

Article

Firefighting Drone Configuration and Scheduling for Wildfire Based on Loss Estimation and Minimization

Rong-Yu Wu, Xi-Cheng Xie and Yu-Jun Zheng * 

School of Information Science and Technology, Hangzhou Normal University, Hangzhou 311121, China; 2021112011047@stu.hznu.edu.cn (R.-Y.W.); 2022112011002@stu.hznu.edu.cn (X.-C.X.)

* Correspondence: yujun.zheng@computer.org

Abstract: Drones have been increasingly used in firefighting to improve the response speed and reduce the dangers to human firefighters. However, few studies simultaneously consider fire spread prediction, drone scheduling, and the configuration of supporting staff and supplies. This paper presents a mathematical model that estimates wildfire spread and economic losses simultaneously. The model can also help us to determine the minimum number of firefighting drones in preparation for wildfire in a given wild area. Next, given a limited number of firefighting drones, we propose a method for scheduling the drones in response to wildfire occurrence to minimize the expected loss using metaheuristic optimization. We demonstrate the performance advantages of water wave optimization over a set of other metaheuristic optimization algorithms on 72 test instances simulated on selected suburb areas of Hangzhou, China. Based on the optimization results, we can pre-define a comprehensive plan of scheduling firefighting drone and configuring support staff in response to a set of scenarios of wildfire occurrences, significantly improving the emergency response efficiency and reducing the potential losses.

Keywords: wildfire; fire spread; firefighting drones; scheduling; metaheuristic optimization; water wave optimization



Citation: Wu, R.-Y.; Xie, X.-C.; Zheng, Y.-J. Firefighting Drone Configuration and Scheduling for Wildfire Based on Loss Estimation and Minimization. *Drones* **2024**, *8*, 17. <https://doi.org/10.3390/drones8010017>

Academic Editor: Diego González-Aguilera

Received: 3 November 2023

Revised: 7 January 2024

Accepted: 7 January 2024

Published: 10 January 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Wildfires, such as forest fires and mountain fires, are one of the most frequent disasters causing great economic losses, environmental damage, and threats to lives worldwide. Efficient firefighting operations are critical to reduce the losses. However, traditional firefighting operations often pose significant dangers to human firefighters. In addition, traditional aerial firefighting using helicopters, air tankers, and other manned aircraft are often limited in low flexibility, scalability, and cost-effectiveness [1,2]. In recent years, unmanned robotics, especially drones (also known as unmanned aerial vehicles, UAVs), have been increasingly used in firefighting operations to improve the response speed and economic effectiveness while reducing the dangers to human firefighters [3–6]. In wild areas, the typical usage of firefighting drones is to deliver water capsules (bags) to fire spots and then release water (or other water-based liquids) in dispersed form to suppress fire in the covered area [7,8]. Depending on the types of drones, the amount of water that a drone carries at a time typically varies from 10 to 1000 kg. Figure 1 shows a TD220 coaxial firefighting drone (made by Zhongzhihang [9]), which has a payload of 120 kg.

Qin et al. [3] designed a drone firefighting system, which consisted of a quadcopter as the platform, a transmission system to collect and release water, a kinematic-based navigation system, and a mission control system to coordinate the drone to find an optimal path (with respect to distance and power consumption) to the fire spot and then suppress the fire. Viegas et al. [10] designed a lightweight tethered UAV forest firefighting drone, which utilized water jet propulsion combined with multi-rotor propulsion to increase the flight endurance. However, receiving a continuous intake of water for firefighting can

be impossible in wild areas, as the operational distance is limited by the length of the aerial hose. Using water capsules delivered by aircraft, to and released on the fire spot, eliminates this limitation [11]. The aircraft releases a water capsule at an appropriate position. When moving in a medium above the critical temperature, the capsule shell accumulates an integral amount of damage; when the damage reaches the value of the thermal stability coefficient of the shell, the shell breaks and releases water in dispersed form [8]. Explosive blasts in water makes shock waves, so shock waves propagate in water first and then beyond water to continue propagation in the air. Explosive water mist lowers the temperature, insulates oxygen, and asphyxiates the absorption of heat radiations. Large momentum also enables the mist to pass through smoke to act on the surface of fuel and (sometimes) soaks into the fuel further [12]. Successive explosions of multiple capsules can distribute water to cover the fire vulnerability zone more fully. How to choose an appropriate water-dropping scheme to achieve the optimal firefighting effect is quite a difficult problem. Śmigielski et al. [13] used a numerical method of distribution propagation and uncertainty propagation to analyze the precision of system controlling delivery of a water capsule by a helicopter. Czerniak et al. [14] employed an artificial bee colony (ABC) to optimize the physical model of flight of a water bag dropped from an aircraft. They also used the ABC method to optimize fuel consumption of the aircraft [15]. Wang et al. [16] proposed a combined genetic algorithm (GA) for optimizing the water-dropping scheme for a fixed-wing firefighting aircraft based on a neural network agent model between the aircraft water-dropping schema and a water distribution polygon.



Figure 1. A firefighting drone with a water capsule [9].

Smart drones are subject to limited endurance and low load capabilities. Effective path planning (if needed, with the consideration of obstacle and collision avoidance) of firefighting drones in a complex, wild environment is important. UAV path planning methods have been extensively studied in the literature, which can be categorized into local and global search methods depending on whether path planning is considered as a global optimization problem [17]. Global search methods can be further divided into problem-specific heuristic methods and metaheuristic methods [18–21]. Luo et al. [22] proposed an improved D*Lite algorithm for planning the paths of UAVs supplemented by unmanned ground vehicles (UGVs), which is effective even when the 3D workspace is only partially known. Harikumar et al. [23] proposed an Oxyrrhis-Marina-inspired search and dynamic formation control method, which first selects between Levy flight, Brownian search, and directionally driven Brownian search for target identification and then controls the UAVs to fly in a dynamic formation to quench the fire using water. Zheng

et al. also studied the heuristic and metaheuristic methods to search and identify targets using UAVs [24] as well as human–UAV cooperation [25,26]. Wang et al. [27] proposed an adaptive vortex search algorithm of planning optimal firefighting UAV paths in terms of the solution quality, length, and energy. Xiang and Wang [28] presented an improved ant colony optimization (ACO) algorithm, which combines pseudo-random rules and roulette, and updates the pheromone concentration based on the distribution rules of the wolf colony algorithm to give strong feedback to the ants. Based on a forest fire risk map, Xu et al. [29] presented a method using ring self-organizing mapping to plan a flight path for forest fire monitoring according to the forest fire risk level. Alsammak et al. [30] used swarm intelligence to model autonomous and decentralized behaviors for a drone swarm, which used an improved random walk algorithm to explore distributed fire spots and a self-coordination mechanism based on the stigmergy to extinguishing fire cooperatively.

