a method to optimize surgery start times for a single operating room with stochastic operation duration. Xiao et al. (2018) propose models and exact combinatorial methods within the context of a single operating room on a single day. (Ito et al., 2018) present a stochastic programming model to minimize the conditional value-at-risk in a single operating room scheduling problem. Wang et al. (2015) solve surgery scheduling problem for single operating room in the single day.

Recently, Wilson et al. (2013) introduce a multi-objective model to decide rescue, surgery and transportation of casualties in major incident response. The model is established on a task scheduling framework, incorporating pre-rescue stabilization, rescue, pre-transportation stabilization and transportation consecutively. Each patient is assigned to a hospital. Variable neighborhood descent heuristic algorithm and a constructive heuristic method are applied. Different from them, the allocation of patients to ambulances is also considered in our paper. Repoussis et al. (2016) model the problem of allocating casualties to hospitals to improve patient outcomes. For each patient, there is a series of tasks, such as transportation, ambulance preparation, patient transfer and hospital service. Each patient is assigned to one of the trips of an ambulance and one of the beds in a hospital. The authors aim to minimize the overall response time and the total flow time for all patients' surgeries. Exact and MIP-based heuristic methods are applied to address the problem. In their assumption, the number of beds in each hospital is fixed and non-recyclable. While in our problem, the operating room is recyclable, which means the operating room is available after the previous patient has completed the surgery. As far as we know, these are the only two work that provide clear task scheduling frameworks from the perspective of individual patient; nevertheless, our approach provides a more comprehensive response to the incidents, including patient surgery sequence in the operating room. In this paper, we combine transport time with operating room scheduling after completing assignment using a hybrid algorithm.

2.4 Heuristic algorithms

Facing the optimization problem, researchers first consider whether it can be solved by some exact algorithms. However, most of the problems in MCIs are proved to be NP-hard (Lee et al., 2013; L. Lei et al., 2015). In addition, solutions to operational problems need to be developed in a very limited time in an emergency. While Repoussis et al. (2016) point that when the number of patients is more than 12, exact algorithms cannot get the optimal solution in a reasonable time (2 h), and thus, the authors propose a TS algorithm to solve the problem. The traditional exact method is usually difficult to compute and its applicability is too limited. Considering the complexity of this class of problems, many researchers solve it by an intelligent algorithm. According to (Zheng et al., 2015), evolutionary algorithms are widely applied in disaster relief operation problems. Fiedrich et al. (2000) present an optimization model for the emergency medical resource allocation problem after earthquake disasters. Casualties are classified according to the level of injury severity and therefore the priorities are determined. The goal function consists of fatalities due to secondary disasters, lack of rescue attempts, duration of the rescue operation, delayed transport and duration of transport. Simulated annealing (SA) and tabu search (TS) are applied to address the problem. (Mills et al., 2018) design two heuristic policies to allocate ambulances for patients and determine destination hospitals for ambulances. Dean and Nair (2014) make decision on which hospital each casualty should be sent to. The authors introduce three implementations in common use: closest-first heuristic, furthest-first heuristic, and cyclical heuristic. Mills et al. (2013) introduce two heuristics for their model of patient triage. L. Lei et al. (2015) develop a rolling horizon heuristic based on mathematical programming to solve the problem of medical teams travelling and medical supplies distribution. Almehdawe et al. (2013) adopt heuristic routing policies to send patient to a specific hospital. Thus, it can be seen that heuristics are widely used in the related papers.

In this paper, a hybrid FA-VNS algorithm is proposed to address the problem. Firefly Algorithm (FA) is a nature-inspired algorithm, first proposed by (Yang, 2010). Since then, FA is developed by many researchers and shows its superiority over some traditional algorithms (Łukasik & Żak, 2009; Yang et al., 2012). Researchers then apply FA in many practical scheduling and optimization problems. FA is based on the flashing patterning of tropical fireflies. The positions of fireflies represent a set of solutions, and fireflies are attracted and move towards the brighter fireflies and thus find the new solutions. Variable neighborhood search (VNS) is a local search meta-heuristic, first proposed by Hansen and Mladenovic in 1997 (Hansen & Mladenović, 2001). The main idea is to systematically change the neighborhood structures in searching for a better solution. In this paper, we develop a novel hybrid FA-VNS algorithm combining the procedures and features of these two meta-heuristic algorithms.

3 Problem description and model

3.1 Problem description

We consider a situation where a large number of trauma patients from an MCI must be quickly transferred to multiple hospitals for surgery. After the disaster happens, emergency medical technicians have categorized patients into four groups according to the START triage protocol in an emergency response time t_0. Patients are all assumed ready for transportation to the hospital by ambulances in the casualty collection area, and all hospitals in the region can accept and provide appropriate quality care for immediate and delayed patients. Hospitals are distributed in different locations, and thus the casualty collection location and the transport time between each hospital is different. The number of available ambulances is assumed to be the number of total operating rooms, which is appropriate as long as the ambulance utilization is very high. The structure of this joint scheduling problem of multi-hospitals is shown in Fig. 2.

Note that patients' health conditions deteriorate over time (Hupert et al., 2007; Xiang & Zhuang, 2016), resulting in longer surgery time in the hospital. The problem is modeled by assuming that the actual surgery duration $p_i = \overline{p_i} + \beta_i t_i$ (Wang et al., 2015), where β_i is the deterioration rate for patient i, and $\overline{p_i}$ denotes the normal surgery duration of patient i. The normal surgery duration $\overline{p_i}$ of the patient in each class follows a known distribution of time based on empirical value.

Most studies on the resource-constrained triage problem focus on delayed patients and immediate patients among the four severity classes (e.g., minor, delayed, immediate, and expectant) in START triage (Dean & Nair, 2014; Jacobson et al., 2012; Mills et al., 2013;

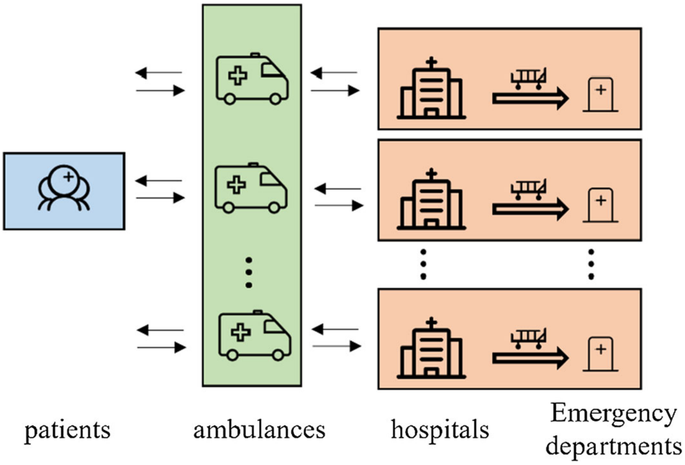


Fig. 2 Structure of joint scheduling problem of ambulances, hospitals, and operating rooms

Sacco et al., 2005; Sung & Lee, 2016). The two classes of patients are in urgent need of surgeries and can endure long-distance transportation. In our problem, we only consider these two classes of patients, i.e., delayed patients and immediate patients. The low-priority patients (delayed patients) could only receive surgeries when surgeries of high-priority patients (immediate patients) have been completed. Let $S_i \in \{1, 2\}$ represents priority level of patient i. In our paper, the priority for immediate patients is set as "$S_i = 2$", higher than the priority for delayed patients "$S_i = 1$".

We assume that no new patients require admission in decision-making point. It is necessary for analyzing the problem since not all patients are ready for surgery at an emergency response time, some casualties may be still trapped at the disaster site and would be rescued and scheduled in the next time section. Considering that information is not timely updated about remaining capacity of EDs in nearby hospitals, and at the same time cannot interrupt the elective surgeries in progress, we limit hospital capacity to only one operating room available in each ED. The assumption is appropriate since under the circumstance of emergency incidents, only limited capacity is available for the current section of the patients.

To avoid frequent ambulance diversion episodes, we assume that each ambulance travels along a fixed route, which means that each ambulance can only transport patients between the casualty collection site and a fixed hospital. Since we focus on single operating room in each hospital, and only one ambulance travels on each fixed route, thus the order of patients to be transported in each specific ambulance is the same as the surgery order in the corresponding hospital. This simplification allows us to gain many insights without making the model too complex.

3.2 Mass casualty patient allocation model

Assuming I is the set of patients who need surgery, and H is the set of hospitals available at the response time t_0 (i.e., decision-making time section). The notations used in the formulation are shown in Table 2.

Table 2 Notations

Sets	
I	Set of all patients that need surgery
H	Set of all available hospitals
$\mathrm{B_h}$	Set of all positions of surgeries in the emergency department of the hospital
Indices	
i	Index of the patient, i = $1, 2, \ldots, n_I$
h	Index of the hospital, h = 1,2,…,n_H
b	Index of the position of the surgery
Parameters	
n_{hb}	Total number of surgeries assigned in the emergency department of the hospital
$\overline{p_i}$	Normal surgery duration of patient i
e_i	Priority level of patient i
β_i	Deterioration rate of surgery duration of patient i
t_0	The emergency response time
t_h	Round-trip transport time between the casualty collection site and the hospital h
M	A sufficiently large positive number
Decision variables	
$x_{ihb} = 1$	If patient i is assigned to position b in the hospital h, 0 otherwise
Auxiliary variables	
e_i	The surgery complete time of patient i
l_i	Round-trip transport time for patient i
p_i	Actual surgery duration of patient i
s_i	The surgery start time of patient i
as_{hb}	Mapping of surgery start time s_i assigned at hospital h at position b
a_i	Arrival time of patient i at the hospital
aa_{hb}	Mapping of arrival time a_i assigned at hospital h at position b
c_i	The surgery complete time of patient i
C_{max}	The maximum of the complete time of all patients' surgeries

Objective function: minimize the makespan

$$Min\ C_{\max} \tag{1}$$

Subject to:

$$\sum_{h \in H, b \in H_b} x_{ihb} \leq 1 \quad \forall i \in I \tag{2}$$

$$\sum_{i \in I} x_{ihb} = 1 \quad \forall h \in H, b \in B_h \tag{3}$$

$$x_{ihb} = 1 \quad l_i = T_h \quad \forall i \in I, b \in B_h \tag{4}$$

$$\sum_{i \in I} x_{ih(b-1)} \geq \sum_{i \in I} x_{ihb} \quad \forall h \in H, b \in B_h \backslash \{1\} \tag{5}$$

$$s_i \geq l_i \quad \forall i \in I \tag{6}$$

$$p_i = \overline{p_i} + \beta_i * s_i \quad \forall i \in I \tag{7}$$

$$x_{ih0} = 1 \Rightarrow a_i = T_h \quad \forall h \in H \tag{8}$$

$$x_{ihb} = 1 \Rightarrow a_i = aa_{hb} \quad \forall i \in I, h \in H, b \in B_h \tag{9}$$

$$x_{ihb} = 1 \Rightarrow s_i = as_{hb} \quad \forall i \in I, h \in H, b \in B_h \tag{10}$$

$$c_i = s_i + p_i \quad \forall i \in I \tag{11}$$

$$as_{hb} \geq as_{h(b-1)} + \sum_{i \in I} p_i x_{ih(b-1)} \quad \forall h \in H, b \in B_h \backslash \{1\} \tag{12}$$

$$as_{hb} \geq as_{h(b-1)} + \sum_{i \in I} l_i x_{ih(b-1)}, \quad \forall h \in H, b \in B_h \backslash \{1\} \tag{13}$$

$$s_i \geq a_i \quad \forall i \in I \tag{14}$$

$$aa_{hb} \geq as_{h(b-1)} + T_h \quad \forall h \in H, b \in B_h \backslash \{1\} \tag{15}$$

$$a_i \geq l_i \quad \forall i \in I \tag{16}$$

$$s_{i'} \geq s_i \quad \forall i, i' \in I, e_i \geq e_{i'} \tag{17}$$

$$C_{\max} \geq c_i \quad \forall i \in I \tag{18}$$

Constraint (2) ensures that each patient has been assigned exactly to one position in one hospital. Constraint (3) ensures that one position in one hospital can only be assigned to one patient. Constraint (4) defines the round-trip time for each patient. Constraint (5) limits that there must be a patient at position *b − 1* if there is another patient assigned at position *b* in the same emergency operating room. Constraint (6) defines the start time of the first operation. Constraint (7) defines the actual surgery duration. Constraint (8) initializes the arrival time for the first patient to the hospital. Constraint (9) associates the auxiliary variables a_i and aa_{hb}. Constraint (10) associates the auxiliary variables s_i and as_{hb}. Constraint (11) defines the surgery complete time. Constraint (12) updates the surgery start time for each position in the emergency operating room. Constraint (13) updates the surgery start time at each position of the emergency operating room according to the surgery duration of the patient assigned to the same operating room at the previous position. Constraint (14–16) impose the lower bounds of the surgery start time. Constraint (17) ensure that the triage protocol is accepted by the patients. Constraint (18) determines the makespan, which is equal to the maximum of the surgery completion time of all patients.

