



HAL
open science

Bi-Objective Scheduling of Fire Engines for Fighting Forest Fires: New Optimization Approaches

Peng Wu, Feng Chu, Ada Che, Mengchu Zhou

► **To cite this version:**

Peng Wu, Feng Chu, Ada Che, Mengchu Zhou. Bi-Objective Scheduling of Fire Engines for Fighting Forest Fires: New Optimization Approaches. *IEEE Transactions on Intelligent Transportation Systems*, 2018, 19 (4), pp.1140–1151. 10.1109/TITS.2017.2717188 . hal-01599180

HAL Id: hal-01599180

<https://hal.science/hal-01599180>

Submitted on 23 May 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Bi-Objective Scheduling of Fire Engines for Fighting Forest Fires: New Optimization Approaches

Peng Wu, Feng Chu, *Senior Member, IEEE*, Ada Che, *Senior Member, IEEE*, and MengChu Zhou, *Fellow, IEEE*

Abstract—It is challenging to perform emergency scheduling for fighting forest fires subject to limited rescue resources (i.e., vehicles with fire engines), since extinguishing each fire point should take into account multiple factors, such as the actual fire spreading speed, distance from fire engine depot to fire points, fire-fighting speed of fire engines, and the number of dispatched vehicles. This paper investigates a bi-objective rescue vehicle scheduling problem for multi-point forest fires, which aims to optimally dispatch a limited number of fire engines to extinguish fires. The objectives are to minimize the total fire-extinguishing time and the number of dispatched fire engines. For this problem, we first develop an integer program that is an improved and simplified version of an existing one. After exploring some properties of the problem, we develop an exact dynamic programming algorithm and a fast greedy heuristic method. Computational results for a real-life instance, and benchmark and large-size randomly generated instances confirm the effectiveness and efficiency of the proposed model and algorithms. Besides, a bi-objective integer program is developed to address the multi-depot fire engine scheduling issue.

Index Terms—Forest fires, emergency scheduling, optimization, multi-objective, Pareto front.

I. INTRODUCTION

IN RECENT years, natural disasters and man-made catastrophic events happen frequently, which have caused tremendous harm to human beings. Effective and efficient emergency management is crucial in ensuring timely rescue and relief operations. Currently, the main

studies about emergency management focus on emergency response management [1]–[5], relief distribution [6]–[11], and emergency logistics management [12]–[16]. To deal with the related problems, a number of modelling approaches, e.g., integer programming [17], [18], fuzzy optimization [19], and dynamic programming [20], and solution techniques, e.g., simulation [21], [22] and meta-heuristic [23], have been proposed. Recent surveys related to various emergency operations can refer to [24]–[26].

Many highlighted works have adequately addressed emergency planning and scheduling problems under various disasters and emergency events such as earthquakes, hurricane, tsunamis, and floods. However, limited focus has been on the forest fires. Notably, with the warming of global climate, forest fires around the world become increasingly frequent. As reported in [27], the forest fires occur over 22 million times each year in the world and burn more than 6.4 million hectares of forest area, accounting for more than 0.23% of the world’s forest coverage. In China, the forest area engulfed by forest fire each year is about 1.1 million hectares, accounting for about 0.8% to 0.9% of the whole national forest area. Moreover, many forest fires involve multiple fire points simultaneously. For example, the forest fires occurred in Nanwenghe region located in Mt. Daxing’anling on March 19, 2003 and Huzhong region located in Mt. Daxing’anling on June 29, 2010 involve four and seven fire points, respectively. Forest fire has been recognized as one of the eight major natural disasters [27]. The basic forest fire-fighting workflow is illustrated in Fig. 1 [28]. Once forest fires happen, the fire information is collected and the fire level is determined. Then, the emergency plan is launched. Next, the fire-fighting plan is executed, which usually involves five main groups (i.e., organization and command, emergency communication, fire engines scheduling, logistics support and medical service, and propaganda), followed by fighting fires and reporting the results. Finally, the firegrounds are cleaned and guarded and the post-disaster disposal is conducted.

In this study, we focus on the fire engine scheduling that is finished by the fire engines scheduling group as indicated by dashed line rectangle in Fig. 1. For each fire engine dispatched, it is usually assigned with an identical quantity of firemen and fire-fighting equipment [29]. The most critical question facing the fire engine scheduling group lies in how to dispatch fire engines to extinguish the fire points in the most efficient manner. Therefore, it is necessary to develop

P. Wu is with the School of Economics and Management, Fuzhou University, Fuzhou 350116, China, and also with the School of Management, Northwestern Polytechnical University, Xi’an710072, China (e-mail: wupeng88857@gmail.com).

F. Chu is with IBISC, Univ Évry, University of Paris-Saclay, 91025 Évry, France, and also with the Management Engineering Research Center, Xihua University, Chengdu 610039, China (e-mail: feng.chu@ibisc.univ-evry.fr).

A. Che is with the School of Management, Northwestern Polytechnical University, Xi’an 710072, China (e-mail: ache@nwpu.edu.cn).

M. Zhou is with the Institute of Systems Engineering, Macau University of Science and Technology, Macau 999078, China and also with the Helen and John C. Hartmann Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: zhou@njit.edu).

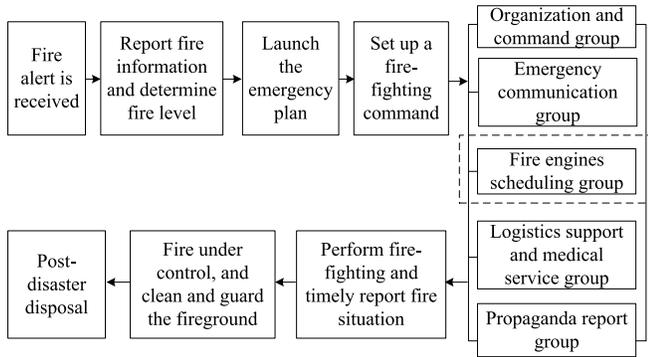


Fig. 1. Schematic illustration of the basic workflow of the forest fire-fighting.

effective scheduling approaches to achieve the highest fire-fighting efficiency while reducing or limiting the expenditures.

Different from the other natural disasters, once forest fires happen, different fire points may spread at various speeds that are usually determined by natural conditions, e.g., wind force, terrain slope, and fuel types [30]. For example, after the forest fires occurred in Mt. Daxing'anling, China in 2010, its seven fire points spread at different speeds varying between 2.2 m/min and 6.98 m/min [29]. Different forest fire starting time would affect the actual fire engine scheduling. For example, if the starting time intervals between different fire points are large, decision-makers can address them one by one. However, if the starting time intervals between different fire points are short, the fire engine scheduling should consider multiple fire points simultaneously. In addition, extinguishing each fire point should take into account multiple other factors such as the fire spreading speed, distance from fire engines depot to fire points, dispatched fire engine count and their fire-fighting speed. Consequently, due to these special characteristics, the existing methodologies for emergency issues like earthquakes cannot be directly applied to the emergency response scheduling to tackle forest fires. Furthermore, we note that most of the existing studies mainly focused on minimizing the emergency response time or response costs. However, to make a well-informed decision, decision-makers may desire to know the service performance corresponding to different resource usage, i.e., the trade-off between the two aspects. To simultaneously optimize the emergency response service and resource usage can help decision-makers make a good trade-off between them. For example, they usually desire to minimize the total fire-extinguishing time and expenditures for dealing with the forest fires at the same time. Besides, a number of studies, e.g., [13], [31], [32], have also pointed out that minimizing emergency response costs is as important as minimizing the emergency response time. Consequently, multi-objective optimization is required to support the desired decision-making for processing emergency issues.