Nevertheless, few works on firefighting drone path planning integrate the modeling of fire propagation, which is a decision basis for firefighting plans. Xavier Viegas [31] studied the linkage between convection and radiation in the fire propagation in laboratory experiments and then generalized the results for wind-driven fires to interpret the global movement of the fire front. Li et al. [32] presented a forest fire spreading simulation system, to visualize the impact of multi-factors on the fire spread, which can be used to identify the key sites for the prevention and the control of forest fires. Cellular automata is one of the major methodologies for describing the dynamics of fire propagation. Alexandridis et al. [33] presented the simulation results of a cellular automata model describing the dynamics of a forest fire spread on a mountainous landscape, taking into account factors such as the type and density of vegetation, the wind speed and direction, and the spotting phenomenon. Some model parameters were tuned using a black-box non-linear optimization approach. Rui et al. [34] constructed an improved model that couples cellular automata with an existing forest fire model, considering the impact of time steps on simulation accuracy to provide an optimal time-step value. They tested the model on a case study of a forest fire at Daxing'an Mountain in China, 2006. Mutthulakshmi et al. [35] adopted a physical model to simulate the spread and extinguishing of fire in the context of Dumai, Indonesia. They applied cellular automata to predict the effects of firefighting intervention with the spatial and propagation dynamics of fire. Wu et al. [36] established a fire propagation model containing multidimensional physical and environmental variables, where an artificial neural network (ANN) was used to analyze spatial time series patterns. A restriction of the grid-based models is that they cannot precisely represent complex topographic differences among cells. There were also studies on fire spreading in and/or between buildings [37,38], for which surface area of fuel/floor, ventilation, and thermal conductivity of the boundary material play critical roles. In comparison, wildfire spreading mainly relies on wind force and combustible vegetation density.

To the best of our knowledge, no previous work simultaneously considers fire spread prediction, drone scheduling, and the configuration of support staff and supplies for firefighting. Moreover, most existing work focus on extinguishing a fire as quickly as possible. However, in most real-world cases, different fire spots have different priorities, e.g., a subarea with expensive cash crops should be prior to an uncultivated subarea. Therefore, scheduling firefighting drones to minimize the total fire loss based on the prediction of fire spread and losses is practically significant, but it is a challenging task due to the complexity of fire dynamics and its relation to fire losses. To address this challenge, in this study, first we present a mathematical model that predicts dynamic wildfire spread together with the related economic losses, which is validated on two real-world wildfires. The model output can also help us to determine the minimum number of firefighting drones in preparation for wildfire in a given wild area. Next, given a limited number of firefighting drones, we propose a method for scheduling the drones in response to wildfire occurrence to minimize the expected loss using metaheuristic optimization. We demonstrate the performance advantages of water wave optimization (WWO) [39] over a set of other popular metaheuristic optimization algorithms. Based on the optimization

results, we can pre-define a comprehensive plan of scheduling firefighting drones and configuring support staff in response to a set of scenarios of wildfire occurrences. We conduct experiments on a set of 72 test instances simulated on some forest park areas in Hangzhou, China, and the results demonstrate that the proposed method significantly improved the emergency response efficiency and reduced potential losses. The main contribution of this paper can be summarized as follows:

- We present a mathematical model of wildfire spreading that simulates dynamic fire development and propagation, and, at the same time, estimates the economic losses caused by the fire.
- We propose a heuristic optimization method for configuring and scheduling firefighting drones to minimize the expected total wildfire loss.
- We conduct expensive computational experiments to validate the effectiveness and efficiency of the proposed method on real-world instances.

In the remainder of this paper, we describe the mathematical model for predicting wildfire spread and the corresponding economic losses in Section 2, present the optimization method for drone scheduling in Section 3, and present the experimental results in Section 4. Finally, we conclude with discussions in Section 5.

2. Wildfire Spread Modeling and Loss Estimation

Given a wild area A , we divided it into a set of m subareas $\{A_1, A_2, \dots, A_m\}$ according to two basic criteria: (1) two adjacent subareas have different topographic and/or environmental features, for example, one subarea is full of tall trees and the other is mainly with shrubs; (2) the area and vegetation biomass of a subarea are not very large, such that the water volume for extinguishing the fire in the subarea is reasonable (normally 1000~10,000 kg, which can be carried by 5 to 20 drones at a time). Typically, we first divide the area according to the first criterion; if some resulting subareas are too large to satisfy the second criterion, we then divide them into smaller subareas according to the second criterion. After division, we estimate the fire dynamics and losses in each subarea and the propagation between different subareas.

2.1. Fire Dynamics and Heat Release

The typical process of a wildfire consists of three stages: preheat (growth), full combustion, and decay [31,40], which are illustrated in Figure 2 [41]. Given the time t_i^{ig} of fire ignition in subarea A_i , temperature $T(t)$, (relative) humidity $M(t)$, wind force $wf(t)$, and wind direction $wd(t)$ in each time slice t , the present model estimates the process of fire growth in A_i and the process of fire propagation to areas adjacent to A_i , as well as the corresponding economic losses. Table 1 presents the model parameters used in this paper.

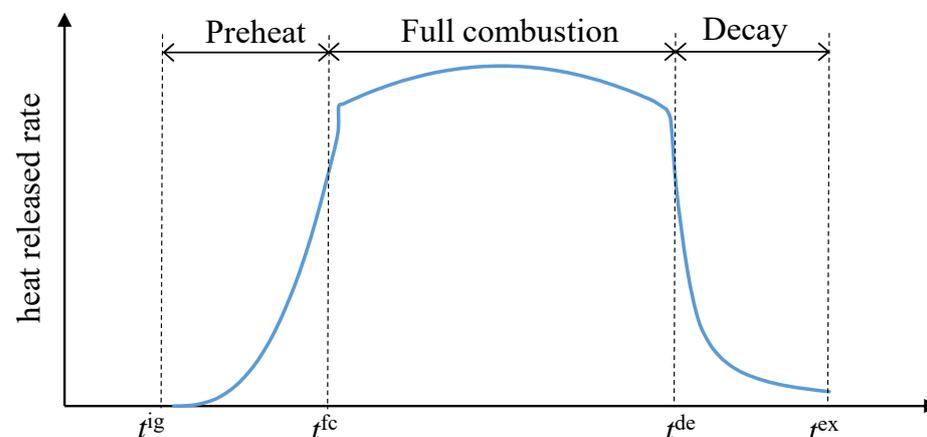


Figure 2. Illustration of the typical stages of a wildfire.

Table 1. Model parameters used in this paper.

Parameter	Description
m	Number of subareas
a_i	Area (m^2) of subarea A_i
t_i^{ig}	Time of fire ignition in subarea A_i
ρ_i	Combustible vegetation density ρ_i of subarea A_i
Q_i	Total combustible heat of the vegetation in subarea A_i
$T(t)$	Temperature at time t
$M(t)$	Humidity at time t
$wf(t)$	Wind force at time t
$wd(t)$	Wind direction at time t
$h(T, M)$	Coefficient used in Equation (1) for calculating the heat release rate in the preheat stage
$g(wf(t))$	Exponent used in Equation (1) for calculating the heat release rate in the preheat stage
$h'(wf(t))$	Coefficient used in Equation (3) for calculating the heat release rate in the full combustion stage
$g'(wf(t))$	Exponent used in Equation (5) for calculating the heat release rate in the decay stage
$\omega(wf(t))$	Coefficient used in Equation (12) for calculating the probability of fire propagation
t_i^{fc}	Time at which the fire in subarea A_i enters into the full combustion stage
t_i^{de}	Time at which the fire in subarea A_i enters into the decay stage
t_i^{ex}	Time at which the fire in subarea A_i is naturally extinguished
$\hat{\theta}$	Threshold of heat release rate for the fire enters into the full combustion stage
$\hat{p}Q$	Threshold of the ratio of the total released heat to Q_i for the fire enters into the decay stage
v_i	Valuation of vegetation in subarea A_i
v'_i	Valuation of vulnerable assets in subarea A_i
$lb_{i,i'}$	Length of the boundary between two adjacent subareas A_i and $A_{i'}$
$\hat{l}b$	Threshold of boundary length for fire propagation
$\angle(l_a, l_b)$	Angle between two lines l_a and l_b
W	Amount of water that can be carried by a drone at a time
D	Maximum distance of the drone
W	Maximum load of the drone
v_u^{\max}	Maximum velocity of the drone
v_u^{\min}	Minimum velocity of the drone