4 Structural properties for single operating room sequence problem

In this paper, a hybrid FA-VNS algorithm is proposed (see Sect. 5) to assign patients to multiple hospitals. Based on the assignment results, we give an example of different arrangements in the single operating room (see Sect. 4.1) considering the ambulance offload delay. We also propose two lemmas in Sect. 4.2, based on which we design Algorithm 2 incorporated in the hybrid FA-VNS algorithm to determine the surgery sequence in each operating room and thus determine the transportation order in each ambulance.

4.1 An example of different arrangements in the single operating room

We can see that the surgery start time of the later patient depends on the round-trip transportation time T_h, and the actual surgery duration p_i ($p_i = \overline{p_i} + \beta_i t_i$) of patient who receives surgery earlier in the same operating room. Figure 3 gives an example of different schemes of patients' surgeries in the operating room.

In Fig. 3, we can see that three patients $\{I_1, I_2$, and $I_3\}$ have been assigned to the same operating room. We give four cases to show different arrangements in the operating room. The blank squares denote the duration of each patient's surgery, the arrows denote the round-trip transportation time between the casualty collection location and hospital location ($p_1 < T_1 < p_2 < T_2 < p_3$), and the shadow squares denote the idle time of the operating room. Operating room's idle time should be minimized to maximize the utilization of the operating room. The idle time depends on the difference between transportation time and the surgery duration. Round-trip transportation time depends on the hospital location. If there is a gap between the round-trip transportation time and surgery duration, the idle time is in-advisably prolonged.

In Case 1, we can see that since the surgery duration of patient I_1 is shorter than the round-trip transportation time T_1, patient I_2 receives the surgery after an idle duration in the operating room. In Case 2, we can see that since the surgery duration of patient I_3 is longer than the round-trip transportation time T_1, patient I_2 can receive the surgery without any idle time in the operating room. Also, patient I_1 can receive the surgery without any idle time in the operating room after patient I_2. Thus, the makespan in Case 2 is shorter than that in Case 1. While in Case 3, since the round-trip transportation time T_2 is very long, the surgery duration of patient I_1 and I_2 is shorter than T_2. Thus, there exits idle time before I_1 receives surgery, and so does I_2. While in Case 4, since the surgery duration of patient I_3 is longer than the round-trip transportation time T_2, there is no idle time before patient I_2 receives surgery. Thus, the makespan in Case 4 is shorter than that in Case 3. From the analysis above, we can see that Case 2 is better than Case 4, Case 1, and Case 3. In this case, the idle time for the operating room is perfectly minimized. Therefore, based on the results of different arrangements for surgeries of patients, we can sum up a set of dispatching rules for minimizing the idle time of the operating room under different situations of round-trip transportation time.

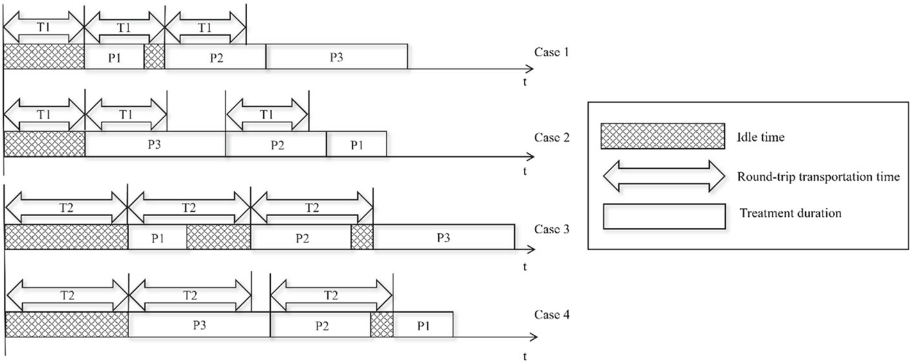


Fig. 3 An example of different arrangements for patients in the single operating room

4.2 Structural properties for single operating room scheduling problem

To solve the single operating room scheduling problem considering ambulance offload delay, we give some properties of the makespan minimization problem for any given patient assignment list in a specific operating room.

Lemma 1 For the patients with the same priority ($\upbeta_1 = \upbeta_2 = \ldots \upbeta_i = \upbeta_j = \cdots = \upbeta$) in the same operating room, if at time t, actual surgery durations of all remaining patients to be scheduled are not smaller than T_h (i.e., $p_i(\mathrm{t}) \geq T_h$), they are ordered according to the smallest normal surgery duration first rule (SNSDF): $\overline{p_1} \leq \overline{p_2} \leq \ldots \leq \overline{p_n}$.

Proof We prove the lemma by contradiction. If at $t = 0$, all patients' surgery durations are not smaller than T_h (i.e., $p_i \geq T_h$). First we show that patients with the same priority are sequenced according to the SNSDF as a schedule $\pi^{'}$, where patient I_j is followed by I_i ($\overline{p_i} > \overline{p_j}$). At the same time, consider an optimal schedule π^* of the OR where patients do not follow the SNSDF, where patient I_i is followed by I_j ($\overline{p_i} > \overline{p_j}$), leaving the remaining patients in their original positions of the sequence. We assume that the start time for I_i in schedule π^* is t.

For schedule π^*, $C_i(\pi^*) = t + p_i = t + (\overline{p_i} + \upbeta\mathrm{t}) = (1 + \upbeta)t + \overline{p_i}$, and then we have $C_j(\pi^*) = C_i + p_j = C_i + (\overline{p_j} + \upbeta C_i) = \overline{p_j} + (1 + \upbeta)C_i = \overline{p_j} + (1 + \upbeta)[(1 + \upbeta)\mathrm{t} + \overline{p_i}] = (1 + \upbeta)^2\mathrm{t} + (1 + \upbeta)\overline{p_i} + \overline{p_j}$.

Similarly, for schedule $\pi^{'}$, we can easily obtain $C_j(\pi^{'}) = t + p_j = t + (\overline{p_j} + \upbeta\mathrm{t}) = (1 + \upbeta)\mathrm{t} + \overline{p_j}$, $C_i(\pi^{'}) = (1 + \upbeta)^2\mathrm{t} + (1 + \upbeta)\overline{p_j} + \overline{p_i}$.

Thus we have $C_j(\pi^*) - C_i(\pi^{'}) = \upbeta(\overline{p_i} - \overline{p_j}) > 0$. It implies that patient operated after I_i and I_j under π^* has a later start time than that under $\pi^{'}$. Thus, the makespan of patients under π^* is strictly greater than that under $\pi^{'}$, which conflicts with our assumption. It should be $\overline{p_i} < \overline{p_j}$. The proof is completed.

Lemma 2 For the patients with the same priority ($\upbeta_1 = \upbeta_2 = \cdots = \upbeta$), if at time t, there is any patient to be scheduled whose actual surgery duration is smaller than T_h (i.e., $p_i(\mathrm{t}) \geq T_h$), an optimized solution exists under different circumstances as shown in Table 3.

The proof is presented in the Appendix.

Table 3 Summary of the sequence rules in Lemma 2

	Cases		Conclusions
$T_h > \mathrm{p_A}$	$\frac{\upbeta S_A+\overline{p_j}}{1-\upbeta} < T_h$	(1)	π^* is better than π'
	$\frac{\left(\upbeta^2+\upbeta\right)S_A+(1+\upbeta)\overline{p_i}}{1-\upbeta-\upbeta^2} < T_h < \frac{\left(\upbeta^2+\upbeta\right)S_A+(1+\upbeta)\overline{p_i}+\overline{p_j}}{1-\upbeta-\upbeta^2}$	(2)	
	Otherwise		π' is better than π^*
$T_h < \mathrm{p_A}$	$\upbeta C_A + \overline{p_i} < T_h < p_A$	(3)	π^* is better than π'
	$\left(\upbeta^2 + \upbeta\right)C_A + \upbeta\overline{p_i} + \overline{p_j} < T_h < \left(\upbeta^2 + \upbeta\right)C_A + \upbeta\overline{p_j} + \overline{p_i}$	(4)	
	Otherwise		π' is better than π^*

Based on Lemma 2, the following Algorithm 1 is designed to solve patient sequence problem in a single operating room for patients with the same priority.

Algorithm 1: Pseudo code of sub function *sub_f*

```
Input: a patient set $\pi_0=\{I_1, I_2, \ldots, I_n\}$ and $T_h$
Output: surgery sequence $\pi$, and the complete time of the patient set $c_\pi$
$\pi = []$, small_duration _list = []
$s_i \leftarrow$ start time of surgery of patient $i$
$I_{r1} \leftarrow$the patient with the shortest $\overline{p_\iota}$ in higher_priority_patient_list
 if $\overline{p_{r1}} \geq T_h$ then
    Add patients in higher_priority_patient_list sorted by the non-decreasing order of
    $\overline{p_\iota}$ into $\pi$
 else
    for i in higher_priority_patient_list do
        while $\overline{p_\iota} \leq T_h$ do
            Add $P_i$ into small_duration _list
        end while
     end for
    $I_{r2} \leftarrow$ the patient with the longest $\overline{p_\iota}$ in small_duration _list
    $\pi.append(I_{r2})$
    $\pi[-1].actual\_duration = \overline{p_{\pi[-1]}} + \beta_{\pi[-1]} \cdot s_{\pi[-1]}$
    for i in higher_priority_patient_list do
        for j in higher_priority_patient_list do
            if $T_h \geq \pi[-1].actual\_duration$
                if formula (1) or formula (2) then
                    $\pi.append(I_i)$
                else
                    $\pi.append(I_j)$
                end if
            else
                if formula (3) or formula (4) then
                    $\pi.append(I_i)$
                else
                    $\pi.append(I_j)$
                end if
            end if
        end for
    end for
end if
```

In this section, based on the above two lemmas for the single operating room scheduling problem, Algorithm 2 is proposed to determine the surgery sequence in each operating room, given that all patients with different priorities have been assigned to the destination hospitals. The framework of Algorithm 2 is as follows.

```
Algorithm 2:
Input: patient set $\pi_0$={$I_1, I_2, ..., I_n$} and $T_h$
Output: surgery sequence $\pi$, and the complete time of the patient set $c_\pi$
1    sub_f denotes the function in Algorithm 2
2    for $i = 1$ to $n$ do
3        if $J_i == 2$ then
4            Add $I_i$ into higher_priority_patient_list
5        else if $J_i == 1$ then
6            Add $I_i$ into lower_priority_patient_list
7        end if
8    end for
9    if higher_priority_patient_list is not $\emptyset$ then
10       $\pi_1$, $c_{\pi1}$ = sub_f(higher_priority_patient_list, $T_h$)
11       $\pi_2$, $c_\pi$ = sub_f(lower_priority_patient_list, $T_h$)
12       $\pi = \pi_1.append(\pi_2)$
13   else
14       $\pi$, $c_\pi$ = sub_f($\pi_0$, $T_h$)
15   end if
```

The time complexity of step 5 is O(n log n). The time complexity of step14-30 is $O(n^2)$, and the time complexity of the other steps is no more than $O(n^2)$. Thus, the total time complexity of Algorithm 1 is $O(n^2)$. The time complexity of Algorithm 2 is $O(n^2)$.

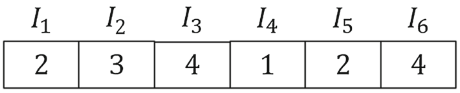
5 Metaheuristic-based hybrid approach

To solve our problem, we propose a hybrid Firefly Algorithm (FA)-Variable Neighborhood Search (VNS) algorithm incorporating the heuristic in this section. The critical procedures of the hybrid FA-VNS are introduced in Sects. 5.2, 5.3, 5.4.