To the best of our knowledge, no studies in the literature have addressed the emergency planning and scheduling for the forest fires except the recent work [29]. Tian *et al.* pioneered in a bi-objective optimization model to simultaneously optimize the total fire-extinguishing time and the total number of fire engines dispatched [29]. To solve the proposed model, they developed a bi-objective hybrid meta-heuristic algorithm

to obtain a set of Pareto solutions. Instances with up to 50 fire points and 250 fire engines were solved. However, their study has three deficiencies. Firstly, their obtained solutions are not guaranteed to be the best (Pareto optimal) due to the nature of meta-heuristic algorithms. Secondly, the computational time of their method is relatively long to provide satisfactory solutions, especially for large-size problems, which may greatly increase the total time to extinguish fires. Thirdly, the work [29] does not address the multi-depot fire engine scheduling issue. To overcome the above deficiencies, this work intends to develop effective and efficient approaches for solving the problem and address the multi-depot fire engine scheduling. The main contributions of this paper are as follows:

- 1) We propose an improved and simplified integer program with fewer variables and constraints. Comparison results show that the improved model significantly outperforms the existing one. Moreover, we extend the proposed model to the multi-depot case.
- 2) We prove that the optimal fire-extinguishing time is a strictly decreasing function of the number of dispatched fire engines. Then, we derive the lower and upper bounds on the number of dispatched fire engines and show that the number of non-dominated points over the Pareto front is their linear function.
- 3) Based on the derived problem properties, we develop an exact dynamic programming algorithm and a fast greedy heuristic method for solving the problem. Computational results for a real-life case, benchmark and randomly generated large-size instances confirm the effectiveness and efficiency of the proposed algorithms. Besides, the ε -constraint method is adapted to obtain the Pareto front of the multi-depot fire engine scheduling problem.

The remainder of the paper proceeds as follows. In Section II, the related literature is reviewed. The problem description is recalled and an improved bi-objective integer program is provided in Section III. Section IV analyzes the properties of the problem. Based on them, an exact dynamic programming algorithm and a greedy heuristic method are developed in Section V. Section VI addresses the fire engine scheduling problem with multiple depots. In Section VII, computational experiments are conducted. Conclusions and future research issues are discussed in Section VIII.

II. LITERATURE REVIEW

Emergency rescue and relief operations are mainly dependent on surface transportation. As a consequence, well-planned roadway networks and efficient rescue vehicle scheduling after a natural disaster or catastrophic event takes place is critical to ensure response efficiency and reduce costs. Over the past decades, a number of studies concentrated on emergency planning and scheduling for various natural disasters and catastrophic events. For instance, Haghani and Oh [13] presented a multi-commodity, multi-modal network model for determining the rescue vehicle routing and scheduling when facing natural disasters (e.g., earthquakes and hurricanes) to minimize the cost. Two heuristics were developed to solve it. Later, Özdamar *et al.* [14] presented a rescue vehicle

routing model for optimally dispatching rescue commodities (e.g., medical materials and personnel) in natural disasters, e.g., earthquakes, to minimize the unsatisfied demands. A Lagrangean relaxation heuristic was proposed to yield near-optimal solutions. Campbell *et al.* [8] formulated a rescue vehicle routing problem in aftermath of a natural disaster (e.g., earthquake, hurricane, and tsunami) as variants of the traveling salesman problem (TSP) and vehicle routing problem (VRP). Effective heuristics were proposed as well. Yuan and Wang [33] constructed a path selection model to minimize the total rescue time in emergency logistics management and proposed a modified Dijkstra algorithm for it. Zhang *et al.* [34] developed a bio-inspired algorithm to address the path selection in an emergency logistics network. Moreover, Wex *et al.* [35] constructed a rescue unit allocation model to process incidents caused by natural disasters, e.g., earthquakes, tsunamis and floods with the objective of minimizing the total completion time. Several heuristics were suggested to solve it.

Particularly, to effectively respond to earthquakes, Hu *et al.* [36] presented an optimal earthquake emergency shelter allocation model to optimize the total operating cost while meeting all victims' needs. A particle swarm optimization based meta-heuristic was suggested to tackle the model. Yan *et al.* [37] developed a logistical support scheduling model for damaged roadway repair work after an earthquake to minimize the total operating cost. Later, Yan *et al.* [10] established an emergency scheduling model for highway emergency repairs under demand-side perturbations to minimize the overall completion time of repairing all highway segments. An ant colony system based hybrid meta-heuristic algorithm was developed to derive near-optimal solutions.

Besides, Haghani *et al.* [21] developed a simulation model for optimal emergency vehicle dispatching and routing during accidents to minimize the average response time. Later, the work [38] presented an online dispatching and routing model for optimally assigning emergency response vehicles to incidents as well as determining the routes to minimize the total emergency response time. Recently, Park *et al.* [39] constructed a stochastic emergency response location-allocation model for processing traffic incidents on highways with the objective of minimizing the expected delay of all scenarios. A heuristic was developed to solve it. Haghani and Qiao [31], [32] developed salting truck routing models for dealing with snow emergencies to minimize the number of trucks used and dead-head time. Constructive heuristics were proposed to obtain satisfactory solutions.

From the existing research results, it can be concluded that the prior studies on emergency planning and scheduling mainly focus on processing emergency issues including earthquakes, hurricane, tsunamis, floods, snow emergencies, and accidents. As previously mentioned, the forest fire has its own special characteristics. The recent work [29] pioneered in the first multi-objective model and hybrid meta-heuristic for addressing the fire engine scheduling problem. However, the existing model is very hard to solve and the existing algorithm is time-consuming and may fail to find the Pareto front. In this

research, we improve the existing research result [29] in a number of ways by improving the formulation, developing new efficient exact and heuristic algorithms, and extending the studied one-depot problem to a multi-depot case.

III. PROBLEM DESCRIPTION AND FORMULATION

The emergency scheduling problem for forest fires can be defined over an undirected graph $G = (V', E)$, where $V' = \{0\} \cup V = \{1, \dots, n\}$ (resp. E) is the set of vertices (resp. the set of edges). Without loss of generality, vertex 0 represents the starting point that hosts all fire engines and vertex $i \in V$ denotes the i -th forest fire point to be extinguished.

As is widely acknowledged, once a major fire disaster with multiple fire points happens simultaneously in a region, to extinguish all the fire points as quickly as possible is desired for each decision-maker. The whole rescue time for fire point $i \in V$ is composed of the travel time of fire engines, denoted by T_{Ai} , and extinguishing time, denoted by t_{Ei} . T_{Ai} is assumed to be calculated by the distance between the starting point 0 and fire point i , denoted by d_{0i} , and the average driving speed of fire engines, denoted by v_{0i} . $T_{Ai} = d_{0i}/v_{0i}$ in this paper. However, it is much more complex and difficult to calculate t_{Ei} , since it depends on multiple factors, such as the fire spread speed v_{si} , travel time of fire engines T_{Ai} , fire-fighting speed of fire engines v_f and dispatched vehicle count y_i . In the paper, we compute t_{Ei} using the following formula as proposed by [29]:

$$t_{Ei} = v_{si}T_{Ai}/(y_iv_f - v_{si}), \quad \forall i \in V$$

It can be seen that the fire lasting time is influenced by vehicle scheduling. Conversely, it would influence vehicle scheduling, since one objective is to minimize the total fire lasting time.

Remark 1: For any fire point $i \in V$, the value of $y_iv_f - v_{si}$ should be greater than 0; otherwise this fire point cannot be extinguished. In other words, $y_i > v_{si}/v_f$ should be ensured in processing fires to derive a feasible scheme.

The fire spread speed for any given fire point is determined by the following fire spread model taking into account natural factors, e.g., wind force, terrain slope, and fuel types [29].

$$v_s = v_0\lambda_s\lambda_\varphi\lambda_w = v_0\lambda_s\lambda_\varphi e^{0.1783v_w}$$

where v_s is the fire spread speed; $v_0 = \alpha T + \beta w + \gamma$ denotes the initial spread speed; λ_s , λ_φ and λ_w are the correction factors of fuel types, terrain slope and wind force, respectively; T and w represent the temperature and wind force, respectively; α , β and γ are the factors related with the terrain, respectively. According to the above formula for computing t_{Ei} , the fire spread speeds of different fire points, denoted by v_{si} , can greatly differ among each other due to different natural conditions. For more details on the above formulas, see [29]. Since the total number of fire engines is limited, it can be imagined that if one fire point is assigned with more fire engines, the available fire engines to be dispatched to the rest becomes fewer. However, different fire points are independent such that the spread of different fire points would not influence each other.