1. Preheat stage. After the ignition time t_i^{ig} , the heat release rate $\theta_i^{ph}(t)$ continuously grows with time t , and its growth rate depends on the combustible vegetation density ρ_i of the subarea, temperature $T(t)$, humidity $M(t)$, and the wind force $wf(t)$:

$$\theta_i^{ph}(t) = \rho_i h(T(t), M(t)) (t - t_i^{ig})^{g(wf(t))} \quad (1)$$

where $h(\cdot)$ is a function of temperature and humidity, and $g(\cdot)$ is a function of wind force. Currently, we define $h(\cdot)$ only on the temperature range from -10° to 50° , which covers the temperature ranges in most of East China and South China. For convenience, we simply define $g(\cdot)$ on ten levels of wind force (wind force above level 10 rarely occurs on the mainland), as shown in the function in Table 2.

2. Full combustion stage, in which gas combustion is dominant. Whenever the heat release rate reaches a threshold $\hat{\theta}$, i.e., $\theta_i^{ph}(t) \geq \hat{\theta}$, the fire in the subarea enters into the full combustion stage, the time at which is denoted as t_i^{fc} :

$$t_i^{fc} = \min_{t' \geq t_i^{ig}} \{ \theta_i^{ph}(t) \geq \hat{\theta} \} \quad (2)$$

The heat release rate during this stage is relatively stable:

$$\theta_i^{fc}(t) = c_1 \rho_i - h'(wf(t)) (t - t_i^{fc})^2 \quad (3)$$

where c_1 is a constant, and $h'(\cdot)$ is a function of the wind force, the values of which is shown in the third row of the function in Table 2.

3. The decay stage, in which charcoal combustion is dominant. Whenever the ratio of the total released heat to the total combustible heat of the vegetation Q_i in the subarea reaches a threshold \hat{p}_Q , the fire in the subarea enters into the decay stage, the time at which is denoted as t_i^{de} :

$$t_i^{de} = \min_{t' \geq t_i^{fc}} \left\{ \left(\int_{t_i^{ig}}^{t_i^{fc}} \theta_i^{ph}(t) dt + \int_{t_i^{fc}}^{t'} \theta_i^{fc}(t) dt \right) \geq \hat{p}_Q Q_i \right\} \tag{4}$$

The heat release rate during this stage decreases with time:

$$\theta_i^{de}(t) = c_2 \rho_i (t - t_i^{de})^{-g'(wf(t))} \tag{5}$$

where c_2 is a constant and $g'(\cdot)$ is a function of the wind force, the values of which is shown in the fourth row of the function in Table 2. Note that $t = t_i^{de}$ will cause a division-by-zero in Equation (5); at this time, the heat release rate should be calculated according to $\theta_i^{fc}(t)$ in Equation (3).

Whenever the heat release rate decreases to a lower limit $\underline{\theta}$ (or the total released heat reaches the total combustible heat), the fire is extinguished, the time at which is denoted as t_i^{ex} :

$$t_i^{ex} = \min_{t' \geq t_i^{de}} \{ \theta_i^{de}(t) \leq \underline{\theta} \} \tag{6}$$

As a result, the heat release rate of the fire in subarea A_i at each time t is:

$$\theta_i(t) = \begin{cases} \theta_i^{ph}(t) & t_i^{ig} < t \leq t_i^{fc} \\ \theta_i^{fc}(t) & t_i^{fc} < t \leq t_i^{de} \\ \theta_i^{de}(t) & t_i^{de} < t \leq t_i^{ex} \\ 0 & \text{else} \end{cases} \tag{7}$$

Table 2. Function values of $g(\cdot)$ used in Equation (1), $h'(\cdot)$ used in Equation (3), $g'(\cdot)$ used in Equation (5), and $\omega(\cdot)$ used in Equation (12).

Wind Force Level	0	1	2	3	4	5	6	7	8	9	10
$g(\cdot)$ values	1.414	1.503	1.691	2.000	2.265	2.673	2.967	3.341	3.568	4.102	4.609
$h'(\cdot)$ values	0.55	0.63	0.77	0.92	1.01	1.18	1.29	1.40	1.53	1.68	1.85
$g'(\cdot)$ values	1.64	1.32	1.21	1.12	0.97	0.90	0.83	0.77	0.74	0.65	0.61
$\omega(\cdot)$ values	0.16	0.25	0.33	0.46	0.6	0.89	1	1	1	1	1

2.2. Loss Estimation

We consider two types of losses: losses of vegetation and losses of other vulnerable assets. Let v_i be the valuation of vegetation and v'_i be the valuation of other vulnerable assets in subarea A_i . At each time τ , we calculate the first type of losses according to the ratio of the total released heat to the total combustible heat of the vegetation:

$$LV_i(\tau) = \begin{cases} \frac{\int_{t_i^{ig}}^{\tau} \theta_i^{ph}(t) dt}{Q_i} v_i & t_i^{ig} < \tau \leq t_i^{fc} \\ \frac{\int_{t_i^{ig}}^{t_i^{fc}} \theta_i^{ph}(t) dt + \int_{t_i^{fc}}^{\tau} \theta_i^{fc}(t) dt}{Q_i} v_i & t_i^{fc} < \tau \leq t_i^{de} \\ \frac{\int_{t_i^{ig}}^{t_i^{fc}} \theta_i^{ph}(t) dt + \int_{t_i^{fc}}^{t_i^{ex}} \theta_i^{fc}(t) dt + \int_{t_i^{de}}^{\tau} \theta_i^{de}(t) dt}{Q_i} v_i & t_i^{de} < \tau \leq t_i^{ex} \\ v_i & \tau > t_i^{ex} \end{cases} \tag{8}$$

For other vulnerable assets, we assume that they are partially lost during the preheat stage and completely lost if the fire enters the full combustion stage:

$$LV'_i(\tau) = \begin{cases} \frac{\int_{t_i^{ig}}^{\tau} \theta_i^{ph}(t) dt}{Q_i} v'_i & t_i^{ig} < \tau \leq t_i^{fc} \\ v'_i & \tau > t_i^{fc} \end{cases} \quad (9)$$

The total loss in A_i at time τ is

$$L_i(\tau) = LV_i(\tau) + LV'_i(\tau) \quad (10)$$

2.3. Fire Propagation

Fire spread in a subarea is unimpeded. On the contrary, the propagation of a fire in a subarea A_i to an adjacent area $A_{i'}$ is considered as probabilistic, and the probability $p_{i,i'}(t)$ depends on the heat release rate, wind power and direction, and open boundary between the subareas (e.g., for two subareas mainly separated by a river but linked by a narrow shrub zone, of which the narrow shrub zone is considered as the open boundary between the subareas). Let $lb_{i,i'}$ be the length of the open boundary between A_i and $A_{i'}$ and $\alpha = \angle(wd, ol_{i,i'})$ be the angle between the wind direction wd and the line $ol_{i,i'}$ orthogonal to the open boundary (as shown in Figure 3). We define

$$\omega(\alpha) = \begin{cases} \left[\frac{\cos(\alpha) + \delta_c}{1 + \delta_c} \right]^{e_c} & 0 \leq \alpha \leq 90^\circ \\ \left[\frac{(1 + 0.75 \cos(\alpha)) \delta_c}{1 + \delta_c} \right]^{e_c} & \text{else} \end{cases} \quad (11)$$

where δ_c and e_c are two constants between 0 and 1 (taking values of 0.2 and 0.5 in our study, respectively, which results in that $\omega(0^\circ) = 1$, $\omega(90^\circ) = 1/6$, and $\omega(180^\circ) \approx 1/25$).