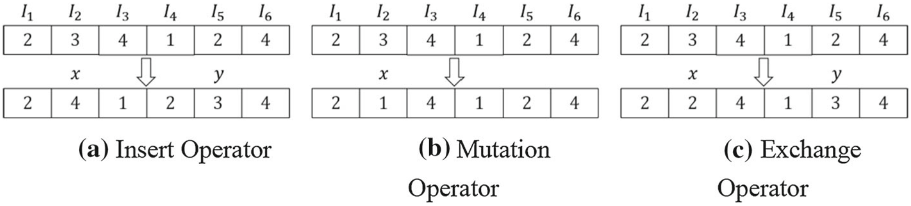
5.1 Coding scheme

In our algorithm, a solution for the problem of assigning patients to the hospital is an array, of which the length is equal to the number of patients. Each position value stands for the index of the hospital that the patient is assigned to. We use an instance of six patients {I_1, I_2, I_3, I_4,I_5, I_6} and four hospitals {H_1, H_2, H_3,H_4}. An instance solution is showed in Fig. 4, and X = {2, 3, 4, 1, 2, 4} where the patients {I_4}, {I_1, I_5}, {I_2}, {I_3, I_6} are assigned to hospital H_1, H_2, H_3, and H_4 respectively.

Fig. 4 Illustration of a particle's solution



I_1	I_2	I_3	I_4	I_5	I_6
2	3	4	1	2	4


I_1 I_2 I_3 I_4 I_5 I_6
2 3 4 1 2 4
x
y
2 4 1 2 3 4
I_1 I_2 I_3 I_4 I_5 I_6
2 3 4 1 2 4
x
2 1 4 1 2 4
I_1 I_2 I_3 I_4 I_5 I_6
2 3 4 1 2 4
x
y
2 2 4 1 3 4

(a) Insert Operator **(b)** Mutation Operator **(c)** Exchange Operator

Fig. 5 Local search operators

5.2 Encoding correction

In the iterative processes, infeasible solutions may be generated under the following circumstances: (1) Hospital slots should be encoded with integers while searching operators may generate decimals. We adopt the coding correction strategy which takes integer approximate values. (2) All position values should be in the range of $[1, n_H]$. Set the numbers in X that are less than "1" to "1" and those greater than "n_H" to "n_H". The initial feasible solution is obtained from the heuristic algorithm designed in Sect. 3 or randomly selected from a feasible solution set.

5.3 Neighborhood structures

In the following, three neighborhood structures are designed (see Fig. 5) and applied in a VNS-based local search procedure for improving the effectiveness of the traditional FA. These neighborhood structures are applied in order, and the local search process needs to be repeated until it finds a good set of base locations. $N_k(X)$ denotes the k-th neighborhood of solution X.

(1) **Insert Operator:** In solution X_i, randomly select two positions x and y. The x position value is taken from its current position and inserted after position y (see Fig. 5a).

(2) **Mutation Operator:** In solution X_i, randomly select a position x. Generate an integer number in $[1, Q]$ and make it the substitute for the value of the position x, and thus, a neighbor of X_i can be obtained (see Fig. 5b).

(3) **Exchange Operator:** In solution X_i, randomly select position x and position y. A neighbor of X_i can be obtained by swapping the numerical values in position x and position y (see Fig. 5c).

The detail of VNS-Based Local Search operation is designed as follows:

VNS-based local search operation:	
Step 1	Define neighborhood structures N_s(s $= 1, \ldots, s_{max}$)
Step 2	Get initial solution X_{new} which is produced by FA
Step 3	Execute the s-th Local Search for X_{new} to obtain a solution X'_{new}
Step 4	If solution X'_{new} is better than X_{new}, then set$X_{new} = X'_{new}$, s $=$ 1 and go to step 5
Step 5	If $s < s_{max}$, then go to step 3; else, stop the iteration

5.4 Framework of the hybrid algorithm

The whole procedures of the hybrid FA-VNS to assign the patients to ambulances and hospitals and determine surgery sequence in each hospital based on the detailed description of our proposed algorithm are shown in Table 4 and Fig. 6.

In the iterations, the time complexity of decoding steps is no more than $O(n^2)$, because the Algorithm 2 is incorporated. The time complexity of step 8 to step 29 is $O(n^2)$. The time complexity of other steps is no more than O(n). Thus, the total time complexity of FA-VNS is $O(n^2)$.

6 Computational experiments and comparison

We perform computational experiments to check the effectiveness and efficiency of our proposed methods. For this purpose, we have randomly generated small- and large-scale of instances based on realistic data. All experiments are conducted on a laptop with an Intel(R) Core (TM)2 Duo CPU @2.93 GHz and 8 GB RAM. Our methods are implemented by Python 2.7. Gurobi 9.0.2 (win64) is used as the MIP solver. 20 runs are performed for each case, and all computational times are recorded in the unit of second.

This section is organized as follows: In Sect. 6.1, we describe the data sets and the experimental parameter setting. In Sect. 6.2, the MIP model given in Sect. 3 is validated by Gurobi and the applicability of our proposed algorithm is examined. In Sect. 6.3, we compare our proposed algorithm with other three widely used algorithms. Section 6.4 is sensitivity analysis.

6.1 Experimental settings and data sets

We consider 24 possible cases reflecting different scales of the mass-casualty incidents. Referring to the actual situation of the hospital and other literature on surgical scheduling (Repoussis et al., 2016), the following experimental data were generated: the number of hospitals is $n_H = \{4, 5, 6, 7\}$, and the number of patients is $n_I = \{16, 18, 20, 30, 40, 50\}$. The proportion of immediate patients is 20–50% and the baseline is 35%. The distance from the disaster site is expressed in terms of round-trip time in minutes (see Table 2). We consider

Table 4 Procedures of FA-VNS algorithm

```
FA-VNS algorithm
Input:  Objective function $f()$;
        Population size popsize;
        Number of dimension dim;
        Number of iterations max_iter
Output: The optimized solution X* and the best objective function value $f$(X*)
Initialize a population. POP = $\{X^1, X^2, \ldots, X^k, \ldots, X^{popsize}\}$.
Encoding correction
Apply the Algorithm 2 to get the patient sequence list.
Calculate the fitness of each firefly and find the best one $X_{best}$
Set $iter = 0$
while $(iter < max\_iter)$ do
    $iter = iter + 1$
    for $i = 1 : popsize$ do
        $temp = X_{best}$
        for $j = 1 : popsize$ do
            if $f(X^i) \geq f(X^j)$ then
                Calculate Cartesian distance $r_{ij} = \|X^i - X^j\| = \sqrt{\sum_{k=1}^{dim}(x_k^i - x_k^j)^2}$
                for $d = 1 : dim$ do
                    $x_d^j = x_d^j + \beta_0 e^{-\gamma r_{ij}^2}(x_d^i - x_d^j)$ +α(rand – 0.5)
                end for
              Encoding correction
            end if
          Calculate the fitness of each firefly
        end for
        Execute VNS-based local search procedure for $X^i$
        Encoding correction
        Apply the Algorithm 2 to get the patient sequence results
        Calculate the fitness of each firefly
        Sequence all the fireflies according to the fitness and find the best one $X_{best}'$
        if $f(X_{best}) \geq f(X_{best}')$ then
            $X_{best} = X_{best}'$
        else
            $X_{best} = temp$
        end if
    end for
    Sequence all the fireflies according to the fitness and find the best one $X_{best}'$
    if $f(X_{best}) \geq f(X_{best}')$ then
        $X_{best} = X_{best}'$
    else
        $X_{best} = temp$
    end if
end while
Sequence all the fireflies by the fitness and find the best one $X_{best}$
$X^* = X_{best}$
Return $X^*$ and $f(X^*)$
```

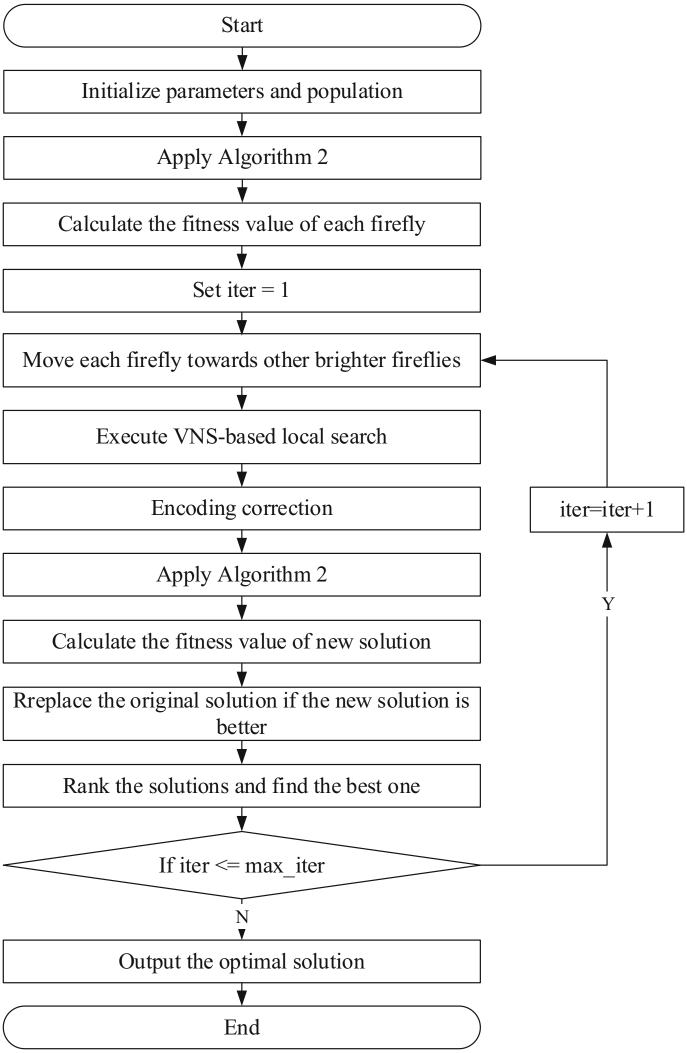

Start
Initialize parameters and population
Apply Algorithm 2
Calculate the fitness value of each firefly
Set iter = 1
Move each firefly towards other brighter fireflies
Execute VNS-based local search
Encoding correction
Apply Algorithm 2
Calculate the fitness value of new solution
Rreplace the original solution if the new solution is better
Rank the solutions and find the best one
If iter <= max_iter
Y
iter=iter+1
N
Output the optimal solution
End

Fig. 6 The flow chart of the proposed FA-VNS algorithm

two hospitals that are included in regional disaster preparedness planning: a "local" hospital (5 min for a round-trip transport) and a "remote" one (20 min for a round-trip transport), and each has one available emergency operating room. One of the characteristics of our model is that it considers multiple hospitals in different geographical locations when making decisions. It should be noted that although only two types of hospitals are considered for illustrative purposes, our method is capable of handling more. The normal surgery times are normally distributed with a mean dependent on the severity class based on empirical value (Marques et al., 2012).

For immediate patients, $\mu_1 = 40$ and $\sigma_1{}^2 = 18^2$. For delayed patients, $\mu_2 = 20$ and $\sigma_2{}^2 = 13^2$. The deteriorating rate for patient surgery duration is set as $\beta_i = 0.01$ for delayed patients and $\beta_i = 0.05$ for immediate patients through the survey in (Wang et al., 2015). Table 5 shows

Table 5 Parameter setting for the experiments

Notation	Parameter	Value
n_I	Number of patients	16, 18, 20, 30, 40, 50
ω	Proportion of immediate patients	20–60% (baseline: 35%)
n_H	Number of hospitals	4, 5, 6, 7
T_h	Distance (local hospital)	Uniform distribution U (2,9)
	Distance (remote hospital)	Uniform distribution U (10,15)
$\overline{p_i}$	Normal surgery duration (immediate patients)	Normal distribution N (40, 18^2)
	Normal surgery duration (delayed patients)	Normal distribution N (28, 13^2)
β_i	deteriorating rate (immediate patients)	0.05
	deteriorating rate (delayed patients)	0.01

the different levels considered.

As mentioned in Sect. 1, there are no specific national or regional criteria for selection a patient's destination hospital, and the results may vary with the location and incident. In the absence of any standard strategy to compare our algorithms, we compare our FA-VNS with three popular metaheuristic algorithms: FA (Marichelvam et al., 2013), VNS (D. Lei & Guo, 2016), and PSO (Taherkhani & Safabakhsh, 2016).

In our proposed FA-VNS algorithm, the parameters that may affect its performance include β_0, γ, and , the number of populations. According to the survey and a series of preliminary experiments, the parameter values are set as follows: β_0=1, $\gamma = 1$, $= 2$, *popsize* $= 20$.

6.2 FA-VNS VS Gurobi

In this section, the MIP model shown in Sect. 3.1 is solved by Gurobi with the time limitation of 1800s and is compared with our proposed algorithm. 17 randomly generated cases are tested, and the number of patients varies from 4 to 36.