TABLE I
NOTATIONS USED TO DESCRIBE THE PROBLEM

Input parameters	
$G = \{V', E\}$	an undirected graph where V' and E denote the set of vertices and edges, respectively;
V	set of fire points, $V \subset V'$;
n	total number of fire points;
m	total number of fire engines for fighting forest fires;
U_i	upper bound on the number of fire engines sent to fire point $i \in V$;
L_i	lower bound on the number of fire engines sent to fire point i , $i \in V$, and note that $L_i \geq \lceil v_{si}/v_f \rceil$ based on Remark 1;
LB	lower bound on the total number of dispatched fire engines, and is set to be the sum of L_i ;
v_{si}	spread speed of fire point i , $i \in V$;
d_{ij}	distance from vertices i to j , $i, j \in V$;
v_{ij}	average travel speed of fire engines from vertices i to j , $i, j \in V'$;
T_{Ai}	arrival time of fire engines to forest fire point i , $T_{Ai} = d_{0i}/v_{0i}$, $i \in V$;
v_f	average fighting speed offered by a fire engine;
Decision variables	
t_{Ei}	the time to extinguish fire point i , $i \in V$;
x_{ij}	$x_{ij} = 1$, if fire engine j is sent to fire point i ; 0 otherwise, $i \in V, j = 1, 2, \dots, m$;
y_i	the number of fire engines dispatched to fire point i from the fire engine depot 0, $i \in V$, for the improved model.

As desired by decision-makers, to minimize the total fire-extinguishing time is the primary goal. Meanwhile, the dispatched fire engine count is desired to be optimized as the second objective. This study aims to optimize the above two objectives simultaneously via an optimal emergency schedule such that an appropriate number of fire engines are dispatched to different fire points.

A. Problem Formulation

With the notations presented in Table I, we first recall the following integer program presented in [29].

$$P_0 : f_1 : \min \sum_{i=1}^n t_{Ei} \quad (1)$$

$$f_2 : \min \sum_{i=1}^n \sum_{j=1}^m x_{ij} \quad (2)$$

$$\text{s.t. } t_{Ei} = v_{si} T_{Ai} / \left(\sum_{j=1}^m x_{ij} v_f - v_{si} \right), \quad \forall i \in V \quad (3)$$

$$LB \leq \sum_{i=1}^n \sum_{j=1}^m x_{ij} \leq m \quad (4)$$

$$L_i \leq \sum_{j=1}^m x_{ij} \leq U_i, \quad \forall i \in V \quad (5)$$

$$x_{ij} \in \{0, 1\}, \quad j = 1, 2, \dots, m, \quad \forall i \in V \quad (6)$$

$$t_{Ei} \geq 0, \quad \forall i \in V \quad (7)$$

Objective (1) is to minimize the total fire-extinguishing time. Objective (2) is to minimize the number of dispatched rescue vehicles. Constraint (3) defines the fire-extinguishing time of

fire point $i \in V$. Constraint (4) aims to guarantee that the total number of fire engines sent to all fire points cannot be greater than the number of available fire engines m and less than LB vehicles such that all fire points can be extinguished. Constraint (5) restricts the number of fire engines dispatched to each fire point. Constraints (6) and (7) impose restrictions on decision variables. It can be observed that model P_0 employs mn integer variables and $3n + 2$ constraints.

B. An Improved Formulation

In this section, we improve the above formulation. Firstly, we can directly employ integer variable y_i to replace binary variable x_{ij} that can reduce $(m - 1)n$ variables, since the fire engines are supposed to be identical [29]. Secondly, we observe that parameter LB in P_0 is redundant because it can be implicitly ensured by the left side of (5). Thus we can remove such bound constraint of (4). Thirdly, the right side of (5) can also be removed since it can be ensured through limiting the total number of fire engines as done in (4). With these above observations, the following equivalent improved and simplified formulation can be derived.

$$P : f_1 : \min \sum_{i=1}^n t_{Ei}$$

$$f_2 : \min \sum_{i=1}^n y_i \quad (8)$$

$$\text{s.t. } t_{Ei} = v_{si} T_{Ai} / (y_i v_f - v_{si}), \quad \forall i \in V \quad (9)$$

$$\sum_{i=1}^n y_i \leq m \quad (10)$$

$$L_i \leq y_i, \quad \forall i \in V \quad (11)$$

$$t_{Ei} \geq 0, \quad y_i \in \mathbb{N}^+, \quad \forall i \in V \quad (12)$$

It can be seen that model P employs n integer variables and $2n + 1$ constraints. Compared with model P_0 , model P reduces $(m - 1)n$ integer variables and $n + 1$ constraints. Comparison results reported later show that P is much more efficient to solve than P_0 .

We note that both P_0 and P involve integer variables and are non-linear due to the non-linear constraints (3) and (9). It is difficult to solve them by commercial optimization software such as LINGO to be reported later. In order to solve P , its property analysis is performed in the following section.

IV. PROPERTIES OF THE MODEL

As a schedule with fewer rescue vehicles dispatched to extinguish fires may lead to a longer fire-extinguishing time, the two objectives of the proposed model (i.e., fire-extinguishing time and number of dispatched fire engines) are conflicting. As such, there does not exist a unique solution to simultaneously optimize the above two objectives but a set of Pareto-optimal solutions. For simplicity, let \mathcal{Y} denote the feasible solution set of P . Before proceeding, we first give the following definitions related to multi-objective optimization [40].

Definition 1: A schedule $\mathbf{y}^* \in \mathcal{Y}$ is Pareto-optimal if and only if there exist no schedule $\mathbf{y} \in \mathcal{Y}$, such that $f_1(\mathbf{y}) \leq f_1(\mathbf{y}^*)$ and $f_2(\mathbf{y}) \leq f_2(\mathbf{y}^*)$ where at least one inequality is strict.

Definition 2: A solution (point) $(f_1(\mathbf{y}), f_2(\mathbf{y}))$, $\mathbf{y} \in \mathcal{Y}$, in the objective space is non-dominated if \mathbf{y} is Pareto-optimal.

The image of all non-dominated points is called the non-dominated set or the *Pareto front*. To exactly solve a multi-objective optimization problem means finding its Pareto front.

To derive the relation between the two objectives of model P , we first define the feasible (expected) number of dispatched fire engines as a parameter R satisfying $\sum_{i=1}^n L_i \leq R \leq m$ where m and L_i is the total available fire engine count and the minimum number of fire engines needed to fight fire point i , respectively. Then, model P can be transformed into P_R for a given R .

$$P_R : \quad \min \quad \sum_{i \in V} t_{Ei}$$

$$\text{s.t.} \quad \sum_{i \in V} y_i \leq R \quad (13)$$

and constraints (9), (11) and (12)

P_R is to minimize the total fire-extinguishing time for a given number of rescue vehicles R . For the convenience of analysis, let $\mathbf{y}_R^* = (y_{1,R}^*, y_{2,R}^*, \dots, y_{n,R}^*)$, t_{Ei}^* , and $T_R^* = \sum_{i=1}^n t_{Ei}^*$ denote an optimal number of fire engines dispatched to fire points $i = 1, 2, \dots, n$, the time of extinguishing fire point i , and the optimal objective value, respectively. For any problem P_R , we have the following results.

Proposition 1: In an optimal solution of P_R , the number of dispatched fire engines is exactly equal to R , i.e., $\sum_{i=1}^n y_{i,R}^* = R$.

Proof: By contradiction. Suppose that $\sum_{i=1}^n y_{i,R}^* < R$. Let $\Delta = R - \sum_{i=1}^n y_{i,R}^*$. Since $R \leq m$, then Δ rescue vehicles can be additionally added to schedule $\mathbf{y}_R^* = (y_{1,R}^*, y_{2,R}^*, \dots, y_{n,R}^*)$ without violating any constraint of P_R . Thus, from the current schedule, we can construct a new feasible schedule $(y_{1,R}^* + \Delta_1, y_{2,R}^* + \Delta_2, \dots, y_{n,R}^* + \Delta_n)$ with $\sum_{i=1}^n \Delta_i = \Delta, \forall \Delta_i \geq 0$. According to Eq. (9), for any $i \in V$, if $\Delta_i > 0$, then the value of t_{Ei} in the new schedule must be less than that of the original one. This means that we have found a new schedule with less fire-extinguishing time and R rescue vehicles are dispatched to fight fires. This contradicts the fact that \mathbf{y}_R^* is an optimal schedule. Consequently, $\sum_{i=1}^n y_{i,R}^* = R$ always holds. ■

Proposition 2: In an optimal solution, the fire-extinguishing time T_R^* is a strictly decreasing function of the number of dispatched fire engines R , i.e., for any two values R_1 and R_2 , if $R_1 < R_2$, then $T_{R_1}^* > T_{R_2}^*$ always holds.