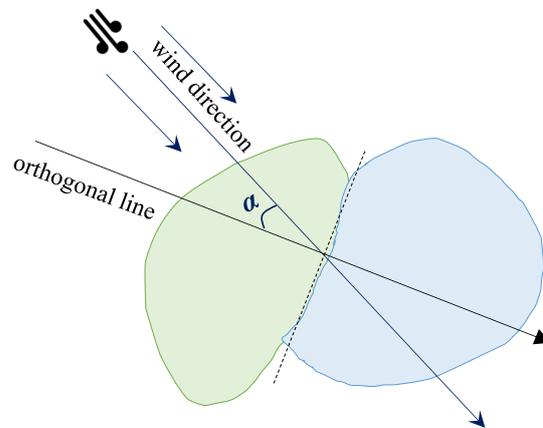


Figure 3. Angle between the wind direction and the line orthogonal to the open boundary between two subareas.

The probability that the fire propagates from A_i to $A_{i'}$ at the next time $t + 1$ is calculated as

$$p_{i,i'}(t + 1) = \frac{\theta_i(t)}{\hat{\theta}} \cdot \frac{lb_{i,i'}}{\hat{lb}} \cdot \omega(\angle(wd(t), ol_{i,i'})) \cdot \omega(wf(t)) \quad (12)$$

where \hat{lb} is a threshold of the boundary length, and $\omega(\cdot)$ is a function of the wind force, the values of which is shown in the fifth row of the function in Table 2. As we can see, the value is always one when the wind force level is above six. That is, if the fire is in full combustion (as the heat release rate reaches the threshold $\hat{\theta}$), the open boundary length is not shorter than the threshold \hat{lb} , the wind direction is in line with the line orthogonal to the open boundary, and the wind force level reaches or exceeds six, of which the probability is then 1, i.e., the fire propagates from A_i to $A_{i'}$ deterministically.

In the case that there are multiple burning subareas $\{A_{i_1}, A_{i_2}, \dots, A_{i_K}\}$ adjacent to a subarea A_i , the joint probability of the ignition in subarea A_i is calculated as

$$p_i^{\text{ig}}(t) = 1 - \prod_{k=1}^K (1 - p_{i_k, i}(t)) \quad (13)$$

2.4. Simulation Process

Suppose that a set \mathbf{A}^* of one or more subareas are burning at the beginning time. From $t = 0$ to a given end time t_{end} , we can simulate the process of fire propagation by iteratively calculating the ignition probabilities and estimating the fire development in related subareas at each time slice t using the following steps:

1. Let $t = 0$; for each subarea $A_i \notin \mathbf{A}^*$, initialize its accumulated ignition probability $pc_i^{\text{ig}}(0) = 0$.
2. Set $t = t + 1$; if $t = t_{\text{end}}$, then exit.
3. For each subarea $A_i \in \mathbf{A}^*$, calculate its heat release rate according to Equations (1)–(7) (if needed, update its fire stage), and then calculate its loss according to Equations (8)–(10).
4. For each subarea $A_i \notin \mathbf{A}^*$ and for each $\tau \in [1, t]$ satisfying $p_i^{\text{ig}}(\tau) > 0$ (i.e., the subarea has a probability of being ignited at time τ), calculate its heat release rate according to Equations (1)–(7) (if needed, update its fire stage), and then calculate its loss according to Equations (8)–(10) under the condition of $t_i^{\text{ig}} = \tau$.
5. For each subarea $A_i \notin \mathbf{A}^*$:
 - (a) Initialize the non-ignition probability $p_i^{-\text{ig}}(t) = 1$.
 - (b) For each subarea $A_{i'} \in \mathbf{A}^*$ that is adjacent to A_i :
 - i. Calculate $p_{i', i}(t)$ according to Equation (12).
 - ii. Update the non-ignition probability as

$$p_i^{-\text{ig}}(t) = p_i^{-\text{ig}}(t)(1 - p_{i', i}(t)) \quad (14)$$

- (c) For each subarea $A_{i'} \notin \mathbf{A}^*$ that is adjacent to A_i while having $pc_{i'}^{\text{ig}}(t-1) > 0$:
 - i. Calculate the probability of the propagation from $A_{i'}$ to A_i as

$$p_{i', i}(t) = \frac{E(\theta_{i'}(t))}{\hat{\theta}} \cdot \frac{lb_{i', i}}{\hat{lb}} \cdot \omega(\angle(wd(t), ol_{i', i})) \cdot \omega(wf(t)) \quad (15)$$

where $E(\theta_{i'}(t))$ is the expected heat release rate of $A_{i'}$ at time t , which is calculated as

$$E(\theta_{i'}(t)) = \sum_{\tau=1}^{t-1} \theta(\tau | t_{i'}^{\text{ig}} = \tau) p_{i'}^{\text{ig}}(\tau) \quad (16)$$

- ii. Update the non-ignition probability according to Equation (14).
- (d) Set $p_i^{\text{ig}}(t) = 1 - p_i^{-\text{ig}}(t)$.
- (e) If $p_i^{\text{ig}}(t) \geq 1 - \epsilon$ (where ϵ is a small value, which is set to 0.001 in this study), then add A_i to \mathbf{A}^* ;
- (f) Otherwise, update the accumulated probability $pc_i^{\text{ig}}(t)$ of ignition in A_i as

$$pc_i^{\text{ig}}(t) = pc_i^{\text{ig}}(t-1) + (1 - pc_i^{\text{ig}}(t-1)) p_i^{\text{ig}}(t) \quad (17)$$

If $pc_i^{\text{ig}}(t) \geq 1 - \epsilon$, then add A_i to \mathbf{A}^* .

6. Go to step 2.

For model verification, we used the above process to simulate two wildfires that occurred in Zhejiang Province, China: one in 2020 and the other in 2022. The data came from the Emergency Management Department of Zhejiang Province, recording the time at which each subarea entered into the full combustion stage and the decay stage (as the preheat stages of wildfires were difficult to monitor, the corresponding times were missing). We conducted Monte Carlo simulations, with 50 trials on each wildfire, and compared the simulated full combustion time and decay time with the actual time, then presented the results of the two fires in Figures 4 and 5, respectively. The results showed that the deviations of the simulated time curves from the corresponding actual time curves were generally small. In particular, the full combustion time curves fitted with the actual time curves well. On the two fires, the mean absolute percentage errors (MPAE) of the full combustion time were 7.77% and 6.98%, and the MPAE of the decay time were 8.56% and 7.81%, respectively. As the full combustion stage plays the most important role in fire spread and causes the most losses, the proposed model for wildfire spread and loss estimation is practically useful.