Table 6 shows the results obtained by Gurobi and FA-VNS. Columns n_I and n_H represent the number of patients and hospitals, respectively. Columns Obj, GAP, and Runtime report the objective function value, GAP value, and run time obtained by Gurobi. Each case is run 20 times by FA-VNS. Columns Best, Avg, Worst, SD and Runtime report the best, average, worst objective function value, the standard deviation (SD) among the 20 times, and the run time by FA-VNS. The last column Impr. shows the improvement by the FA-VNS over Gurobi in terms of the solution's objective value. The calculation formula of Impr. is as follows:

$$Impr.(\%) = \frac{BestObj(FA - VNS) - Obj(Gurobi)}{Obj(Gurobi)} * 100\%$$

As can be seen from Table 6, for most instances with the number of patients less than 12, the quality of the function values obtained by Gurobi is better than the best one obtained by FA-VNS, and Gurobi obtains the optimal solution in a shorter time than FA-VNS. When the number of patients is greater than or equal to 12, FA-VNS reports better results than Gurobi. In addition, FA-VNS takes much less time than Gurobi. The experiment verifies the correctness of the model in Sect. 3 and quality of the results obtained by our method. Meanwhile, the correctness of Algorithm 1 and Algorithm 2 designed in FA-VNS is also verified.

Table 6 Experimental results of Gurobi and TS-ALNS

n_I	n_H	Gurobi			FA-VNS					Impr.(%) over Gurobi
		Obj	GAP (%)	Runtime (sec)	Best	Avg	Worst	SD	Runtime (sec)	
4	2	44.50	0.00	0.02	44.50	44.50	44.50	0.00	1.17	0.00
6	2	61.60	0.00	0.17	62.25	62.25	62.25	0.00	5.87	− 1.06
8	2	79.69	0.00	0.24	80.36	80.36	80.36	0.00	6.17	− 0.84
10	2	100.67	0.00	6.19	100.67	100.67	100.67	0.00	6.29	0.00
12	2	–	–	1800.00	123.11	123.11	123.11	0.00	9.73	–
14	2	–	–	1800.00	143.00	143.00	143.00	0.00	11.24	–
16	2	–	–	1800.00	170.255	170.255	170.255	0.00	17.85	–
18	2	–	–	1800.00	190.57	190.57	190.57	0.00	21.99	–
20	2	–	–	1800.00	316.79	316.92	317.18	0.02	31.31	–
22	2	–	–	1800.00	339.48	339.59	339.77	0.02	34.02	–
24	2	–	–	1800.00	251.89	251.89	251.89	0.00	22.37	–
26	2	–	–	1800.00	276.23	276.23	276.23	0.00	24.23	–
28	2	–	–	1800.00	454.79	455.15	455.53	0.06	23.10	–
30	2	–	–	1800.00	450.96	451.08	451.25	0.01	23.21	–
32	2	–	–	1800.00	591.39	591.47	591.54	0.00	27.73	–
34	2	–	–	1800.00	593.16	593.57	594.08	0.14	93.59	–
36	2	–	–	1800.00	639.30	639.79	640.16	0.11	64.02	–

‘–’ represents Gurobi cannot find a feasible solution in 1800s

6.3 FA-VNS VS FA, VNS and PSO

The comparison of FA-VNS, FA, VNS and PSO are conducted in this section. The parameters are introduced in Sect. 6.1. We examine 24 cases with up to 50 patients for large-scale instances. The average objective value (Ave) and the minimization objective value (Best) are measured over 24 cases in Table 7. We also analyze and compare the performance of these four algorithms by Relative Percent Deviation (RPD) (Vallada and Ruiz 2011) defined as follows:

$$RPD(M) = \frac{Max\ Obj(F) - Ave(M)}{Max\ Obj(F)} * 100\%$$

where M denotes each algorithm, and $Ave(M)$ denotes the average fitness value for each algorithm. $Max\ Obj(F)$ denotes the best fitness value we have gotten.

In order to ensure the algorithms can converge to a good solution, the number of populations is set as 20, and the maximum of iteration is 200. The average fitness value (Ave) and the best fitness value (Min) obtained from 1 to 200 iterations are reported to analyze the performance of each algorithm in Table 7. Also, Relative Percent Deviation (RPD) (Vallada and Ruiz 2011) is calculated to evaluate and compare the performance of these four methods, defined as:

$$RPD = \frac{Method(sol) - Best(sol)}{Best(sol)} \times 100\%$$

where $Method(sol)$ is the average fitness value obtained by and $Best(sol)$ is the lowest makespan obtained for that instance among the four algorithms. As the objective is minimizing the maximum makespan, the smaller the RPD, the better the performance. In order to ensure the reliability of experiments, each instance is run for 20 times. The initial solutions of each instance are the same for the algorithms to ensure that each algorithm starts at the same level to search for the optimized solutions. In Table 7, the last two rows show the best and average RPD (ARPD) values of instances 1–12 and instances 13–24 for each algorithm.

According to the experimental result for case 1 ($n_I = 16$, $n_H = 4$) in Table 7, the obtained schedule scheme for patients (I1–I16) who are assigned among ambulances (A1–A4) and hospitals (H1–H4) is provided in Fig. 7. The length of T_i and P_i represent round -trip transportation time and surgery duration for each patient, respectively. For example, for A1 (i.e., Ambulance 1) and H1 (i.e., Hospital 1), the order of patients is P5, P6, P12, and P7. Following the scheme, the ambulance drivers and the doctors in the emergency department can prepare the relevant emergency medical resources and service the patients in an orderly manner.

From the values in bold, we can see that VNS can obtain as good objective values as FA-VNS in some cases. While in most cases, FA-VNS has the best performance in obtaining average makespan, lowest makespan, best RPD and average RPD compared with other three algorithms. In order to further verify the statistical validity of RPD values and find out the best algorithm, we design a series of experiments and variance analysis, in which we consider a different algorithm as a factor and set the response variable as the average RPD value.

By SPSS, the 20 results generated from each algorithm are analyzed with paired-samples t-test for all the 24 instances, shown in Table 8. Statistical significance is set at an alpha of 0.05, and a ρ-value of < 0.05 is deemed statistically significant. Compared with VNS, FA-VNS generates remarkably better results for 18 out of 24 instances and is competitive for the remaining instances where there is no statistical difference between the two algorithms.

Table 7 Comparative results for the four algorithms

No	n_I	n_H	FA-VNS			FA			VNS			PSO		
			Ave	Best	RPD	Ave	Best	RPD	Ave	Best	RPD	Ave	Best	RPD
1	16	4	90.4709	**90.0000**	0.0052	116.7903	104.9540	0.2977	93.3359	**90.0000**	0.0371	120.9187	112.7063	0.3435
2	18	4	105.6270	**103.1167**	0.0243	137.3395	120.0000	0.3319	108.8056	106.5395	0.0552	140.4969	120.0000	0.3625
3	20	4	137.0790	**136.8243**	0.0019	167.7626	158.9152	0.2261	137.7856	**136.8243**	0.0070	169.4415	154.3419	0.2384
4	30	4	146.6081	**136.2348**	0.0761	222.9673	196.3532	0.6366	155.6179	147.7587	0.1423	217.6785	187.8553	0.5978
5	40	4	211.2429	**193.0462**	0.0943	429.7455	358.8132	1.2261	249.2118	224.9103	0.2909	402.5263	359.4495	1.0851
6	50	4	397.5813	**343.0098**	0.1591	863.6234	784.4524	1.5178	451.9787	405.8186	0.3177	738.6753	613.2277	1.1535
7	16	5	88.6563	**77.7794**	0.1398	115.9822	101.5604	0.4912	88.7785	84.1987	0.1414	119.9598	109.9441	0.5423
8	18	5	87.8504	**77.1828**	0.1382	119.9664	107.7485	0.5543	88.5887	85.4522	0.1478	124.8747	99.5416	0.6179
9	20	5	105.5609	**96.5419**	0.0934	145.5653	130.6331	0.5078	107.7280	102.5660	0.1159	155.2704	143.4851	0.6083
10	30	5	138.0280	**129.7103**	0.0641	241.4678	222.0356	0.8616	141.6214	136.2043	0.0918	229.0760	218.6783	0.7661
11	40	5	185.2118	**166.4102**	0.1130	432.6465	366.2872	1.5999	207.9505	195.0000	0.2496	428.9231	396.6467	1.5775
12	50	5	282.1811	**259.5622**	0.0871	805.1693	674.8823	2.1020	330.2890	307.2686	0.2725	747.7489	578.2376	1.8808
13	16	6	64.5265	**63.1037**	0.0225	96.5458	85.4585	0.5300	64.9358	**63.1037**	0.0290	100.4858	89.5458	0.5924
14	18	6	70.7798	**67.5000**	0.0486	111.2442	96.4831	0.6481	76.5266	69.6668	0.1337	116.0950	102.1950	0.7199
15	20	6	83.7335	**77.8030**	0.0762	119.1194	109.6681	0.5310	84.5193	**77.8030**	0.0863	125.1675	117.8284	0.6088
16	30	6	104.1219	**94.8064**	0.0983	193.4240	173.1429	1.0402	112.0195	103.3905	0.1816	198.2637	189.8789	1.0912
17	40	6	147.9852	**137.2220**	0.0784	359.6649	313.4325	1.6210	167.6370	160.6567	0.2216	360.2306	308.9200	1.6252
18	50	6	197.8810	**178.7238**	0.1072	560.5702	503.3437	2.1365	221.9488	209.0437	0.2419	551.6478	497.4566	2.0866
19	16	7	60.1318	**59.2859**	0.0143	90.3454	85.1702	0.5239	60.1547	**59.2859**	0.0147	94.6618	85.9450	0.5967
20	18	7	82.4274	**78.6161**	0.0485	140.9181	132.9181	0.7925	86.8986	82.4428	0.1054	146.2983	137.6315	0.8609

Table 7 (continued)

No	n_I	n_H	FA-VNS			FA			VNS			PSO		
			Ave	Best	RPD	Ave	Best	RPD	Ave	Best	RPD	Ave	Best	RPD
21	20	7	78.8166	**73.7789**	0.0683	125.8356	117.1526	0.7056	81.4711	79.4841	0.1043	133.1775	126.4025	0.8051
22	30	7	103.7780	**99.2563**	0.0456	224.6025	200.2710	1.2629	113.9653	105.9285	0.1482	242.2248	206.2796	1.4404
23	40	7	145.8699	**139.2009**	0.0479	355.6207	335.5079	1.5547	159.7128	148.2856	0.1474	356.6792	356.6792	1.5623
24	50	7	169.9914	**156.7389**	0.0846	511.6239	473.2111	2.2642	186.4880	163.6786	0.1898	520.1581	417.7155	2.3186
(1–12) ARPD				0.0019(0.0831)			0.2261(0.8627)				0.0070(0.1558)		0.2384(0.8145)	
(13–24) ARPD				0.0143(0.0617)			0.5239(1.1342)			0.0147(0.1336)			0.5924(1.1923)	

The bold font is to emphasize the best results

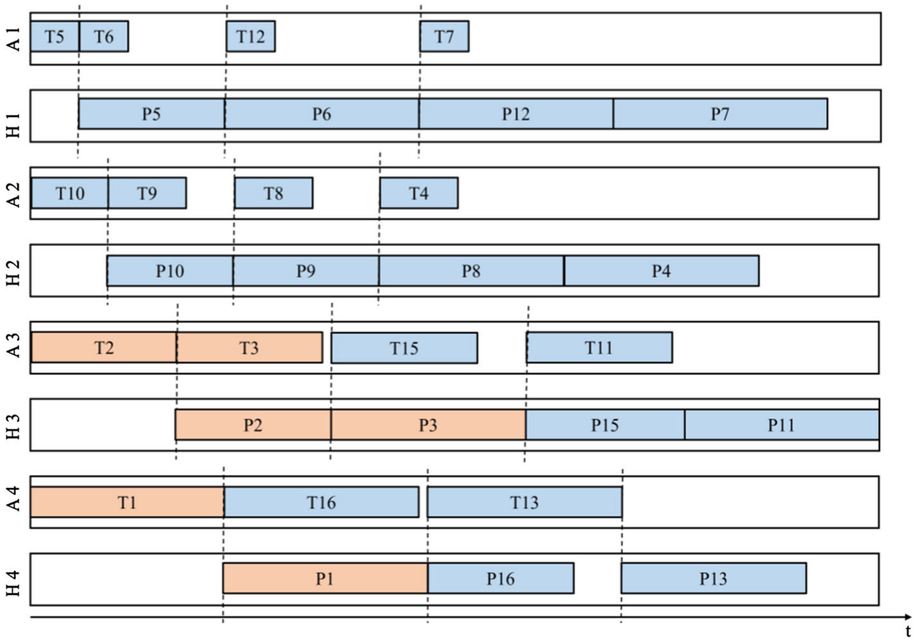


Fig. 7 The obtained patients' schedule scheme among ambulances and hospitals for case 1

When compared to FA and PSO, FA-VNS achieves significantly better results in all instances, which proves the improvement of our proposed algorithm.