Proof: We firstly show that if $R_1 < R_2$, then $T_{R_1}^* \geq T_{R_2}^*$ always holds. For any given two values of R , i.e., R_1 and R_2 , if $R_1 < R_2$, then constraint (13) of P_{R_1} is tighter than that of P_{R_2} and all the other constraints of both problems are the same. Thus, $T_{R_1}^*$ is not less than $T_{R_2}^*$, i.e., $T_{R_1}^* \geq T_{R_2}^*$. Then, we show $T_{R_1}^* = T_{R_2}^*$ is impossible by contradiction. Assume that $T_{R_1}^* = T_{R_2}^*$. Then, the optimal schedule of P_{R_1} is also an optimal one of P_{R_2} . By Proposition 1, we have $\sum_{i=1}^n y_{i,R_1}^* = R_1 = R_2$, which contradicts the fact that $R_1 < R_2$. This indicates that it is impossible $T_{R_1}^* = T_{R_2}^*$. Consequently, if $R_1 < R_2$, then $T_{R_1}^* > T_{R_2}^*$ always holds. ■

By Propositions 1 and 2, we have the following corollary.

Corollary 1: A Pareto-optimal solution can always be obtained by exactly solving P_R for a given R .

Furthermore, as $\sum_{i=1}^n L_i \leq R \leq m$, the lower and upper bounds on the number of dispatched fire engines (i.e., f_2) over the Pareto front are $\sum_{i=1}^n L_i$ and m , respectively. We can also have the following theorem.

Theorem 1: There exist $m - \sum_{i=1}^n L_i + 1$ non-dominated points in the Pareto front of model P and they can be obtained by separately solving P_R with R being each value in the set $\Omega = \{\sum_{i=1}^n L_i, \sum_{i=1}^n L_i + 1, \dots, m\}$.

Proof: By Propositions 1 and 2 and the fact that the value of f_2 must be an integer. ■

In the following, new optimization approaches based on the derived properties are developed for the considered problem.

V. SOLUTION APPROACH

As indicated by Theorem 1, the key to solving P becomes how to efficiently solve P_R . Since P_R is a non-linear integer program because of constraint (9), it is difficult to be tackled by a commercial solver such as LINGO. Therefore, it is necessary to develop solution approaches for it. In what follows, we design an exact dynamic programming algorithm and a fast greedy heuristic method for solving P_R . For the former, the Pareto front is ensured at the expense of computational time, while for the latter, an approximated Pareto front is derived but the computational time can be greatly saved.

A. Dynamic Programming Algorithm

To solve problem P_R exactly, we develop in this subsection a dynamic programming (DP) algorithm based on the properties of the problem.

From P_R , we can find that for each fire point i , a minimum number of rescue vehicles L_i must be dispatched such that it can be extinguished (see constraint (11)). Consequently, for P_R , the key problem becomes how to dispatch the remaining $R - \sum_{i=1}^n L_i$ vehicles such that the total fire-extinguishing time is minimized. Before proceeding, we first define a new variable as follows.

z_i : additional number of fire engines except a minimum value L_i dispatched to fire point i , $i \in V$. Note that $y_i = z_i + L_i$.

With the new definition and Proposition 1, P_R can be rewritten as follows:

$$\min \quad \sum_{i=1}^n v_{si} T_{Ai} / ((z_i + L_i)v_f - v_{si})$$

$$\text{s.t.} \quad \sum_{i=1}^n (z_i + L_i) = R \quad (14)$$

$$z_i \geq 0, \quad \forall i \in V \quad (15)$$

Let $T_i = v_{si} T_{Ai} / (L_i v_f - v_{si})$, which denotes the fire-extinguishing time of fire point i with L_i fire engines. Then,

the above program is equivalent to the following one:

$$P'_R : \max \sum_{i=1}^n (T_i - v_{si} T_{Ai} / ((z_i + L_i) v_f - v_{si})) \quad (16)$$

$$\text{s.t. } \sum_{i=1}^n z_i = R - \sum_{i=1}^n L_i$$

$$z_i \geq 0, \quad \forall i \in V \quad (17)$$

To solve P'_R , a DP algorithm is proposed as follows. For the convenience of analysis, let $r_i(z_i) = T_i - v_{si} T_{Ai} / ((z_i + L_i) v_f - v_{si})$; and for fire point i , let $F_i(z)$ denote the maximum objective value among all vehicle schedules that contain fire point set $\{i, i+1, \dots, n\}$, given z fire engines to be dispatched. By using the above notations, the optimal solution of P'_R can be computed in a recursive way

$$\begin{cases} F_n(z) = r_n(z) \\ F_i(z) = \max_{0 \leq z_i \leq z} \{r_i(z_i) + F_{i+1}(z - z_i)\}, \\ \forall 1 \leq i < n \end{cases} \quad (18)$$

The optimal objective value of P_R can be obtained by $F_1(R - \sum_{i=1}^n L_i)$ in $\mathcal{O}(nm^2)$ time in the worst case. Thus, the optimal objective value of P_R is $\sum_{i=1}^n T_i - F_1(R - \sum_{i=1}^n L_i)$. By Theorem 1, there are $m - \sum_{i=1}^n L_i$ (at most m) single-objective problems P_R 's need to be solved, thus the whole solution of model P with the DP algorithm requires $\mathcal{O}(nm^3)$ time in the worst case, which is pseudo polynomial.

B. Greedy Heuristic Algorithm

In order to more efficiently solve P , especially for large-size problems, we also propose a greedy heuristic method for P_R whose core idea is to first ensure the minimum requirements of fire engines for each fire point and then to assign remaining available vehicles one by one to the fire point with the largest reduction in the total fire-extinguishing time. Its procedure is outlined in Algorithm 1.

Algorithm 1 A Greedy Heuristic Algorithm for P_R

- 1: Let $y_i = L_i, \forall i \in V$ and obtain $\mathbf{y} = (L_1, L_2, \dots, L_n)$
 - 2: Calculate $f_1(\mathbf{y})$ and let R_v be $R - \sum_{i=1}^n L_i$, which denotes the remaining fire engine count expect the minimum requirement.
 - 3: **while** ($R_v > 0$)
 - 4: Let $Rt_i = 0, \forall i \in V$
 - 5: **for** $i = 1; i \leq n$ **do**
 - 6: Let $y_i = y_i + 1$ and obtain solution $\mathbf{y}' = (y_1, y_2, \dots, y_n)$
 - 7: Calculate $f_1(\mathbf{y}')$ and let $Rt_i = f_1(\mathbf{y}) - f_1(\mathbf{y}')$
 - 8: $y_i = y_i - 1$
 - 9: **end for**
 - 10: Select the fire point $i = \arg \max Rt_i$
 - 11: $y_i = y_i + 1$ and $R_v = R_v - 1$
 - 12: **end while**
 - 13: Output the solution $\mathbf{y} = (y_1, y_2, \dots, y_n)$ and the corresponding objective value $f_1(\mathbf{y})$.
-

In Algorithm 1, step 1 is to assign the minimum number of fire engines to each fire point; step 2 calculates the

corresponding fire-extinguishing time and the rest fire engine count; steps 3-12 are to assign the remaining fire engines one by one where steps 5-9 are to determine which fire point to be assigned at each iteration; and step 13 outputs the solution and its corresponding objective value. Clearly, the most time-consuming parts of Algorithm 1 are steps 3-12. Since $0 \leq R_i \leq m - n$, then at most $m - n$ additional rescue vehicles are to be assigned. Then, for each assignment (i.e., steps 5-9), we need to check at most n fire points. Thus, Algorithm 1 requires $\mathcal{O}(nm)$ time in the worst case.