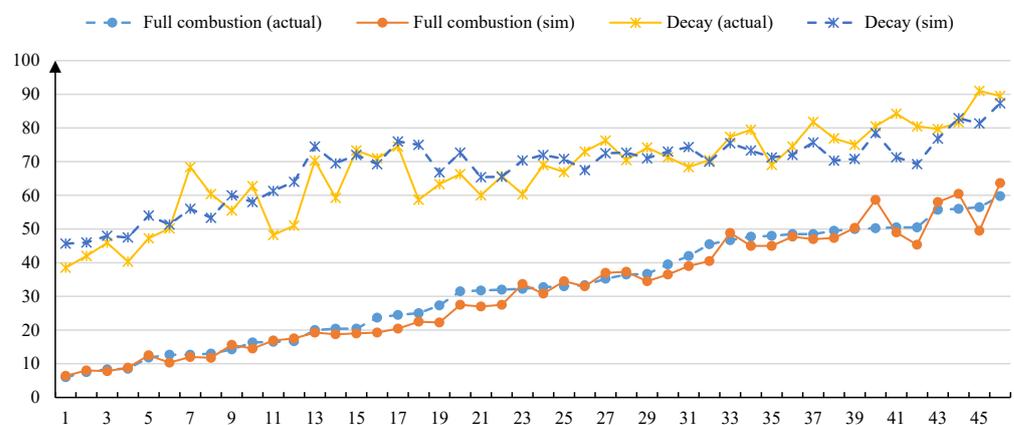


Figure 4. Comparison of the simulated and actual fire spread in the first wildfire (fire area consisting of 46 subareas).

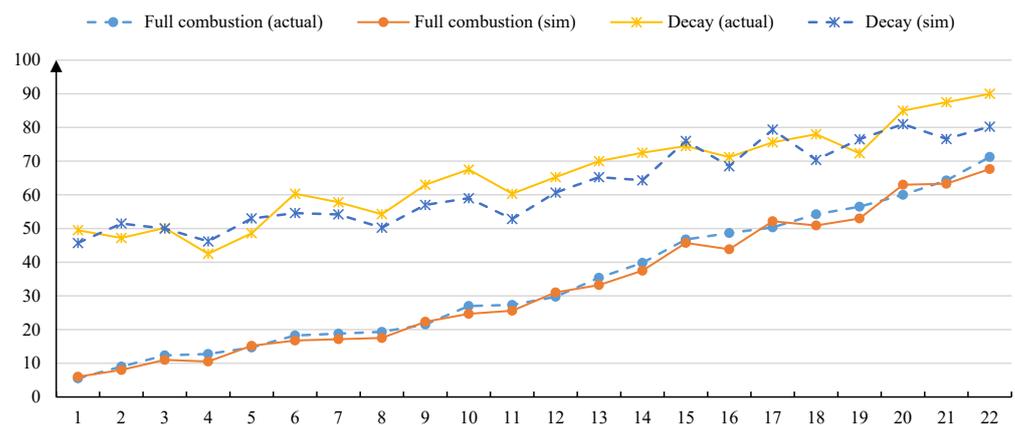


Figure 5. Comparison of the simulated and actual fire spread in the second wildfire (fire area consisting of 22 subareas).

3. Drone Configuration and Scheduling

3.1. Minimum Number of Firefighting Drones in Preparation for Wildfire

Let a_i be the area (in m^2) of subarea A_i . The amount of water for extinguishing the fire in the area is proportional to a_i and $\theta_i(t)$ at time t . Let W be the amount of water that can be carried by a drone at a time (under the assumption of homogeneous drones, which is

not difficult to extend to heterogeneous drones). The least number of firefighting drones required for extinguishing the fire in subarea A_i at time t is

$$N_i(t) = \left\lceil c_3 \frac{a_i \theta_i(t)}{W} \right\rceil \quad (18)$$

where $\lceil \cdot \rceil$ denotes rounding up to the closest integer, and c_3 is a constant. The value of $\theta_i(t)$ is at most $\hat{\theta}$, and the number is at most

$$\hat{N}_i = \left\lceil c_3 \frac{a_i \hat{\theta}}{W} \right\rceil \quad (19)$$

A basic principle of firefighting drone preparation for wildfire is that the usage of all drones is sufficient to extinguish the fire in any subarea at any time. Therefore, the minimum number of firefighting drones in preparation for wildfire in the while wild area can be determined as

$$N_{drones}^{\min} = \min_{1 \leq i \leq m} \{ \hat{N}_i \} \quad (20)$$

3.2. Optimization Problem of Firefighting Drone Scheduling

For fire prevention for a wild area \mathbf{A} , we normally establish a lightweight fire station near a water resource and equip the station with N_{drones} firefighting drones ($N_{drones} \geq N_{drones}^{\min}$). Let d_i be the distance from the fire station to subarea A_i , v_i be the speed of a fully-loaded drone from the fire station to subarea A_i , and v'_i be the speed of an empty drone (after releasing water) from A_i to the fire station. We also set up at least one fire sensor in each subarea of \mathbf{A} : the probability of the sensor perceiving a fire in the subarea is 1 in the full combustion stage and smaller than 1 in the preheat stage. Anyway, whenever we receive the warning from a sensor in subarea A_i at time 0, we pessimistically assume that $t_i^{fc} = 0$, i.e., the fire is in the full combustion stage.

Suppose that we receive the warning from a set $\mathbf{A}^*(0)$ of one or more subareas at time 0. A firefighting decision \mathbf{x} is defined as an order (sequence) of subareas in \mathbf{A} . Let $\Delta t(\mathbf{A}^*)$ be the time for a drone flying to the shortest subarea in \mathbf{A}^* , i.e.,

$$\Delta t(\mathbf{A}^*) = \min_{A_i \in \mathbf{A}^*} d_i / v_i \quad (21)$$

Whenever there are available drones at time t , we select a candidate set $\mathbf{A}^C(t)$ of subareas, each of which has already been ignited (but not yet extinguished) or has an accumulated ignition probability $pc_i^{ig}(t + \Delta t(\mathbf{A}^*(t)))$ larger than a threshold e_{pc} . We arrange the drones to extinguish the fires in the subareas in $\mathbf{A}^C(t)$ in the same order as in \mathbf{x} .

Based on the above principle, the fitness of a firefighting solution \mathbf{x} is evaluated using the following steps:

1. Let $t = 0$, $N(t) = N_{drones}$ be the initial number of available drones.
2. Calculate $\Delta t(\mathbf{A}^*)(t)$ according to Equation (21) and select the candidate set $\mathbf{A}^C(t)$ of subareas satisfying $pc_i^{ig}(t + \Delta t(\mathbf{A}^*(t))) > e_{pc}$, which are sorted in the same order as in \mathbf{x} .
3. For each subarea $A_i \in \mathbf{A}^C(t)$:
 - (a) Calculate $N_i(t)$ according to Equation (18).
 - (b) If $N_i(t) \leq N(t)$, then assign $N_i(t)$ drones to subarea x_i , whose fire will be extinguished at time $t + d_{x_i} / v_{x_i}$, and these drones will be available at the station at time $t + d_{x_i} / v_{x_i} + d_{x_i} / v'_{x_i}$, and then set $N(t) = N(t) - N_i(t)$.
 - (c) If $N(t) < (\min_{i < i' \leq n_x} \{ N_{i'}(t) \})$, then go to step 4.
4. Set $t = t + 1$.
5. Check whether there is any burning subarea whose fire will be extinguished at time t ; if so, set the extinguish time to t and heat release rate to zero and remove it from $\mathbf{A}^*(t)$.
6. If there is no burning subareas, calculate the total loss and exit.