Figure 8 intuitively shows the RPD of the compared algorithms. It shows that the RPD values of FA-VNS and VNS are maximal when the number of patients and hospitals is (50, 4). The RPD values of FA and PSO are maximal when the number of patients and hospitals is (50, 7). FA-VNS has more stable RPD values than the other three algorithms. The RPD values of VNS are smaller than those of FA and PSO. The RPD values of FA and PSO are similar. FA and PSO are especially unstable and VNS cannot converge to get a stable value. From Fig. 8 we can obtain the deduction that GWO-VNS is more stable and efficient compared with other three algorithms.

Figure 9 shows the differences of RPD values among the four algorithms at 95% confidence level, where the minimum, the lower and upper quartiles, median, maximum and mean value for all 24 instances are shown. It can be seen that the confidence intervals of FA and PSO are overlapped, which proves that the performance of these two algorithms is at the same level, and they are not statistically different. VNS obtains smaller minimum, lower and upper quartiles, median, maximum and mean value than those of FA and PSO. Additionally, lower and upper quartiles, median, mean value, and the difference between the upper and lower quartiles of FA-VNS are much smaller than other three algorithms. This clearly shows the best performance of FA-VNS among the all four algorithms. Besides, this result is consistent with those in Table 7 and Fig. 8.

Furthermore, the convergence curve graphs of FA-VNS, FA, VNS, PSO for the 24 instances are shown in Fig. 10 to verify the performance of convergence speed and solution qualify for the proposed algorithm. The average of fitness values in each iteration is shown in each figure. In Fig. 10, we can see that the differences of the best solutions among

Table 8 The test results for the FA-VNS with the compared algorithms

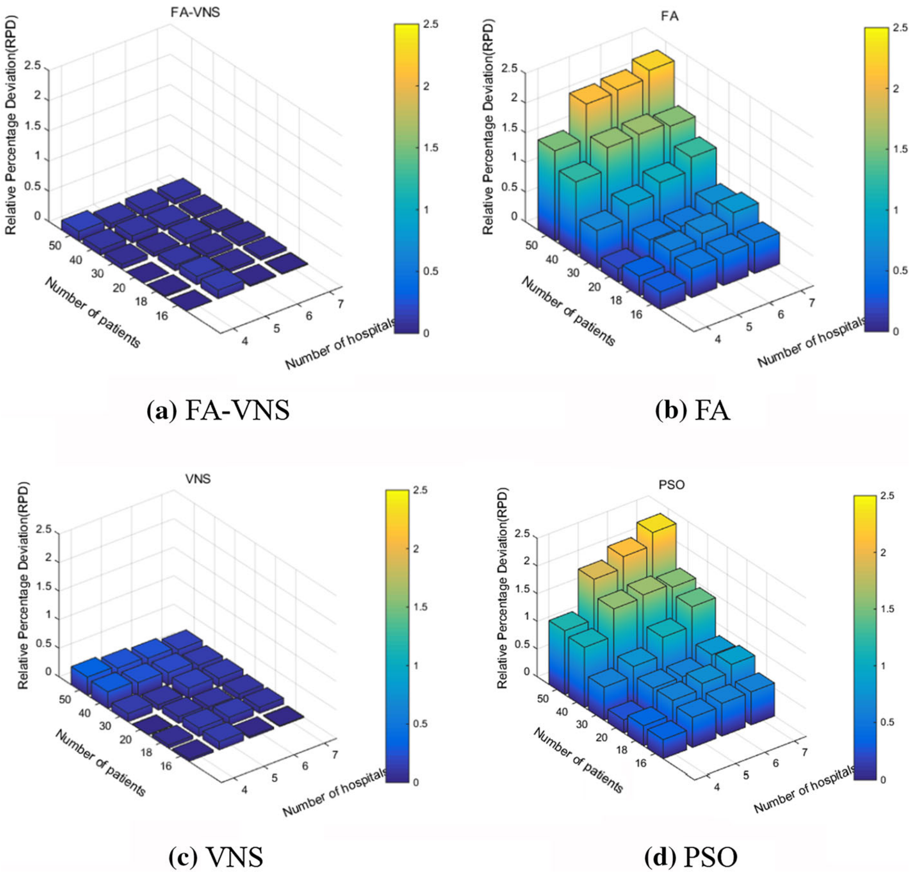
Case	(FA,FA-VNS)		(VNS,FA-VNS)		(PSO,FA-VNS)	
	ρ	h	ρ	h	ρ	h
16 × 4	0.0000	1	0.000176	1	0.0000	1
18 × 4	0.0000	1	0.000007	1	0.0000	1
20 × 4	0.0000	1	0.002222	1	0.0000	1
30 × 4	0.0000	1	0.000004	1	0.0000	1
40 × 4	0.0000	1	0.0000	1	0.0000	1
50 × 4	0.0000	1	0.000001	1	0.0000	1
16 × 5	0.0000	1	0.902961	0	0.0000	1
18 × 5	0.0000	1	0.233691	0	0.0000	1
20 × 5	0.0000	1	0.051784	0	0.0000	1
30 × 5	0.0000	1	0.001017	1	0.0000	1
40 × 5	0.0000	1	0.000001	1	0.0000	1
50 × 5	0.0000	1	0.0000	1	0.0000	1
16 × 6	0.0000	1	0.448695	0	0.0000	1
18 × 6	0.0000	1	0.000301	1	0.0000	1
20 × 6	0.0000	1	0.367648	0	0.0000	1
30 × 6	0.0000	1	0.000018	1	0.0000	1
40 × 6	0.0000	1	0.0000	1	0.0000	1
50 × 6	0.0000	1	0.0000	1	0.0000	1
16 × 7	0.0000	1	0.917625	0	0.0000	1
18 × 7	0.0000	1	0.000001	1	0.0000	1
20 × 7	0.0000	1	0.000582	1	0.0000	1
30 × 7	0.0000	1	0.0000	1	0.0000	1
40 × 7	0.0000	1	0.0000	1	0.0000	1
50 × 7	0.0000	1	0.0000	1	0.0000	1

FA-VNS, FA, VNS, and PSO become greater with the number of hospitals increasing. VNS gets better solutions than FA and PSO. Compared with FA, VNS, and PSO, FA-VNS has the fastest convergence speed in all instances and it can always get better solutions than the other three algorithms. VNS also convergences sharply, but its results are worse than ours. In nearly all instances, fast convergence speed and best solutions are realized by our proposed FA-VNS algorithm.

Based on above description and discussion, we can conclude that our FA-VNS is stable and effective in solution quality and performs well in convergence speed. In other words, our FA-VNS algorithm not only outperforms other algorithms in convergence speed and solution quality, but also maintains robust in all instances.

6.4 Sensitivity analysis

In addition to the number of patients, another important parameter that can affect the solution is the proportion critically injured. Therefore, we consider the proportion of critically injured



(a) FA-VNS

(b) FA

(c) VNS

(d) PSO

Fig. 8 RPD results in different instances for the algorithms (e.g., FA-VNS, FA, VNS, PSO)

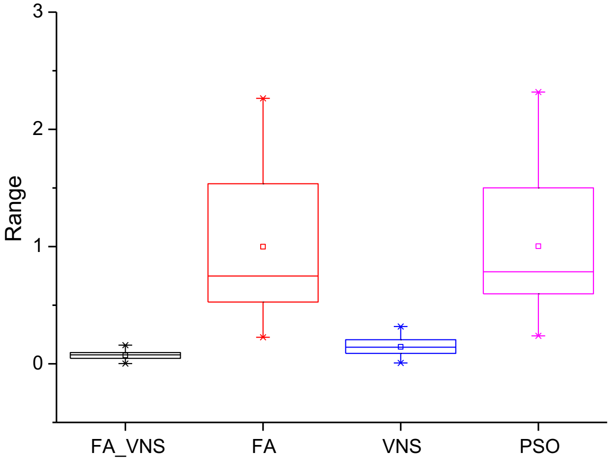
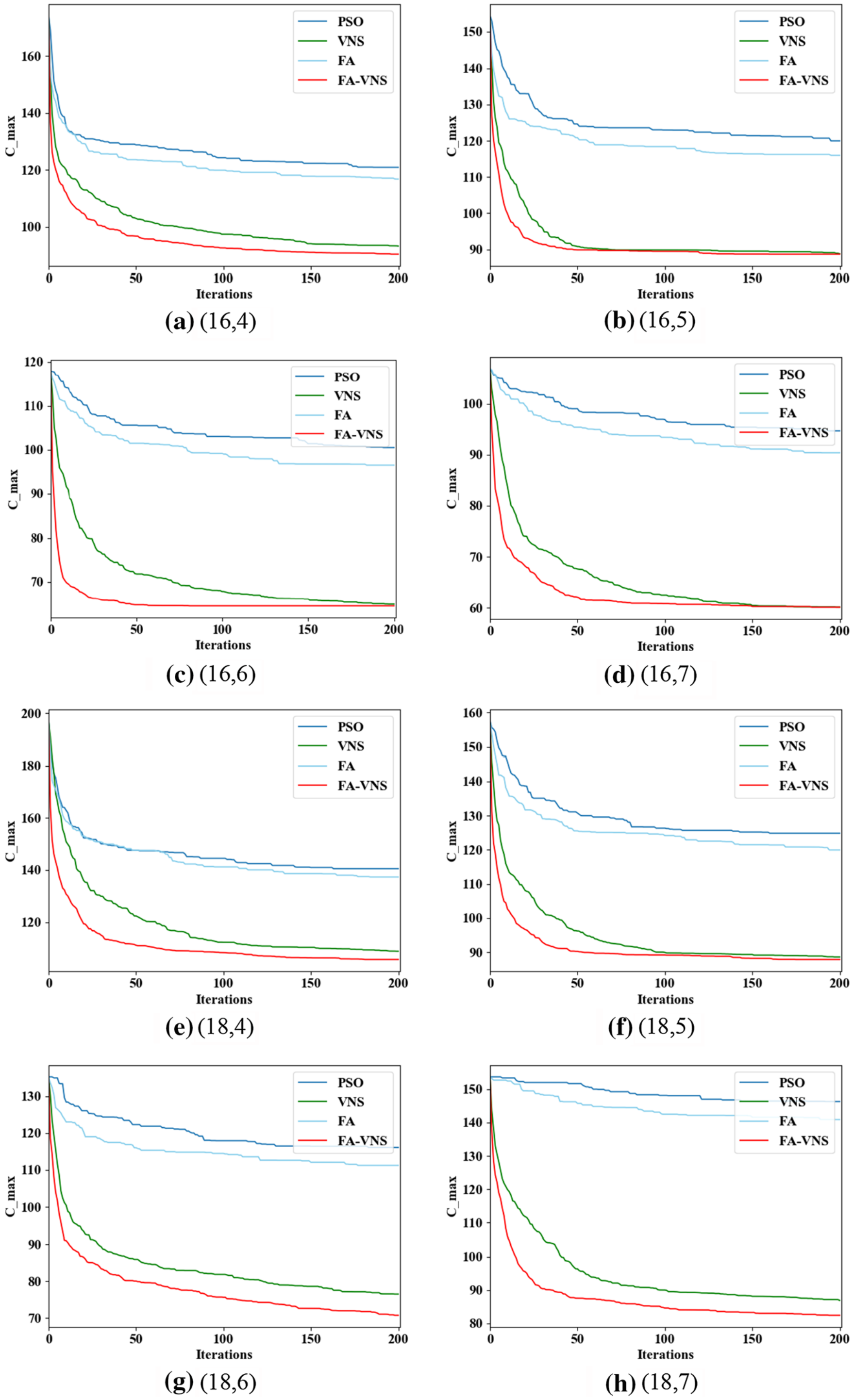


Fig. 9 The box-plot of RPD over the four algorithms (e.g., FA-VNS, FA, VNS, PSO)



(a) (16,4)

(b) (16,5)

(c) (16,6)

(d) (16,7)