As Algorithm 1 cannot guarantee the solution optimality, an approximated Pareto front of P is found by solving all P_R 's using Algorithm 1. Similarly, since at most m problems need to be solved, the whole solution of model P using Algorithm 1 for each P_R requires $\mathcal{O}(nm^2)$ time in the worst case, which enjoys lower time complexity than the DP algorithm.

VI. EXTENSION TO MULTI-DEPOT CASE

In practical situations, it might be possible that a single fire-fighting service point cannot meet all the fire-fighting needs, i.e., the capability of the fire-fighting service station is not enough. This motivates us to deploy multiple service points to collaboratively extinguish forest fires. This subsection expands model P to a multi-depot case, thus formulating a multi-depot fire engine scheduling problem for the first time.

To formulate this problem, one of the key problems is to calculate the fire-extinguishing time. For the sake of convenience, let K, m_k, T_{Aik} and y_{ik} denote the set of all service points, the number of fire engines involved in the k -th depot, the travel time of fire engines from the k -th service point (depot) to fire point i , and the number of fire engines dispatched to fire point i from the k -th service point, respectively. Note that y_{ik} is the decision variable. Similarly, T_{Aik} can be calculated as d_{ki}/v_{ki} , where d_{ki} and v_{ki} are the distance between depot k and fire point i and the average driving speed of fire engines, respectively [29]. Let t_{Ri} denote the whole rescue time for fire point $i \in V$. Then, according to [29], the mathematic relation among v_{si}, v_f and T_{Ri} can be described as follows:

$$t_{Ri} \cdot v_{si} = \sum_{k \in K} (t_{Ri} - T_{Aik}) \cdot v_f y_{ik}, \quad \forall i \in V,$$

where the left (resp. right) side represents the fire-fighting (resp. fire spread) area of the i -th fire point and $t_{Ri} - T_{Aik}$ denotes the fire-fighting time of the fire engines from the k -th depot. Thus, the whole rescue time for fire point i is:

$$t_{Ri} = \frac{\sum_{k \in K} v_f T_{Aik} y_{ik}}{\sum_{k \in K} v_f y_{ik} - v_{si}}, \quad \forall i \in V.$$

Remark 2: For any fire point $i \in V$, $\sum_{k \in K} y_{ik} > v_{si}/v_f$ should be ensured to derive a feasible scheme.

Thus, the multi-depot fire engine scheduling problem can be formulated as the following program P_M .

$$P_M : f_1 : \min \sum_{i \in V} t_{Ri} \quad (19)$$

$$f_2 : \min \sum_{i \in V} \sum_{k \in K} y_{ik} \quad (20)$$

$$\text{s.t. } t_{Ri} = \frac{\sum_{k \in K} T_{Aik} v_f y_{ik}}{\sum_{k \in K} v_f y_{ik} - v_{si}}, \quad \forall i \in V \quad (21)$$

$$\sum_{i=1}^n y_{ik} \leq m_k, \quad \forall k \in K \quad (22)$$

$$\sum_{k \in K} y_{ik} \geq L_i, \quad \forall i \in V \quad (23)$$

$$t_{Ri} \geq 0, y_{ik} \in \{0\} \cup \mathbb{N}^+, \quad \forall i \in V, \forall k \in K \quad (24)$$

where constraint (22) ensures that the dispatched fire engines from the k -th depot should not exceed its capacity and constraint (23) ensures that the number of fire engines dispatched to fire point i should exceed the given lower bound value $L_i \geq \lceil v_{si}/v_f \rceil$ based on Remark 2. As P_M has its own specialities, the proposed DP and heuristic for P cannot be used to solve it despite our best efforts. To solve P_M , the ε -constraint method [41] is adapted to solve it. Its basic idea is to transform the bi-objective model to a single-objective one by optimizing a preferred objective while restricting the other to a given bound ε . For more details on the ε -constraint method, please refer to [40] and [41]. For P_M , we select the first objective f_1 as the preferred one. Then, P_M can be transformed to $P_M(\varepsilon)$ shown as follows:

$$P_M(\varepsilon) : \min \sum_{i \in V} t_{Ri} \quad (25)$$

$$\text{s.t. } \sum_{i \in V} \sum_{k \in K} y_{ik} \leq \varepsilon$$

and constraints (21) – (24)

where ε is an upper bound of f_2 . It is not hard to find that the range of ε would be $[\sum_{i=1}^n L_i, \sum_{k \in K} m_k]$. Given a value of ε , one non-dominated point may be found by exactly solving $P_M(\varepsilon)$. The Pareto front can be obtained by varying the value of ε within its range. Its procedure can be described in Algorithm 2.

Algorithm 2 The ε -Constraint Method for P_M

- 1: Transform P_M to $P_M(\varepsilon)$
 - 2: Calculate the range of f_2 , i.e., $[\sum_{i=1}^n L_i, \sum_{k \in K} m_k]$.
 - 3: **for** $\varepsilon \leftarrow \sum_{i=1}^n L_i$ to $\sum_{k \in K} m_k$ **do**
 - 4: Solve $P_M(\varepsilon)$ by using LINGO and obtain the solution and its corresponding objective vector.
 - 5: **end for**
 - 6: Remove dominated points and output the final solution set.
-

VII. COMPUTATIONAL EXPERIMENTS

In this section, benchmark instances in [29] and larger-size randomly generated instances are tested to evaluate the

TABLE II
MODEL COMPARISON RESULTS FOR SOLVING INSTANCES IN [29]

n	m	P_0			P		
		<i>Var</i>	<i>Cons</i>	CPU time (ms)	<i>Var</i>	<i>Cons</i>	CPU time (ms)
7	40	287	24	-	14	16	9000
10	48	490	33	-	20	22	22000
15	72	1095	48	-	30	32	105000
20	90	1820	63	-	40	42	373000
50	250	12550	153	-	100	102	-

TABLE III
COMPARISON RESULTS FOR A REAL-LIFE INSTANCE

n	m	Generated Pareto solutions			CPU time (ms)			
		TRZ	Heuristic	DP/LINGO	TRZ ^a	Heuristic	DP	LINGO
7	40	(6.08, 40)	(6.06, 40)	(6.06, 40)	284	< 1	9	9000
		(6.47, 39)	(6.47, 39)	(6.47, 39)				
		(7.60, 37)	(6.97, 38)	(6.97, 38)				
		(8.57, 36)	(7.60, 37)	(7.60, 37)				
		(9.56, 35)	(8.57, 36)	(8.57, 36)				
		(10.54, 34)	(9.56, 35)	(9.56, 35)				
		(12.38, 33)	(10.54, 34)	(10.54, 34)				
		(15.55, 32)	(12.38, 33)	(12.38, 33)				
		(18.61, 31)	(15.45, 32)	(15.45, 32)				
		(24.32, 30)	(18.61, 31)	(18.61, 31)				
		(39.60, 29)	(24.32, 30)	(24.32, 30)				
			(39.60, 29)	(39.60, 29)				

^a Result on 2.53GHz computer with 4 GB RAM.

performance of the proposed model and algorithms. Both models P_0 and P are solved by using commercial optimization software LINGO (V14.0) and the proposed algorithms are coded in C++ with Visual Studio 2010. All the computational experiments are conducted on a PC with 2.5 GHz and 2.95 GB RAM under windows 10. The computational time (CPU time) spent by each method is limited to 10 min (600000 ms) for facilitating fast decision-making in actual emergencies.

A. Model Comparison

To evaluate the performance of the proposed model, we test the benchmark instances in [29] by directly using P and P_0 solved by LINGO. For the sake of convenience, let Var and $Cons$ denote the number of variables and constraints, respectively. Table II presents their comparison results.

From Table II, we can see that our model P employs much fewer variables and constraints than the existing one P_0 . P can exactly solve four of the five instances, while P_0 fails to solve any of them within 10 minutes. This indicates that our model is much more efficient than the existing one, because the proposed model is more compact and tighter than the one in [29].