7. Use steps 3 to 5 described in Section 2.4 to update the states of the other subareas at time t ; if there is any subarea entering into the full combustion stage, add it to $\mathbf{A}^*(t)$; if there is any burning subarea whose fire is naturally extinguished at time t , remove it from $\mathbf{A}^*(t)$.
8. Check whether there are some drones returning to the station at time t ; if so, update the value of $N(t)$.
9. Go to step 2.

For each subarea $A_i \in \mathbf{A}$, if it is ignited as a deterministic time t_i^{ig} and extinguished at time $t_i^{\text{ex}}(\mathbf{x})$ based on the firefighting solution \mathbf{x} , then its expected loss $E(L_i(\mathbf{x}))$ is $L(t_i^{\text{ex}}(\mathbf{x}))$ calculated according to Equations (8)–(10); otherwise, its expected loss is calculated as

$$E(L_i(\mathbf{x})) = \sum_{\tau} L(t_i^{\text{ex}}(\mathbf{x}) | t_i^{\text{ig}} = \tau) p_i^{\text{ig}}(\tau) \quad (22)$$

This problem aims to minimize the total loss:

$$\min f(\mathbf{x}) = \sum_{i=1}^m E(L_i(\mathbf{x})) \quad (23)$$

3.3. Optimization Algorithms

The above firefighting drone scheduling problem is to find an optimal sequence of subareas, which is essentially a permutation optimization problem. The solution space has the same structure as the traveling salesman problem (TSP) and permutation flowshop scheduling problem (PFSP), while the fitness evaluation function is significantly more complex and computationally expensive than TSP and PFSP. For a wild area with dozens to hundreds of subareas, the computational cost of exact optimization methods can be unaffordable. Here, we adapt several popular metaheuristic optimization algorithms for TSP/PFSP, which are briefly described as follows:

- GA using order-based solution representation, partial mapping crossover, and swap mutation [42].
- Particle swarm optimization (PSO) using discrete sequence-based particle representation [43], where velocity trail values are used as the probabilities of the components being placed in certain positions of the sequence. We also incorporate a comprehensive learning strategy [44,45] and an adaptive parameter control mechanism [46].
- Differential evolution (DE) adapted for permutation optimization based on floating-to-integer mapping [47], where solutions are encoded as floating vectors and evolved via standard DE mutation and crossover, then decoded to integer sequences based on the order of floating values.
- Biogeography-based optimization (BBO) for permutation optimization based on subsequence migration [48,49]. The migration operator selects a subsequence from the emigrating solution and uses it to replace the corresponding part in the immigrating solution while using the original components in the part to substitute the corresponding components in the other part to avoid duplication.
- Ecogeography-based optimization (EBO) that extends BBO by integrating local and global migration [50,51].

As the fitness evaluation of a solution to the problem involves the computationally expensive simulation process, we mainly focus on WWO [39], which uses a small population to avoid too many fitness evaluations. WWO is a metaheuristic mimicking water wave motions to search the solution space. In WWO, each solution \mathbf{x} is analogous to a wave with a wavelength $\lambda(\mathbf{x})$, which is inversely proportional to the solution fitness; at each iteration, each solution searches in a range proportional to its wavelength such that high-fitness solutions search in small ranges, while low-fitness solutions search in large ranges (as illustrated in Figure 6), which leads to a good balance between diversity and convergence.

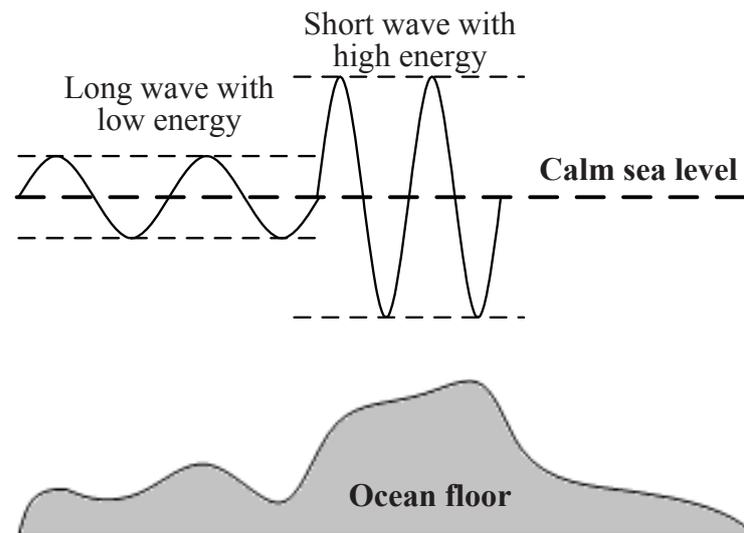


Figure 6. Illustration of wavelength-based search in WWO.

To adapt the original WWO for continuous optimization to this firefighting drone scheduling problem, we redefine its propagation and breaking operators based on the principles from [52]. Propagation of a solution \mathbf{x} is performed, for m times, by each, with a probability of $\lambda(\mathbf{x})$, randomly choosing and reversing a subsequence of \mathbf{x} . In this way, the expected number of subsequence reversals on \mathbf{x} is $m\lambda(\mathbf{x})$, which is inversely proportional to the solution fitness. If the propagated solution is better than the original one, it replaces the original one in the population. After each iteration, the wavelength of each solution is updated as follows (note that the objective function $f(\mathbf{x})$ defined in Equation (23) is inversely proportional to its fitness):

$$\lambda(\mathbf{x}') = \lambda(\mathbf{x})\alpha^{f_{\max} - (f(\mathbf{x} + \epsilon) / (f_{\max} - f_{\min} + \epsilon))} \quad (24)$$

where f_{\max} and f_{\min} are the maximum and minimum objective function values in the population, respectively; α is a constant taking a value of 1.0026; and ϵ is a very small positive value to avoid division by zero.

Whenever the algorithms find a new best-known solution \mathbf{x}^* , a breaking operation is performed by generating K_N (a random number in $[1, m]$) neighboring solutions, each being obtained by randomly swapping two subareas x_i and x_j in the sequence. The best neighbor, if better than \mathbf{x}^* , will replace \mathbf{x}^* in the population. In this way, the algorithm uses diverse solutions to facilitate global search in the early stages while focusing on a small number of solutions to enhance local search in later stages.

We also employ a population reduction policy [53], which reduces the population size NP from an upper limit NP^{\max} to a lower limit NP^{\min} :

$$NP = NP^{\max} - \left(\frac{n_g}{n_g^{\max}} \right)^2 (NP^{\max} - NP^{\min}) \quad (25)$$

where NP^{\max} and NP^{\min} are the upper and lower limit of the population sizes, respectively; n_g is the current generation number; and n_g^{\max} is the maximum generation number. Whenever the population size should be decreased by 1, the current worst solution is removed.

Algorithm 1 presents the pseudo-code of the WWO algorithm for firefighting drone scheduling, where each solution \mathbf{x} is evaluated using the procedure described in Section 3.2.