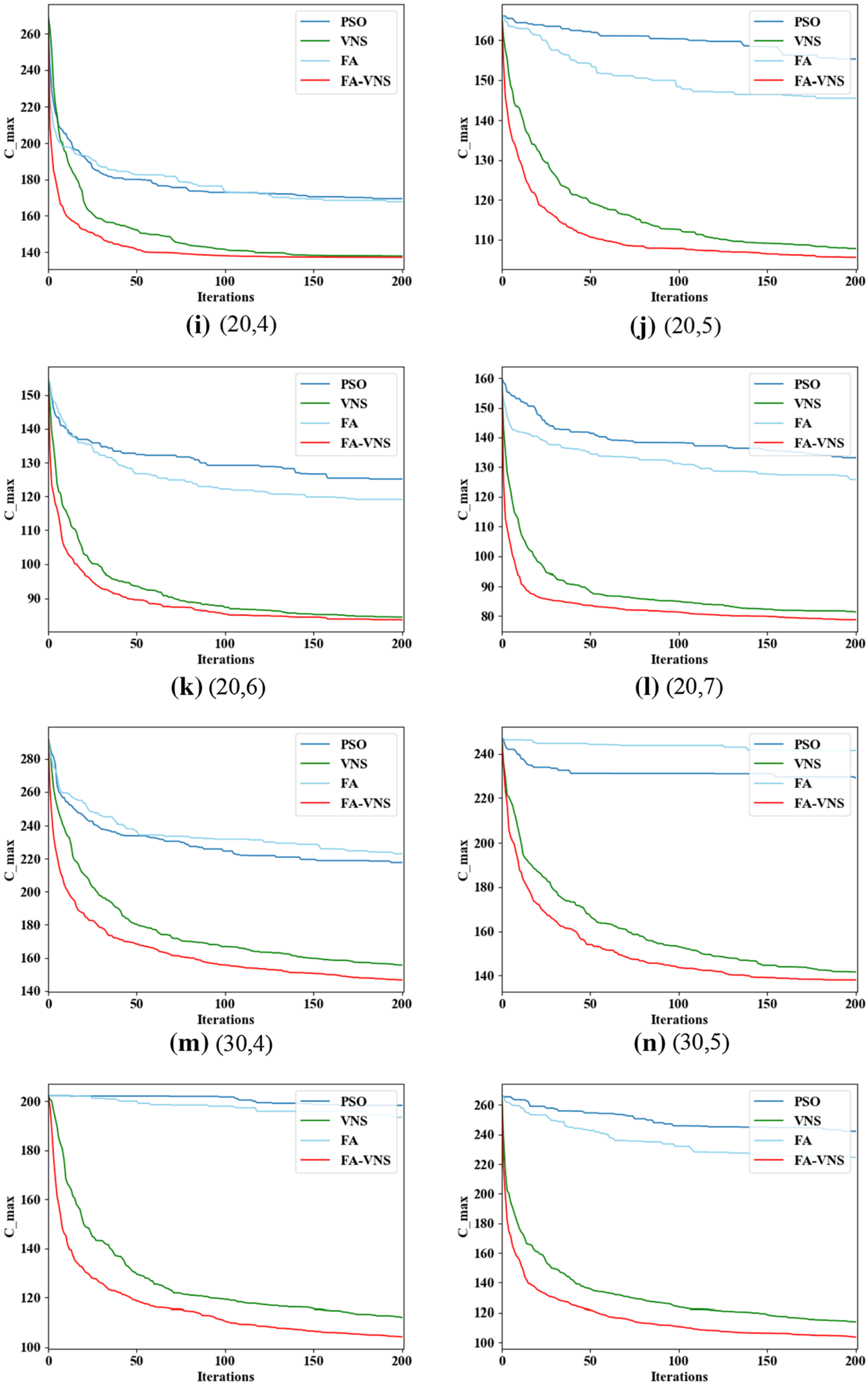
(e) (18,4)

(f) (18,5)

(g) (18,6)

(h) (18,7)

Fig. 10 Convergence curves for 24 instances over the four algorithms



(i) (20,4)

(j) (20,5)

(k) (20,6)

(l) (20,7)

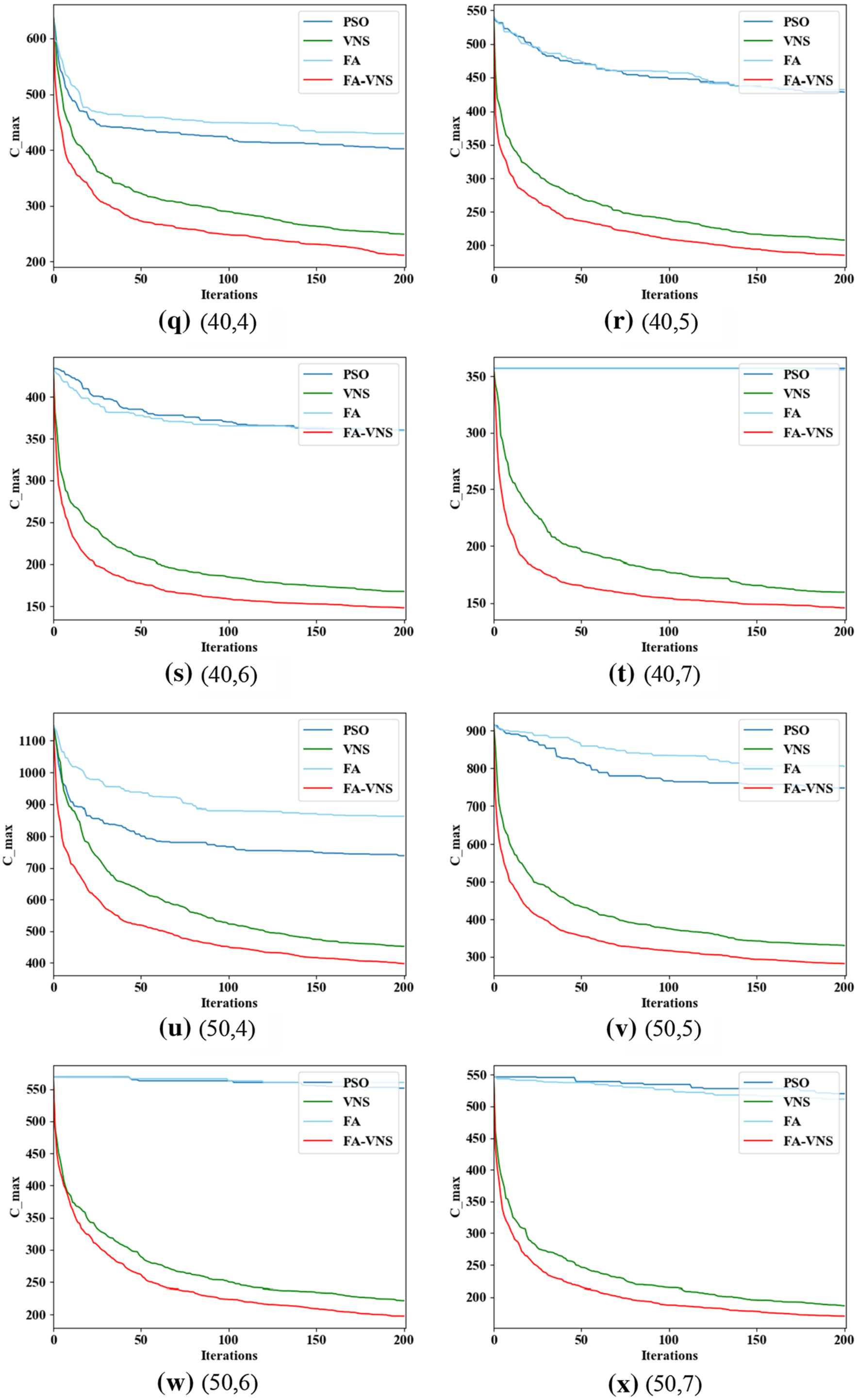
(m) (30,4)

(n) (30,5)

(o) (30,6)

(p) (30,7)

Fig. 10 continued



(q) (40,4)

(r) (40,5)

(s) (40,6)

(t) (40,7)

(u) (50,4)

(v) (50,5)

(w) (50,6)

(x) (50,7)

Fig. 10 continued

Table 9 Solution values with different parameter setting ω ($n_H = 2$)

n_I	ω	FA-VNS					
		Avg	Best	Worst	SD	No. of patients transferred to the local hospital	No. of patients transferred to the remote hospital
20	0.2	316.79	316.92	317.18	0.02	12	8
20	0.4	325.63	326.08	326.58	0.05	12	8
20	0.6	332.18	332.30	332.69	0.03	11	9
30	0.2	421.11	422.28	423.78	0.86	18	12
30	0.4	431.21	433.00	434.41	0.99	17	13
30	0.6	456.64	457.49	458.60	0.27	14	16
40	0.2	570.40	574.17	575.66	2.69	26	14
40	0.4	592.00	593.13	594.14	0.61	24	16
40	0.6	622.81	624.13	624.82	0.54	24	16
50	0.2	758.78	763.85	767.13	6.07	31	19
50	0.4	813.59	816.05	818.52	2.76	30	20
50	0.6	860.68	866.78	870.38	11.67	29	21

patients ω, that is, the larger ω, the larger the proportion of immediate patients is. We also consider the distance between the casualty collection area and hospital T_h, that is, the larger T_h, the longer the time spent in delivering the patients.

To analyze the influence of two parameters, additional experiments are conducted. The three values of ω, i.e., 0.2, 0.4, and 0.6 are considered. The number of hospitals is 2, in which one is a local hospital and another is a remote hospital.

Table 9 records the Avg, Best, Worst, SD, the number of patients transferred to the local hospital, and the number of patients transferred to the remote hospital with different values of ω. Table 8 shows that the higher the proportion of critically injured patients, the longer the surgery completion time is needed, and the more patients are needed to transported to the remote hospital. When the proportion critically injured changes, it can guide the decision-makers to adjust the schedules and predict the time required for surgeries in MCIs.

7 Conclusions

In this paper, we aim to improve the integrated problem of patient assignment and operating room scheduling considering ambulance offload delay and deteriorating condition in MCIs. A MIP model is proposed to effectively assign the limited ambulance and operating room resources for patients. The objective is to minimize the makespan. Because of the complexity of the model, only heuristic solution procedures may be used. Some structural properties of the studied problem are proposed, and a heuristic is developed to solve the single operating room scheduling problem based on these structural properties. Since the studied problem is proved to be NP-hard, a hybrid Firefly Algorithm (FA) - Variable Neighborhood Search (VNS) algorithm incorporating a heuristic method is proposed to solve it. The exact solver Gurobi is used to solve the model to verify the correctness and rationality of the established model and the known characteristics of the scheduling problem proposed in this paper. The applicability of the model is verified. Our proposed algorithm can solve the problem within a short computation time. A set of experiments is conducted to test our algorithm's performance, compared with FA, VNS, and PSO. The computational results demonstrate the superiority of our proposed algorithm over the compared algorithms. In addition, the effects of the parameters, the proportion critically injured, on the problem are analyzed. The sensitivity analysis shows the higher the proportion of critically injured patients, the longer the surgery completion time is needed, and the more patients needed to transported to the remote hospitals. When the proportion critically injured changes, it can guide the decision-makers to adjust the schedules and predict the time required for surgeries in MCIs.

At the same time, we believe that our proposed model and algorithms can help decision-makers decrease the time spent on deciding destination hospital, transportation sequence and surgery sequence, and respond to MCIs more effectively and efficiently. Some readers would realize the limitations of our algorithm, including that we assume only one ambulance assigned to each fixed route, while the assumption is appropriate as long as the ambulance utilization is very high and the available ED resource is limited.

In our future work, we consider extending our model to accommodate multiple operating rooms and dynamic routing of ambulances. Also, take death possibility before receiving surgeries into consideration is of great value as it may arise in real-world applications. In addition, if demand is higher than all hospitals available, we could consider transferring stabilizing patients to more distant facilities and clear emergency departments to accommodate more casualties. It will also be of great value to extend the current model for stochastic environments and incorporate some uncertainty into the parameters of the model, like arrival rate, damaged road networks, transport cost and time to make the problem more realistic. Through further research, we would develop more effective algorithms to solve the practical problems and improve the health care systems' surge capacity to handle more significant numbers of casualties.

Acknowledgements This work is supported by the National Natural Science Foundation of China (Nos. 72071057, 71922009, and 72188101), the Basic scientific research Projects in central colleges and Universities (JZ2022HGQA0135).

Appendix

In this appendix, the proof of Lemma 2 in Sect. 4.2 is presented.