B. Real-Life Instance

We then test the real-life instance from the emergency scheduling of forest fires in Huzhong region located in Mt. Daxing'anling of Heilongjiang Province, China [29]. In this instance, seven fire points are involved and 40 rescue vehicles are available for fighting fires. For more details, interested readers can refer to [29]. Table III reports the comparison results between the method proposed by Tian *et al.* [29]

TABLE IV
COMPARISON RESULTS FOR BENCHMARK INSTANCES

n	m	$ \mathcal{F} $	Solution quality			CPU time (ms)			
			\bar{R}_{TRZ} (%)	\bar{R}_{H} (%)	\bar{R}_{DP} (%)	TRZ ^a	Heuristic	DP	LINGO
10	48	12	58.33	100	100	525	< 1	12	22000
15	72	14	35.71	100	100	4704	< 1	14	105000
20	90	18	27.78	100	100	6091	< 1	20	373000
50	250	64	0.00	100	100	51526	6	82	-

^a Result on 2.53GHz computer with 4 GB RAM.

(called TRZ for short) and the proposed algorithms, in which columns 3 to 5 list the found Pareto points. For each point, the first value represents the fire-extinguishing time, while the second one denotes the number of dispatched fire engines.

It can be seen in Table III that each method obtains a set of Pareto solutions. To better understand different possible decisions, take the first and last points of TRZ as an example, the results mean that it will take 6.08 h to extinguish all fire points with 40 fire engines for the former, while 39.6 h will be consumed with 29 fire engines for the latter. Decision-makers can select a preferred solution from the Pareto front. Furthermore, from Table III we can conclude that:

- 1) In terms of solution quality, the number of Pareto-optimal solutions obtained by our algorithms is more than that by TRZ. Moreover, three new non-dominated points (i.e., (6.06, 40), (6.97, 38) and (15.45, 32)) are found by our algorithms. The heuristic can also obtain the Pareto front as the DP algorithm. This shows that the proposed heuristic is able to obtain high-quality solutions.
- 2) In terms of computational time, both the DP and heuristic algorithms are much more efficient in solving this real-life instance than TRZ and LINGO. Moreover, it can be observed that the heuristic performs better than the DP algorithm in terms of computational efficiency.
- 3) The fire-extinguishing time decreases as the number of dispatched fire engines increases due to the strictly decreasing relationship between them, which matches the conclusion made in Proposition 2.

C. Benchmark Instances

To further show the performance of the proposed methods, we conduct numerical comparison tests on a set of benchmark instances in [29] with a different number of fire points, i.e., 10, 15, 20, and 50. Let \bar{R}_{TRZ} , \bar{R}_{H} , and \bar{R}_{DP} denote the coverage rate of non-dominated points found by TRZ, the proposed DP and heuristic algorithms to the exact Pareto front, respectively. This indicator measures the solution quality. As the DP algorithm can obtain all the non-dominated points over the Pareto front, \bar{R}_{DP} is 100%. Note that both \bar{R}_{TRZ} and \bar{R}_{H} are equal to or less than 100%, as their obtained solutions are not guaranteed to be Pareto-optimal. For brevity, let $|\mathcal{F}|$ denote the number of obtained non-dominated points in the Pareto front. With the above notations, the comparison results are summarized in Table IV.

It can be seen from Table IV that the values of \bar{R}_{H} and \bar{R}_{DP} are 100% for all the instances, while that of \bar{R}_{TRZ} varies from

TABLE V
COMPARISON RESULTS FOR BENCHMARK INSTANCES
WITH DIFFERENT m VALUES

n	m	$ \mathcal{F} $	Solution quality			CPU time (ms)		
			\bar{R}_{TRZ} (%)	\bar{R}_{H} (%)	\bar{R}_{DP} (%)	TRZ ^a	Heuristic	DP
10	40	4	75.00	100	100	96050	< 1	6
	42	6	66.67	100	100	8283	< 1	5
	45	9	33.33	100	100	1021	< 1	9
15	68	10	40.00	100	100	153578	< 1	9
	69	11	54.55	100	100	49972	< 1	10
	70	12	50.00	100	100	22369	< 1	10
20	86	14	28.57	100	100	136548	< 1	15
	87	15	12.50	100	100	45413	< 1	16
	89	17	17.65	100	100	15788	< 1	17

^a Result on 2.53GHz computer with 4 GB RAM.

0 to 58.33%. This implies that the proposed greedy heuristic can also obtain the Pareto front just like the DP algorithm, while TRZ fails to achieve it. Moreover, we can see that the value of \bar{R}_{TRZ} decreases as the problem size increases. It is worthwhile to point out that the value of \bar{R}_{TRZ} is 0 for the largest-size instance with $n = 50$ and $m = 250$, which implies that TRZ fails to derive Pareto-optimal solutions for this instance; whereas the value of \bar{R}_{H} does not decrease with the problem size. The results show that both the proposed DP and heuristic methods outperform TRZ in terms of solution quality. Furthermore, it can be seen that our algorithms can obtain the Pareto front for these instances in short time, i.e., the computational time spent by the proposed DP and heuristic algorithm slightly vary from less than 1 to 6 ms and from 12 to 82 ms, respectively, while the computational time of TRZ and LINGO varies from 525 to 51526 ms and 22000 to over 600000 ms, respectively, which sharply increases with the problem size. This shows that our algorithms are more efficient than TRZ and LINGO. In addition, the heuristic method achieves the best performance in terms of computational time.

In practical emergency scheduling, it is necessary to take into account the impact of the quantity of available rescue resources, i.e., the number of available fire engines to our problem. We also test the benchmark instances under different settings of the number of available fire engines (i.e., m). As it is very time-consuming to solve the model with LINGO, we only compare our algorithms with TRZ. Their comparison results are reported in Table V.

From Table V, we can see that the proposed DP and heuristic can obtain all non-dominated points (i.e., Pareto front), while TRZ can at most find 75% non-dominated points in the Pareto front. In addition, for each given n , the value of \bar{R}_{TRZ} decreases with the value of m , while the coverage rates of our algorithms keep the same. The results show that the proposed algorithms are more effective than TRZ in solving these instances. Moreover, it can be seen that for a fixed n , the number of non-dominated points $|\mathcal{F}|$ increases with m , which implies the complexity of the problem increases with m . In terms of computational time, we can see that the proposed algorithms consume far less time than TRZ and also, the heuristic algorithm is the fastest one. The results again confirm

TABLE VI

COMPUTATIONAL RESULTS FOR LARGER-SIZE RANDOM INSTANCES

n	m	$ \mathcal{F} $	\bar{R}_H (%)	T_{DP} (ms)	T_H (ms)
100	500	185.0	100	136	34
200	1000	269.0	100	951	250
300	1500	541.6	100	2445	808
400	2000	544.4	100	5545	1904
500	2500	666.6	100	9448	3386
600	3000	782.6	100	15921	6107
700	3500	927.0	100	25526	9853
800	4000	1048.8	100	37484	14529
900	4500	1195.6	100	54693	21047
1000	5000	1316.8	100	79938	28953

the efficiency of the proposed algorithms.

D. Larger-Size Instances

To further evaluate the performance of the proposed methods for solving larger-size problems, we perform numerical tests with more fire points varying from 100 to 1000. Note that although these large-size instances might not happen in real life, we conduct such experiments to further demonstrate the performance of the proposed algorithms. For each problem size, five instances are tested. All instances are randomly generated based on the way in [29], which is described as follows. The distance among points in the network d_{ij} is uniformly and randomly generated in the interval [50, 100]. The driving speed of the rescue vehicle from vertices i and j , $\forall i, j \in V'$, v_{ij} is set to be 54 km/h. The fire-extinguishing speed of a fire engine v_f is set as 2.5 m/min. The fire spread speed v_{si} is uniformly and randomly generated in the interval [2, 6] m/min. As the TRZ consumes too much computational time for large-size problems, we only compare our two algorithms in this subsection. For the sake of convenience, let T_{DP} and T_H denote the computational time by the proposed DP and heuristic algorithms, respectively. The computational results are summarized in Table VI. Note that each value in the table is the average value of five instances.