Algorithm 1: WWO algorithm for firefighting drone scheduling.

```

1 Randomly initialize a population of solutions with wavelengths of 0.5;
2 Let  $n_g = 0$  and  $\mathbf{x}^*$  be the best solution in the population;
3 while  $n_g < n_g^{\max}$  do
4   Calculate the wavelengths of the solutions according to Equation (24);
5   for each solution  $\mathbf{x}$  in the population do
6     Let  $\mathbf{x}' = \mathbf{x}$ ;
7     for  $i = 1$  to  $m$  do
8       if  $\text{rand}(0,1) < \lambda(\mathbf{x})$  then
9         Randomly select and reverse a subsequence of  $\mathbf{x}'$ ;
10      if  $f(\mathbf{x}') < f(\mathbf{x})$  then
11        Set  $\mathbf{x} \leftarrow \mathbf{x}'$ ;
12        if  $f(\mathbf{x}) < f(\mathbf{x}^*)$  then
13          Set  $\mathbf{x}^* \leftarrow \mathbf{x}$ ;
14          Let  $K_N = \text{rand}(1, m)$ ;
15          for  $k = 1$  to  $K_N$  do
16            Generate a neighbor  $\mathbf{x}'$  by randomly selecting and swapping
17              two components of  $\mathbf{x}^*$ ;
18            if  $f(\mathbf{x}) < f(\mathbf{x}^*)$  then
19              Set  $\mathbf{x}^* \leftarrow \mathbf{x}'$ ;
19   Update the population size  $NP$  according to Equation (25) by removing the
20     worst solution;
20  $n_g \leftarrow n_g + 1$ ;
21 return the best-known solution  $\mathbf{x}^*$ .

```

The above WWO algorithm uses component swap for local search. To further improve the performance, we propose an enhanced WWO (EWWO) algorithm that includes two additional local search: one using NEH reconstruction [54] and the other using reinsertion, i.e., randomly selecting a component and inserting it into another position. For each breaking operation, EWWO adaptively selects one from the three local search operators based on their past performance. Initially, the three operators have the same selection probability of 1/3. At each generation after the first LP generations (where LP is the learning period), the probability of each l -th operator ($1 \leq l \leq 3$) is updated based on its performance during the previous LP generations:

$$\rho_l = \frac{c_l n_l^I / n_l}{\sum_{i=1}^3 c_i n_i^I / n_i} \quad (26)$$

where n_l is the number of invocations of the l -th operator, n_l^I is the number of invocations of the operator that produce better solutions in the recent learning period, and c_l is the computational complexity of the l -th operator ($c_1 = 1$, $c_2 = n/2$, and $c_3 = 1$). Using the adaptive local search, Lines 15–18 of Algorithm 1 are replaced by the procedure shown in Algorithm 2.

Algorithm 2: WWO breaking using self-adaptive local search.

```

1 for  $k = 1$  to  $K_N$  do
2   Randomly select the  $l$ -th local search operator according to the probability  $\rho_l$ ;
3   Generate a neighbor  $\mathbf{x}'$  by performing the operator on  $\mathbf{x}^*$ ;
4   if  $f(\mathbf{x}) < f(\mathbf{x}^*)$  then
5     Set  $\mathbf{x}^* \leftarrow \mathbf{x}'$ ;
6     Set  $n_l^I \leftarrow n_l^I + 1$ ;
7 if  $n_g \geq LP$  then
8   Update the selection probabilities according to Equation (26);

```

3.4. Drone/Staff Configuration and Preplanning

Normally, when configuring N_{drones} drones for a wild area, we fit out $2N_{drones}$ water capsules at the fire station such that each drone returning from the fire spot can directly replace the empty capsule with a capsule full of water. Suppose that the (average) time duration for a staff member to fill up a water capsule is \tilde{t} ; if N empty water capsules are left by drones at the station at time t and are required to be filled up at time $t + \Delta t$, the least number of staff is

$$n_s^{\min} = \lceil N \frac{\tilde{t}}{\Delta t} \rceil \quad (27)$$

Given a given fire instance, according to the best-known solution obtained by the optimization algorithms, we select a round of drone dispatch that has the maximum $N/\Delta t$ and then obtain the required number of staff according to Equation (27).

Moreover, let \hat{T} be the maximum flight time duration of a drone with a fully charged battery and T_j be the total flight time duration of each j -th drone in the solution, where the total number of fully charged batteries required for the operation can be calculated as

$$n_{bat} = \sum_{j=1}^{N_{drones}} \left\lceil \frac{T_j}{\hat{T}} \right\rceil \quad (28)$$

Consequently, for a given wild area, we can identify a subset of risky subareas that easily catch fire, as well as record the most common wind force levels and temperature grades. For each risky subarea, we simulate the ignition under each wind force level and temperature grade and use the optimization algorithms to solve the instance. By saving these instances and the corresponding best-known solutions, when encountering a real fire warning from a subarea, we can directly select an existing solution to schedule the drones as well as the required number of staff for firefighting.

Furthermore, we can identify those subareas, where the ignition will cause the most significant losses and/or require the largest number of staff, as the most important subareas and, therefore, to strengthen the management and surveillance of those subareas, we must reduce the risks and consequences as much as possible.

4. Computational Experiments

We test the proposed method in a wild area belonging to the West Mountain Forest Park, Hangzhou, Zhejiang Province, China. The area is divided into 127 subareas, among which eight subareas are identified as high-risk subareas. According to our simulation, when the wind force level is zero (no wind) or one (light air) and the temperature is below 5°C , most fires ignited from the risky areas can be extinguished by one or two rounds of drone dispatch, i.e., the instances of the firefighting drone scheduling problem are quite easy to solve. Therefore, we select a set of three wind levels of {2, 4, 6} (as the differences between two adjacent wind force levels are relatively small, and wind force levels above eight rarely appear in the area) and a set of three temperature grades of {15–20, 25–30, 35–40} ($^\circ\text{C}$). We also set the number of drones to a random integer in $[N_{drones}^{\min}, 1.2N_{drones}^{\min}]$. By simulating the

ignition in each high-risk subarea under each wind level and each temperature grade (the wind direction is always assumed to be toward the center of the area), we construct a suite of 72 test instances.

On the test suite, we compare the seven metaheuristic optimization algorithms (GA, PSO, DE, BBO, EBO, WWO, and EWWO). Each algorithm is run for 30 times on each instance, for which we record the minimum and median objective function values (in thousand RMB Yuan or CNY) among the 30 runs and the standard deviation (std). To ensure a fair comparison, for all algorithms, the termination condition is set so that the number of objective function evaluations reaches 50,000. Table 3 presents the results of the seven algorithms on the 72 test instances. We also conduct a non-parametric Wilcoxon rank-sum test to compare the result of EWWO and the result of other algorithms on each instance. Table 4 summarizes the averaged values of the results obtained by each algorithm on the test instances as well as the corresponding ranks among the seven algorithms.

From the results, we can observe that EWWO always obtains the best result on each test instance. Among the other six popular metaheuristic optimization algorithms, GA performs the worst mainly because the genetic selection and crossover operations are elitism-based, making GA to be easily trapped by the local optima. The migration operator of BBO is similar to genetic crossover, and, therefore, the performance of BBO is close to that of GA, although the BBO migration model provide better diversity than genetic selection. Using velocity-based solution movement, the PSO algorithm converges fast, but it also easily falls into premature convergence. Both the DE mutation schema and the EBO global migration model have powerful global exploration abilities and can result in good diversities, and their performances are better than GA, PSO, and BBO. WWO performs the best among the first six algorithms because its wavelength-based control model brings a quite good balance between global exploration and local exploitation. The performance advantage of EWWO over WWO, obviously, is due to the integration and adaptive control of the three local search operators. For the instances with significant losses, using the solutions of EWWO, we can reduce approximately 300,000~600,000 CNY of losses compared to the solutions of GA and approximately 100,000~400,000 CNY compared to the solutions of other algorithms. In summary, the experimental results demonstrate that EWWO exhibits a significantly better performance than the other popular algorithms on the test suite. Therefore, we recommend EWWO as the most suitable algorithm to solve the firefighting drone scheduling problem to the fire department.