Proof Given that an operating room has been assigned a set of patients, there exist two schedules for the operating room. We assume that π^* is an optimized schedule where patient i precedes patient$j$ to start the surgery, while in $\pi^{'}$ schedule, patient$j$ precedes patient i to start the surgery. That is, $\pi^* = (\ldots, I_A, I_i, I_j, I_B, \ldots)$, and $\pi^{'} = (\ldots, I_A, I_j, I_i, I_B, \ldots)$.

For π^* and $\pi^{'}$ schedule, the surgery start time of patient A is given as $S_A(\pi^*) = S_A(\pi^{'}) = S_A$, and the surgery duration of patient A is $p_A^* = p_A^{'} = p_A$, and thus the complete time of patient A is $C_A(\pi^*) = C_A(\pi^{'}) = C_A$.

For π^*, $S_i(\pi^*) = \max\{S_A(\pi^*) + T_h, C_A(\pi^*)\} = \max\{S_A + T_h, C_A(\pi^*)\}$. There exist two situations: (1) $T_h > p_A$; (2) $T_h < p_A$.

(1) In situation (1), $\mathrm{T_h} > \mathrm{p_A}$:
$S_i(\pi^*) = S_A + T_h$, $C_i(\pi^*) = S_i(\pi^*) + p_i^* = S_i(\pi^*) + (\overline{p_i} + \beta S_i(\pi^*)) = (1+\beta)S_i(\pi^*) + \overline{p_i} = (1+\beta)(S_A + T_h) + \overline{p_i}$, and $S_j(\pi^*) = \max\{S_i(\pi^*) + T_h, C_i(\pi^*)\} = \max\{S_A + 2T_h, (1+\beta)(S_A + T_h) + \overline{p_i}\}$. There are two cases:

(a) When $T_h > \frac{\beta S_A+\overline{p_i}}{1-\beta}$:

$S_j(\pi^*) = S_A + 2T_h$, $C_j(\pi^*) = S_j(\pi^*) + p_j^* = S_j(\pi^*) + \left(\overline{p_j} + \beta S_j(\pi^*)\right) = (1+\beta)S_j(\pi^*) + \overline{p_j} = (1+\beta)(S_A + 2T_h) + \overline{p_j}$. Then $S_B(\pi^*) = \max\{S_j(\pi^*) + T_h, C_j(\pi^*)\} = \max\{S_A + 3T_h, (1+\beta)(S_A + 2T) + \overline{p_j}\}$. If $T_h > \frac{\beta S_A+\overline{p_j}}{1-2\beta}$,$S_B(\pi^*) = S_A + 3T_h$. Otherwise, $S_B^* = (1+\beta)(S_A + 2T_h) + \overline{p_j}$.

(b) When $T_h < \frac{\beta S_A+\overline{p_i}}{1-\beta}$:
$S_j(\pi^*) = (1+\beta)(S_A + T_h) + \overline{p_i}$, $C_j(\pi^*) = S_j(\pi^*) + p_j^* = S_j(\pi^*) + \left(\overline{p_j} + \beta S_j(\pi^*)\right) = (1+\beta)S_j(\pi^*) + \overline{p_j} = (1+\beta)[(1+\beta)(S_A + T_h) + \overline{p_i}] + \overline{p_j}$. Then $S_B(\pi^*) = \max\{S_j(\pi^*)+T_h, C_j(\pi^*)\} = \max\{(1+\beta)(S_A+T_h)+\overline{p_i}+T_h, (1+\beta)[(1+\beta)(S_A + T_h) + \overline{p_i}] + \overline{p_j}\}$. If $T_h > \frac{\left(\beta^2+\beta\right)S_A+(1+\beta)\overline{p_i}+\overline{p_j}}{1-\beta-\beta^2}$, $S_B(\pi^*) = (1+\beta)(S_A+T_h)+\overline{p_i}+T_h$. Otherwise, $S_B(\pi^*) = (1+\beta)[(1+\beta)(S_A + T_h) + \overline{p_i}]+\overline{p_j}$.

(2) In situation (2),$\mathrm{T_h} < \mathrm{p_A}$:
$S_i(\pi^*) = C_A$,$C_i(\pi^*) = S_i(\pi^*) + p_i^* = S_i(\pi^*) + (\overline{p_i} + \beta S_i(\pi^*)) = (1+\beta)S_i(\pi^*) + \overline{p_i} = (1+\beta)C_A + \overline{p_i}$, and $S_j(\pi^*) = \max\{S_i^* + T_h, C_i(\pi^*)\} = \max\{C_A + T_h, (1+\beta)C_A + \overline{p_i}\}$. There are two cases:

(a) When $T_h > \beta C_A + \overline{p_i}$:
$S_j(\pi^*) = C_A + T_h$, $C_j(\pi^*) = S_j(\pi^*) + p_j^* = (1+\beta)S_j(\pi^*) + \overline{p_j} = (1+\beta)(C_A + T_h) + \overline{p_j}$. Then $S_B(\pi^*) = \max\{S_j(\pi^*) + T_h, C_j(\pi^*)\} = \max\{C_A + 2T_h, (1+\beta)(C_A + T_h) + \overline{p_j}\}$. If $T_h > \frac{\beta C_A+\overline{p_j}}{1-\beta}$, $S_B(\pi^*) = C_A + 2T_h$. Otherwise, $T_h < \frac{\beta C_A+\overline{p_j}}{1-\beta}$,$S_B(\pi^*) = (1+\beta)(C_A + T) + \overline{p_j}$

(b) When $T_h < \beta C_A + \overline{p_i}$,
$S_j(\pi^*) = (1+\beta)C_A + \overline{p_i}$, $C_j(\pi^*) = S_j(\pi^*) + p_j^* = (1+\beta)S_j(\pi^*) + \overline{p_j} = (1+\beta)[(1+\beta)C_A + \overline{p_i}]+\overline{p_j}$. Then $S_B(\pi^*) = \max\{S_j(\pi^*)+T_h, C_j(\pi^*)\} = \max\{(1+\beta)C_A + \overline{p_i} + T_h, (1+\beta)[(1+\beta)C_A + \overline{p_i}] + \overline{p_j}\}$. If $T_h > \left(\beta^2+\beta\right)C_A + \beta\overline{p_i} + \overline{p_j}$, $S_B(\pi^*) = (1+\beta)C_A+\overline{p_i}+T_h$. Otherwise, $T_h < \left(\beta^2+\beta\right)C_A+\beta\overline{p_i}+\overline{p_j}$,$S_B(\pi^*) = (1+\beta)[(1+\beta)C_A + \overline{p_i}] + \overline{p_j}$.

Similarly, we can obtain that for $\pi^{'}$:

(1) In situation (1),$T_h > \mathrm{p_A}$

(a) When $T_h > \max\{\frac{\beta S_A+\overline{p_j}}{1-\beta}, \frac{\beta S_A+\overline{p_i}}{1-2\beta}\}$, $S_B(\pi^{'}) = S_A + 3T_h$.
(b) When $\frac{\beta S_A+\overline{p_j}}{1-\beta} < T_h < \frac{\beta S_A+\overline{p_i}}{1-2\beta}$, $S_B(\pi^{'}) = (1+\beta)(S_A + 2T_h) + \overline{p_i}$.
(c) When $\frac{\left(\beta^2+\beta\right)S_A+(1+\beta)\overline{p_j}+\overline{p_i}}{1-\beta-\beta^2} < T_h < \frac{\beta S_A+\overline{p_j}}{1-\beta}$, $S_B(\pi^{'}) = (1+\beta)(S_A+T_h)+\overline{p_j}+T_h$.
(d) When $T_h < \frac{\left(\beta^2+\beta\right)S_A+(1+\beta)\overline{p_j}+\overline{p_i}}{1-\beta-\beta^2}$, $S_B(\pi^{'}) = (1+\beta)\left[(1+\beta)(S_A + T_h) + \overline{p_j}\right]+\overline{p_i}$.

(2) In situation $T_h < \mathrm{p_A}$, there are four cases:

(a) When $T_h > \max\{\beta C_A + \overline{p_j}, \frac{\beta C_A+\overline{p_i}}{1-\beta}\}$, $S_B(\pi^{'}) = C_A + 2T_h$.
(b) When $\beta C_A + \overline{p_j} < T_h < \frac{\beta C_A+\overline{p_i}}{1-\beta}$, $S_B(\pi^{'}) = (1+\beta)(C_A + T_h) + \overline{p_i}$.
(c) When $\left(\beta^2+\beta\right)C_A + \beta\overline{p_j} + \overline{p_i} < T_h < \beta C_A + \overline{p_j}$, $S_B(\pi^{'}) = (1+\beta)C_A + \overline{p_j} + T_h$.
(d) When $T_h < \min\{\beta C_A + \overline{p_j}, \left(\beta^2+\beta\right)C_A + \beta\overline{p_j} + \overline{p_i}\}$, $S_B(\pi^{'}) = (1+\beta)\left[(1+\beta)C_A + \overline{p_j}\right] + \overline{p_i}$

By the analysis above, we conclude 14 cases in total:

(1) When $T_h > \frac{\beta S_A + \overline{p_i}}{1-2\beta}$, $S_B(\pi^*) = S_B(\pi^{'}) = S_A + 3T_h$

(2) When $\frac{\beta S_A + \overline{p_j}}{1-2\beta} < T_h < \frac{\beta S_A + \overline{p_i}}{1-2\beta}$,$S_B(\pi^*) = S_A + 3T, S_B(\pi^{'}) = (1+\beta)(S_A + 2T_h) + \overline{p_i}$

(3) When $\frac{\beta S_A + \overline{p_i}}{1-\beta} < T_h < \frac{\beta S_A + \overline{p_j}}{1-2\beta}$, $S_B(\pi^*) = (1+\beta)(S_A + 2T_h) + \overline{p_j}$, $S_B(\pi^{'}) = (1+\beta)(S_A + 2T_h) + \overline{p_i}$

(4) When $\frac{\beta S_A + \overline{p_j}}{1-\beta} < T_h < \frac{\beta S_A + \overline{p_i}}{1-\beta}$, $S_B(\pi^*) = (1+\beta)(S_A + T_h) + \overline{p_i} + T_h$, $S_B(\pi^{'}) = (1+\beta)(S_A + 2T_h) + \overline{p_i}$

(5) When $\frac{(\beta^2+\beta)S_A + (1+\beta)\overline{p_i} + \overline{p_j}}{1-\beta-\beta^2} < T_h < \frac{\beta S_A + \overline{p_j}}{1-\beta}$, $S_B(\pi^*) = (1+\beta)(S_A + T_h) + \overline{p_i} + T_h$, $S_B(\pi^{'}) = (1+\beta)(S_A + T_h) + \overline{p_j} + T$

(6) When $\frac{(\beta^2+\beta)S_A + (1+\beta)\overline{p_j} + \overline{p_i}}{1-\beta-\beta^2} < T_h < \frac{(\beta^2+\beta)S_A + (1+\beta)\overline{p_i} + \overline{p_j}}{1-\beta-\beta^2}$, $S_B(\pi^*) = (1+\beta)[(1+\beta)(S_A + T_h) + \overline{p_i}] + \overline{p_j}$, $S_B(\pi^{'}) = (1+\beta)(S_A + T_h) + \overline{p_j} + T_h$

(7) When $\mathrm{p_A} < T_h < \frac{(\beta^2+\beta)S_A + (1+\beta)\overline{p_j} + \overline{p_i}}{1-\beta-\beta^2}$,$S_B(\pi^*) = (1+\beta)[(1+\beta)(S_A + T_h) + \overline{p_i}] + \overline{p_j}$, $S_B(\pi^{'}) = (1+\beta)\left[(1+\beta)(S_A + T_h) + \overline{p_j}\right] + \overline{p_i}$

(8) When $\frac{\beta C_A + \overline{p_i}}{1-\beta} < T_h < \mathrm{p_A}$, $S_B^* = S_B^{'} = C_A + 2T_h$

(9) When $\frac{\beta C_A + \overline{p_j}}{1-\beta} < T_h < \frac{\beta C_A + \overline{p_i}}{1-\beta}$, $S_B^* = C_A + 2T_h$, $S_B^{'} = (1+\beta)(C_A + T_h) + \overline{p_i}$

(10) When $\beta C_A + \overline{p_i} < T_h < \frac{\beta C_A + \overline{p_j}}{1-\beta}$, $S_B^* = (1+\beta)(C_A + T_h) + \overline{p_j}$, $S_B^{'} = (1+\beta)(C_A + T_h) + \overline{p_i}$

(11) When $\beta C_A + \overline{p_j} < T_h < \beta C_A + \overline{p_i}$, $S_B^* = (1+\beta)C_A + \overline{p_i} + T_h$,$S_B^{'} = (1+\beta)(C_A + T_h) + \overline{p_i}$

(12) When $\left(\beta^2 + \beta\right)C_A + \beta\overline{p_j} + \overline{p_i} < T_h < \beta C_A + \overline{p_j}$, $S_B^* = (1+\beta)C_A + \overline{p_i} + T_h$,$S_B^{'} = (1+\beta)C_A + \overline{p_j} + T_h$

(13) When $\left(\beta^2 + \beta\right)C_A + \beta\overline{p_i} + \overline{p_j} < T_h < \left(\beta^2 + \beta\right)C_A + \beta\overline{p_j} + \overline{p_i}$,$S_B^* = (1+\beta)C_A + \overline{p_i} + T_h$,$S_B^{'} = (1+\beta)\left[(1+\beta)C_A + \overline{p_j}\right] + \overline{p_i}$

(14) When $T_h < \left(\beta^2 + \beta\right)C_A + \beta\overline{p_i} + \overline{p_j}$, $S_B^* = (1+\beta)[(1+\beta)C_A + \overline{p_i}] + \overline{p_j}$,$S_B^{'} = (1+\beta)\left[(1+\beta)C_A + \overline{p_j}\right] + \overline{p_i}$

These 14 cases can be summarized as shown in Table 3 in the main body of the paper.