From Table V, we can conclude that:

- 1) The value of \bar{R}_H is 100% for all problem sets, which implies that the proposed heuristic obtains the Pareto fronts of all instances.
- 2) The proposed algorithms obtain the Pareto fronts with up to 1000 fire points and 5000 fire engines within 79938 ms (less than 80 s). The results show that our algorithms are effective and efficient in solving the large-size instances. Also, it can be observed that T_H is less than T_{DP} over all problem sets, and T_{DP} rapidly increases with the problem size, whereas T_H increases in a much slower pace than DP does. This indicates that the heuristic method is more efficient than the DP algorithm. Besides, the results in this table suggest that $|\mathcal{F}|$ has an obvious increasing trend as the number of available fire engines (i.e., m) increases. This can be explained in part by Theorem 1.

E. Results of Multi-Depot Fire Engine Scheduling

We also test the real-life instance by considering multiple depots to evaluate the proposed model P_M . Three cases with

TABLE VII

COMPUTATIONAL RESULTS FOR THE REAL-LIFE INSTANCE WITH MULTIPLE DEPOTS

n	m	$ K $	Var	$Cons$	$ \mathcal{F} $	CPU time (s)
7	40	2	21	18	12	121
7	40	3	28	19	12	289
7	40	4	35	20	12	365

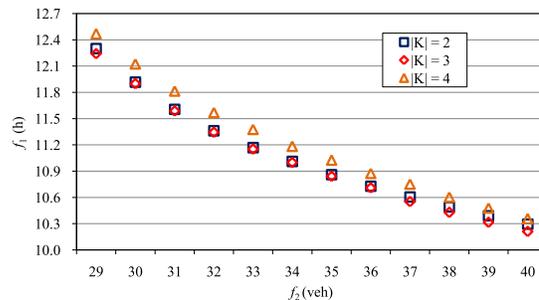


Fig. 2. The Pareto fronts of the real-case instance with multiple depots.

$|K|=2, 3$, and 4 are considered, where $|K|$ denotes the number of fire service points (i.e., depots). According to [29], the distances between the newly added depots and fire points are uniformly and randomly generated in [50, 100]. The total number of fire engines for each case is set the same as the work [29]. For $|K|=2, 3$, and 4 , we set $m_1 = m_2 = 20$; $m_1 = 15, m_2 = 13, m_3 = 12$; and $m_1 = m_2 = m_3 = m_4 = 10$, respectively. The other related data is kept unchanged. Table VII and Fig. 2 present the computational results.

From Table VII, we can see that the proposed algorithm (i.e., Algorithm 2) is able to obtain the Pareto front of the real-life instance with multiple depots within the given time. Moreover, the computational time increases as the number of depots increases, which is mainly because the number of variables and constraints increases with $|K|$. This indicates that the complexity of the problem increases with the number of depots. Besides, it can be observed that the number of non-dominated points found for each case is the same, which means that Algorithm 2 can find a non-dominated point at each iteration for each case. This implies that a non-dominated solution exists for each value of f_2 .

From Fig. 2, we can see a clear trade-off between the total rescue time and the number of fire engines dispatched for each $|K|$. The Pareto fronts derived can provide useful information for decision-makers to achieve a balance between rescue time and dispatched resources. In particular, decision-makers can select their most preferred solution from the solution set. For instance, if they prefer a solution with the least emergency response time, then the last solution can be chosen.

VIII. CONCLUSIONS

This work has studied a bi-objective emergency scheduling problem for forest fires to simultaneously minimize the total fire-extinguishing time and the total dispatched fire engines. An improved bi-objective integer program containing less variables and constraints has been proposed for the studied problem. Experimental results indicate that the improved model

significantly outperforms the existing one [29]. Based on the property analysis results of the problem, an exact dynamic programming algorithm and a fast greedy heuristic method are devised. Computational results for a real-life instance and benchmark instances show that the proposed DP and heuristic algorithms significantly outperform the existing method [29] in terms of both solution quality and computational efficiency. The Pareto fronts of all the benchmark instances are obtained. Larger-size randomly generated instances are also tested to evaluate the performance of the proposed algorithms. Experimental results indicate that large-size problems with up to 1000 fire points and 5000 fire engines can be exactly solved within 80s on a personal computer. This demonstrates the effectiveness and efficiency of the proposed algorithms in solving large-scale rescue vehicle scheduling problems to deal with forest fires. Moreover, this study has also constructed a multi-depot fire engine scheduling model for the multi-depot case and the ε -constraint method is adapted to solve it. The trade-off between the two objectives to be optimized is clearly illustrated.

Although the proposed model is much more efficient than the existing one, it could not address the case with heterogeneous fire engines. One future research direction is to construct new and effective models for the heterogeneous fire engine scheduling. Other future research following this work may include extending the studied problem to more general cases. For example, in practice, the travel time from the fire service station to fire points can dynamically change with the roadway conditions. Extending the proposed model to a stochastic or time-dependent one by considering stochastic or varying travel time would be an important direction. Besides, it can be found that the multi-depot fire engine scheduling problem is much more difficult to solve. New effective and efficient approaches [42]–[51] should be designed for addressing such optimization problems.

ACKNOWLEDGEMENT

We would like to thank the associate editor and four anonymous reviewers for their constructive comments and suggestions, which greatly improved the quality of this paper. We are also grateful to Dr. Guangdong Tian for providing his test instances, enabling us to compare the algorithms.

REFERENCES

- [1] S. A. Ardekani and A. G. Hobeika, "Logistics problems in the aftermath of the 1985 Mexico city earthquake," *Transp. Quart.*, vol. 42, no. 1, pp. 107–124, 1988.
- [2] Y.-C. Chiu and H. Zheng, "Real-time mobilization decisions for multi-priority emergency response resources and evacuation groups: Model formulation and solution," *Transp. Res. E, Logistics Transp. Rev.*, vol. 43, no. 6, pp. 710–736, 2007.
- [3] D. C. Whybark, "Issues in managing disaster relief inventories," *Int. J. Prod. Econ.*, vol. 108, no. 1, pp. 228–235, 2007.
- [4] P. S. Georgiadou, I. A. Papazoglou, C. T. Kiranoudis, and N. C. Markatos, "Multi-objective evolutionary emergency response optimization for major accidents," *J. Hazardous Mater.*, vol. 178, nos. 1–3, pp. 792–803, 2010.
- [5] J. W. Wang, W. H. Ip, and W. J. Zhang, "An integrated road construction and resource planning approach to the evacuation of victims from single source to multiple destinations," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 277–289, Jun. 2010.

- [6] R. P. Knott, "Vehicle scheduling for emergency relief management: A knowledge-based approach," *Disasters*, vol. 12, no. 4, pp. 285–293, 1988.
- [7] S. Oh and A. Haghani, "Testing and evaluating of a multi-commodity network flow model for disaster relief management," *J. Adv. Transp.*, vol. 31, no. 2, pp. 249–282, 1996.
- [8] A. M. Campbell, D. Vandenbussche, and W. Hermann, "Routing for relief efforts," *Transp. Sci.*, vol. 42, no. 2, pp. 127–145, 2008.
- [9] H. O. Mete and Z. B. Zabinsky, "Stochastic optimization of medical supply location and distribution in disaster management," *Int. J. Prod. Econ.*, vol. 126, no. 1, pp. 76–84, 2010.
- [10] S. Yan, J. C. Chu, and Y.-L. Shih, "Optimal scheduling for highway emergency repairs under large-scale supply-demand perturbations," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2378–2393, Dec. 2014.
- [11] S. Misra and S. Sarkar, "Priority-based time-slot allocation in wireless body area networks during medical emergency situations: An evolutionary game-theoretic perspective," *IEEE J. Biomed. Health Informat.*, vol. 19, no. 2, pp. 541–548, Mar. 2015.
- [12] S. A. Ardekani, "Transportation operations following the 1989 Loma Prieta earthquake," *Transp. Quart.*, vol. 46, no. 2, pp. 219–233, 1992.
- [13] A. Haghani, "Formulation and solution of a multi-commodity, multi-modal network flow model for disaster relief operations," *Transp. Res. A, Policy Pract.*, vol. 30, no. 3, pp. 231–250, 1996.
- [14] L. Özdamar, E. Ekinci, and B. Küçükyazici, "Emergency logistics planning in natural disasters," *Ann. Oper. Res.*, vol. 129, no. 1, pp. 217–245, 2004.
- [15] B. Balcik and B. M. Beamon, "Facility location in humanitarian relief," *Int. J. Logistics*, vol. 11, no. 2, pp. 101–121, 2008.
- [16] A. Y. Saber and G. K. Venayagamoorthy, "Resource scheduling under uncertainty in a smart grid with renewables and plug-in vehicles," *IEEE Syst. J.*, vol. 6, no. 1, pp. 103–109, Mar. 2012.
- [17] J.-B. Sheu, "An emergency logistics distribution approach for quick response to urgent relief demand in disasters," *Transp. Res. E, Logistics Transp. Rev.*, vol. 43, no. 6, pp. 687–709, 2007.
- [18] Z.-H. Hu, "A container multimodal transportation scheduling approach based on immune affinity model for emergency relief," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2632–2639, 2011.
- [19] Y.-J. Zheng and H.-F. Ling, "Emergency transportation planning in disaster relief supply chain management: A cooperative fuzzy optimization approach," *Soft Comput.*, vol. 17, no. 7, pp. 1301–1314, 2013.
- [20] W. Yi and L. Özdamar, "A dynamic logistics coordination model for evacuation and support in disaster response activities," *Eur. J. Oper. Res.*, vol. 179, no. 3, pp. 1177–1193, 2007.
- [21] A. Haghani, Q. Tian, and H. Hu, "Simulation model for real-time emergency vehicle dispatching and routing," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1882, pp. 176–183, Jan. 2004.
- [22] H. Fu, A. J. Pel, and S. P. Hoogendoorn, "Optimization of evacuation traffic management with intersection control constraints," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 376–386, Feb. 2015.
- [23] W. Yi and A. Kumar, "Ant colony optimization for disaster relief operations," *Transp. Res. E, Logistics Transp. Rev.*, vol. 43, no. 6, pp. 660–672, 2007.
- [24] N. C. Simpson and P. G. Hancock, "Fifty years of operational research and emergency response," *J. Oper. Res. Soc.*, vol. 60, pp. S126–S139, May 2009.
- [25] L. E. de la Torre, I. S. Dolinskaya, and K. R. Smilowitz, "Disaster relief routing: Integrating research and practice," *Soc.-Econ. Planning Sci.*, vol. 46, no. 1, pp. 88–97, 2012.
- [26] A. M. Caunhye, X. Nie, and S. Pokharel, "Optimization models in emergency logistics: A literature review," *Soc.-Econ. Planning Sci.*, vol. 46, no. 1, pp. 4–13, 2012.
- [27] M. Wei and K. Wang, "The usage state and the prospect of forest fire extinguish equipments," *Forestry Mach. Woodw. Equip.*, vol. 7, no. 34, pp. 11–14, 2006.
- [28] Y. Zhang, "The Standardized Operational Workflow of Forest Fire Fighting in Daxing'Anling Area," accessed on Apr. 2017. [Online]. Available: <http://www.huma.gov.cn/content/detail/572c53167f8b9a216a53ec98.html>
- [29] G. Tian, Y. Ren, and M. Zhou, "Dual-objective scheduling of rescue vehicles to distinguish forest fires via differential evolution and particle swarm optimization combined algorithm," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 11, pp. 3009–3021, Nov. 2016.
- [30] Z. Wang, "The measurement method of the wildfire initial spread rate," *Mountain Res.*, vol. 1, pp. 42–51, 1983.
- [31] A. Haghani and H. Qiao, "Decision support system for snow emergency vehicle routing: Algorithms and application," *Transp. Res. Rec.*, vol. 1771, pp. 172–178, Jan. 2001.

- [32] A. Haghani and H. Qiao, "Snow emergency vehicle routing with route continuity constraints," *Transp. Res. Rec.*, vol. 1783, pp. 119–124, Jan. 2002.
- [33] Y. Yuan and D. Wang, "Path selection model and algorithm for emergency logistics management," *Comput. Ind. Eng.*, vol. 56, no. 3, pp. 1081–1094, 2009.
- [34] X. Zhang, Z. Zhang, Y. Zhang, D. Wei, and Y. Deng, "Route selection for emergency logistics management: A bio-inspired algorithm," *Safety Sci.*, vol. 54, pp. 87–91, Apr. 2013.
- [35] F. Wex, G. Schryen, S. Feuerriegel, and D. Neumann, "Emergency response in natural disaster management: Allocation and scheduling of rescue units," *Eur. J. Oper. Res.*, vol. 235, no. 3, pp. 697–708, Jun. 2014.
- [36] F. Hu, W. Xu, and X. Li, "A modified particle swarm optimization algorithm for optimal allocation of earthquake emergency shelters," *Int. J. Geograph. Inf. Sci.*, vol. 26, no. 9, pp. 1643–1666, 2012.
- [37] S. Yan, C. K. Lin, and S. Y. Chen, "Optimal scheduling of logistical support for an emergency roadway repair work schedule," *Eng. Optim.*, vol. 44, no. 9, pp. 1035–1055, 2012.
- [38] S. Yang, M. Hamed, and A. Haghani, "Online dispatching and routing model for emergency vehicles with area coverage constraints," *Transp. Res. Rec., J. Transp. Res. Board.*, vol. 1923, pp. 1–8, Jan. 2005.
- [39] H. Park, A. Shafahi, and A. Haghani, "A stochastic emergency response location model considering secondary incidents on freeways," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 9, pp. 2528–2540, Sep. 2016.
- [40] P. Wu, A. Che, F. Chu, and M. Zhou, "An improved exact ε -constraint and cut-and-solve combined method for biobjective robust lane reservation," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 3, pp. 1479–1492, Jun. 2015.
- [41] Y. Y. Haimes, L. Ladson, and D. A. Wismer, "On a bicriterion formulation of the problems of integrated system identification and system optimization," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-1, no. 3, pp. 296–297, Jul. 1971.
- [42] C. D. Tarantilis, E. E. Zachariadis, and C. T. Kiranoudis, "A hybrid metaheuristic algorithm for the integrated vehicle routing and three-dimensional container-loading problem," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 2, pp. 255–271, Jun. 2009.
- [43] A. Che, P. Wu, F. Chu, and M. C. Zhou, "Improved quantum-inspired evolutionary algorithm for large-size lane reservation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 12, pp. 1535–1548, Dec. 2015.
- [44] G. Tian, M. Zhou, J. Chu, T. Qiang, and H. Hu, "Stochastic cost-profit tradeoff model for locating an automotive service enterprise," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 2, pp. 580–587, Apr. 2015.
- [45] N. Wu, L. Bai, and M. Zhou, "An efficient scheduling method for crude oil operations in refinery with crude oil type mixing requirements," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 3, pp. 413–426, Mar. 2016.
- [46] P. Wu, A. Che, F. Chu, and Y. Fang, "Exact and heuristic algorithms for rapid and station arrival-time guaranteed bus transportation via lane reservation," *IEEE Trans. Intell. Transp. Syst.*, to be published. doi: 10.1109/TITS.2016.2631893.
- [47] M. Ghahramani, M. Zhou, and C. T. Hon, "Toward cloud computing QoS architecture: Analysis of cloud systems and cloud services," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 1, pp. 6–18, Jan. 2017.
- [48] L. Wang, S. Wang, and X. Zheng, "A hybrid estimation of distribution algorithm for unrelated parallel machine scheduling with sequence-dependent setup times," *IEEE/CAA J. Autom. Sinica*, vol. 3, no. 3, pp. 235–246, Jul. 2016.
- [49] Y. Hou, N. Wu, M. Zhou, and Z. Li, "Pareto-optimization for scheduling of crude oil operations in refinery via genetic algorithm," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 47, no. 3, pp. 517–530, Mar. 2017.
- [50] W. Han *et al.*, "Cuckoo search and particle filter-based inverting approach to estimating defects via magnetic flux leakage signals," *IEEE Trans. Magn.*, vol. 52, no. 4, pp. 1–11, Apr. 2016.
- [51] J. Li, M. Zhou, Q. Sun, X. Dai, and X. Yu, "Colored traveling salesman problem," *IEEE Trans. Cybern.*, vol. 45, no. 11, pp. 2390–2401, Nov. 2015.