References

Almehdawe, E., Jewkes, B., & He, Q.-M. (2013). A Markovian queueing model for ambulance offload delays. *European Journal of Operational Research, 226*(3), 602–614.

Almehdawe, E., Jewkes, B., & He, Q.-M. (2016). Analysis and optimization of an ambulance offload delay and allocation problem. *Omega, 65*, 148–158.

Barbarosoğlu, G., & Arda, Y. (2004). A two-stage stochastic programming framework for transportation planning in disaster response. *Journal of the Operational Research Society, 55*(1), 43–53.

Bernstein, S. L., Aronsky, D., Duseja, R., Epstein, S., Handel, D., Hwang, U., et al. (2009). The effect of emergency department crowding on clinically oriented outcomes. *Academic Emergency Medicine, 16*(1), 1–10.

Besiou, M., Pedraza-Martinez, A. J., & Van Wassenhove, L. N. (2018). OR applied to humanitarian operations. *European Journal of Operational Research, 269*(2), 397–405.

Carter, A. J., Overton, J., Terashima, M., & Cone, D. C. (2014). Can emergency medical services use turnaround time as a proxy for measuring ambulance offload time? *The Journal of Emergency Medicine, 47*(1), 30–35.

Cone, D. C., Middleton, P. M., & Marashi Pour, S. (2012). Analysis and impact of delays in ambulance to emergency department handovers. *Emergency Medicine Australasia, 24*(5), 525–533.

Cooney, D. R., Millin, M. G., Carter, A., Lawner, B. J., Nable, J. V., & Wallus, H. J. (2011). Ambulance diversion and emergency department offload delay: Resource document for the national association of EMS physicians position statement. *Prehospital Emergency Care, 15*(4), 555–561.

Cooney, D. R., Wojcik, S., Seth, N., Vasisko, C., & Stimson, K. (2013). Evaluation of ambulance offload delay at a university hospital emergency department. *International Journal of Emergency Medicine, 6*(1), 1–4.

Crilly, J., Keijzers, G., Tippett, V., O'Dwyer, J., Lind, J., Bost, N., et al. (2015). Improved outcomes for emergency department patients whose ambulance off-stretcher time is not delayed. *Emergency Medicine Australasia, 27*(3), 216–224.

Dean, M. D., & Nair, S. K. (2014). Mass-casualty triage: Distribution of victims to multiple hospitals using the SAVE model. *European Journal of Operational Research, 238*(1), 363–373.

Denton, B., Viapiano, J., & Vogl, A. (2007). Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health Care Management Science, 10*(1), 13–24.

Eun, J., Kim, S.-P., Yih, Y., & Tiwari, V. (2019). Scheduling elective surgery patients considering time-dependent health urgency: Modeling and solution approaches. *Omega, 86*, 137–153.

Farahani, R. Z., Lotfi, M. M., Baghaian, A., Ruiz, R., & Rezapour, S. (2020). Mass casualty management in disaster scene: A systematic review of OR&MS research in humanitarian operations. *European Journal of Operational Research, 287*(3), 787–819.

Fiedrich, F., Gehbauer, F., & Rickers, U. (2000). Optimized resource allocation for emergency response after earthquake disasters. *Safety Science, 35*(1–3), 41–57.

Frykberg, E. R. (2005). Triage: Principles and practice. *Scandinavian Journal of Surgery, 94*(4), 272–278.

Garner, A. (2003). Documentation and tagging of casualties in multiple casualty incidents. *Emergency Medicine (fremantle, WA), 15*(5–6), 475–479.

Gu, J., Zhou, Y., Das, A., Moon, I., & Lee, G. M. (2018). Medical relief shelter location problem with patient severity under a limited relief budget. *Computers & Industrial Engineering, 125*, 720–728.

Hansen, P., & Mladenović, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research, 130*(3), 449–467.

Hupert, N., Hollingsworth, E., & Xiong, W. (2007). Is overtriage associated with increased mortality? Insights from a simulation model of mass casualty trauma care. *Disaster Medicine and Public Health Preparedness, 1*(S1), S14–S24.

Ingolfsson, A., Budge, S., & Erkut, E. (2008). Optimal ambulance location with random delays and travel times. *Health Care Management Science, 11*(3), 262–274.

Ito, M., Kobayashi, F., & Takashima, R. (2018). Minimizing conditional-value-at-risk for a single operating room scheduling problems. In Proceedings of the International MultiConference of Engineers and Computer Scientists (Vol. 2).

Jacobson, E. U., Argon, N. T., & Ziya, S. (2012). Priority assignment in emergency response. *Operations Research, 60*(4), 813–832.

Jenkins, J. L., McCarthy, M. L., Sauer, L. M., Green, G. B., Stuart, S., Thomas, T. L., & Hsu, E. B. (2008). Mass-casualty triage: Time for an evidence-based approach. *Prehospital and Disaster Medicine, 23*(1), 3–8.

Kamali, B., Bish, D., & Glick, R. (2017). Optimal service order for mass-casualty incident response. *European Journal of Operational Research, 261*(1), 355–367.

Kim, C. H., Park, J. O., Park, C. B., Kim, S. C., Kim, S. J., & Hong, K. J. (2014). Scientific framework for research on disaster and mass casualty incident in Korea: Building consensus using Delphi method. *Journal of Korean Medical Science, 29*(1), 122–128.

Laan, C. M., Vanberkel, P. T., Boucherie, R. J., & Carter, A. J. (2016). Offload zone patient selection criteria to reduce ambulance offload delay. *Operations Research for Health Care, 11*, 13–19.

Lee, K., Lei, L., Pinedo, M., & Wang, S. (2013). Operations scheduling with multiple resources and transportation considerations. *International Journal of Production Research, 51*(23–24), 7071–7090.

Lei, D., & Guo, X. (2016). Variable neighborhood search for the second type of two-sided assembly line balancing problem. *Computers & Operations Research, 72*, 183–188.

Lei, L., Pinedo, M., Qi, L., Wang, S., & Yang, J. (2015). Personnel scheduling and supplies provisioning in emergency relief operations. *Annals of Operations Research, 235*(1), 487–515.

Leo, G., Lodi, A., Tubertini, P., & Di Martino, M. (2016). Emergency department management in Lazio, Italy. *Omega, 58*, 128–138.

Li, M., Vanberkel, P., & Carter, A. J. E. (2019). A review on ambulance offload delay literature. *Health Care Management Science, 22*(4), 658–675.

Łukasik, S., & Żak, S. (2009). Firefly algorithm for continuous constrained optimization tasks. In International conference on computational collective intelligence (pp. 97–106).

Majedi, M. (2008). *A queueing model to study ambulance offload delays*. University of Waterloo.

Marichelvam, M. K., Prabaharan, T., & Yang, X. S. (2013). A discrete firefly algorithm for the multi-objective hybrid flowshop scheduling problems. *IEEE Transactions on Evolutionary Computation, 18*(2), 301–305.

Marques, I., Captivo, M. E., & Vaz Pato, M. (2012). An integer programming approach to elective surgery scheduling. *Or Spectrum, 34*(2), 407–427.

Melton, R. J., & Riner, R. M. (1981). Revising the rural hospital disaster plan: A role for the EMS system in managing the multiple casualty incident. *Annals of Emergency Medicine, 10*(1), 39–44.

Mete, H. O., & Zabinsky, Z. B. (2010). Stochastic optimization of medical supply location and distribution in disaster management. *International Journal of Production Economics, 126*(1), 76–84.

Mills, A. F., Argon, N. T., & Ziya, S. (2013). Resource-based patient prioritization in mass-casualty incidents. *Manufacturing & Service Operations Management, 15*(3), 361–377.

Mills, A. F., Argon, N. T., & Ziya, S. (2018). Dynamic distribution of patients to medical facilities in the aftermath of a disaster. *Operations Research, 66*(3), 716–732.

Rachel Lu, J., Tsai, T., & Liu, S. P. (2011). Building a Hospital Alliance—Taiwan Landseed Medical Alliance. *Asian Case Research Journal, 15*(01), 123–148.

Repoussis, P. P., Paraskevopoulos, D. C., Vazacopoulos, A., & Hupert, N. (2016). Optimizing emergency preparedness and resource utilization in mass-casualty incidents. *European Journal of Operational Research, 255*(2), 531–544.

Sacco, W. J., Navin, D. M., Fiedler, K. E., Waddell, R. K., II., Long, W. B., & Buckman, R. F., Jr. (2005). Precise formulation and evidence-based application of resource-constrained triage. *Academic Emergency Medicine, 12*(8), 759–770.

Sun, Y., & Li, X. (2011). Optimizing surgery start times for a single operating room via simulation. In Proceedings of the 2011 winter simulation conference (WSC) (pp. 1306–1313).

Sun, H., Wang, Y., & Xue, Y. (2021). A bi-objective robust optimization model for disaster response planning under uncertainties. *Computers & Industrial Engineering, 155*, 107213.

Sung, I., & Lee, T. (2016). Optimal allocation of emergency medical resources in a mass casualty incident: Patient prioritization by column generation. *European Journal of Operational Research, 252*(2), 623–634.

Taherkhani, M., & Safabakhsh, R. (2016). A novel stability-based adaptive inertia weight for particle swarm optimization. *Applied Soft Computing, 38*, 281–295.

Vallada, E., & Ruiz, R. (2011). A genetic algorithm for the unrelated parallel machine scheduling problem with sequence dependent setup times. *European Journal of Operational Research, 211*(3), 612–622.

Wang, D., Liu, F., Yin, Y., Wang, J., & Wang, Y. (2015). Prioritized surgery scheduling in face of surgeon tiredness and fixed off-duty period. *Journal of Combinatorial Optimization, 30*(4), 967–981.

Wilson, D. T., Hawe, G. I., Coates, G., & Crouch, R. S. (2013). A multi-objective combinatorial model of casualty processing in major incident response. *European Journal of Operational Research, 230*(3), 643–655.

Xiang, Y., & Zhuang, J. (2016). A medical resource allocation model for serving emergency victims with deteriorating health conditions. *Annals of Operations Research, 236*(1), 177–196.

Xiao, G., van Jaarsveld, W., Dong, M., & van de Klundert, J. (2018). Models, algorithms and performance analysis for adaptive operating room scheduling. *International Journal of Production Research, 56*(4), 1389–1413.

Yan, K., Jiang, Y., Qiu, J., Zhong, X., Wang, Y., Deng, J., et al. (2017). The equity of China's emergency medical services from 2010–2014. *International Journal for Equity in Health, 16*(1), 10.

Yang, X.-S. (2010). *Nature-inspired metaheuristic algorithms*. Luniver press.

Yang, X.-S., Hosseini, S. S. S., & Gandomi, A. H. (2012). Firefly algorithm for solving non-convex economic dispatch problems with valve loading effect. *Applied Soft Computing, 12*(3), 1180–1186.

Ye, Y. (2018). *Based on pre-hospital emergency investigation to explore the configuration of emergency vehicles in China*. Hainan Medical University.

Zheng, Y. J., Chen, S. Y., & Ling, H. F. (2015). *Evolutionary optimization for disaster relief operations: a survey*. Elsevier Science Publishers B. V.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.