According to the best-known results obtained by EWWO, among the eight subareas, under the most serious conditions (highest temperature and wind force), the losses in six subareas are approximately 1.1~1.6 billion CNY, while the losses in the other two subareas are 2.2 and 1.9 billion CNY, respectively. Therefore, the fire department close the major entrances to the top risk subarea enhances the entrance management for the second top risk subarea, significantly reducing the potential fire risks.

Table 3. Resulting minimum, median, and standard deviation of the objective function values (i.e., total fire loss in thousand CNY) obtained by the algorithms on the test instances. Symbol [†] indicates that the result is significantly different from the result of EWWO at a confidence level of 95%.

Instance	Metrics	GA	PSO	DE	BBO	EBO	WWO	EWWO
1	minimum	788	787	782	786	782	779	778
	median	[†] 808	[†] 806	801	[†] 808	801	801	798
	std	17	11	15	12	8	15	16
2	minimum	936	932	927	930	923	919	916
	median	[†] 980	[†] 970	951	[†] 974	[†] 964	955	946
	std	29	27	15	35	21	31	25
3	minimum	1059	1052	1044	1057	1051	1046	1039
	median	[†] 1172	[†] 1139	[†] 1109	[†] 1136	[†] 1114	1096	1084
	std	62	55	36	59	40	34	29

Table 3. Cont.

Instance	Metrics	GA	PSO	DE	BBO	EBO	WWO	EWWO
4	minimum	1133	1132	1118	1132	1121	1110	1108
	median	[†] 1221	[†] 1207	[†] 1189	[†] 1203	[†] 1194	1180	1162
	std	79	53	43	45	46	50	28
5	minimum	1256	1235	1218	1260	1225	1204	1162
	median	[†] 1345	[†] 1345	[†] 1282	[†] 1325	[†] 1301	[†] 1253	1226
	std	57	49	28	41	60	31	49
6	minimum	1428	1394	1356	1405	1365	1329	1290
	median	[†] 1576	[†] 1514	[†] 1454	[†] 1564	[†] 1466	1392	1369
	std	67	78	51	75	74	46	54
7	minimum	1640	1640	1526	158	1517	1428	1354
	median	[†] 1788	[†] 1694	[†] 1641	[†] 1745	[†] 1622	[†] 1590	1428
	std	111	24	85	98	86	71	64
8	minimum	1807	1784	1642	1752	1671	1558	1517
	median	[†] 1980	^{1†} 897	[†] 1778	[†] 1889	[†] 1819	[†] 1699	1620
	std	97	50	84	90	110	75	70
9	minimum	1921	1874	1787	1927	1778	1710	1610
	median	[†] 2201	[†] 2135	[†] 1984	[†] 2193	[†] 2036	[†] 1852	1729
	std	230	226	127	188	219	127	64
10	minimum	675	674	672	674	672	667	665
	median	[†] 699	[†] 690	684	[†] 697	680	682	670
	std	14	11	8	17	6	9	4
11	minimum	778	771	757	775	765	752	749
	median	[†] 788	[†] 782	[†] 779	[†] 784	772	769	763
	std	9	6	15	6	4	14	8
12	minimum	822	816	804	822	805	803	793
	median	[†] 837	[†] 836	817	[†] 837	[†] 820	809	808
	std	6	17	9	13	6	3	13
13	minimum	946	920	895	931	889	872	832
	median	[†] 975	[†] 961	[†] 921	[†] 973	930	897	862
	std	18	23	13	34	26	17	14
14	minimum	1074	1062	992	1024	993	931	906
	median	[†] 1133	[†] 1112	[†] 1047	[†] 1128	[†] 1056	[†] 992	926
	std	50	35	44	79	52	32	12
15	minimum	1281	1243	1083	1205	1136	1115	990
	median	[†] 1340	[†] 1286	[†] 1232	[†] 1316	[†] 1185	[†] 1183	1039
	std	42	36	130	67	20	28	24
16	minimum	1468	1335	1230	1359	1264	1172	1128
	median	[†] 1556	[†] 1435	[†] 1331	[†] 1546	[†] 1365	[†] 1275	1221
	std	61	79	66	107	46	53	49
17	minimum	1625	1573	1356	1599	1427	1281	1251
	median	[†] 1739	[†] 1634	[†] 1424	[†] 1755	[†] 1565	[†] 1489	1335
	std	62	41	36	85	103	178	34
18	minimum	1891	1811	1587	1751	1591	1615	1423
	median	[†] 2024	[†] 2024	[†] 1789	[†] 2013	[†] 1769	[†] 1668	1502
	std	64	140	97	232	145	32	43
19	minimum	655	652	648	652	650	649	645
	median	[†] 655	[†] 653	[†] 651	[†] 655	[†] 651	[†] 649	645
	std	0	1	1	2	1	0	0
20	minimum	734	708	686	728	711	693	665
	median	[†] 758	[†] 738	[†] 708	[†] 757	[†] 716	[†] 707	685
	std	19	25	9	18	3	8	14
21	minimum	788	788	723	788	737	694	685
	median	[†] 822	[†] 807	[†] 742	[†] 813	[†] 763	[†] 746	694
	std	18	13	8	13	19	28	7

Table 3. Cont.

Instance	Metrics	GA	PSO	DE	BBO	EBO	WWO	EWWO
22	minimum	891	836	763	844	834	755	729
	median	[†] 970	[†] 937	[†] 833	[†] 927	[†] 848	[†] 820	749
	std	45	57	62	36	6	46	14
23	minimum	1010	953	891	968	908	865	837
	median	[†] 1148	[†] 1085	[†] 991	[†] 1101	[†] 1009	[†] 993	872
	std	119	60	43	82	49	101	26
24	minimum	1084	1043	1018	1071	1002	967	951
	median	[†] 1300	[†] 1291	[†] 1100	[†] 1287	[†] 1152	[†] 1095	985
	std	134	176	45	105	123	111	26
25	minimum	1325	1266	1201	1257	1155	1049	1015
	median	[†] 1492	[†] 1462	[†] 1250	[†] 1449	[†] 1293	[†] 1239	1059
	std	125	103	32	166	84	89	22
26	minimum	1586	1462	1353	1519	1405	1219	1069
	median	[†] 1714	[†] 1603	[†] 1427	[†] 1691	[†] 1556	[†] 1249	1128
	std	95	96	50	116	76	16	42
27	minimum	1852	1729	1381	1871	1620	1291	1207
	median	[†] 2014	[†] 1992	[†] 1620	[†] 1947	[†] 1671	[†] 1450	1285
	std	90	188	213	42	24	137	41
28	minimum	773	773	767	770	768	765	763
	median	[†] 788	[†] 784	784	[†] 789	[†] 785	780	778
	std	6	6	11	14	10	8	10
29	minimum	891	864	828	855	832	828	788
	median	[†] 916	[†] 882	[†] 868	[†] 900	[†] 845	[†] 863	803
	std	11	8	28	27	11	30	13
30	minimum	101	989	930	983	943	916	891
	median	[†] 1054	[†] 1031	[†] 973	[†] 1032	[†] 967	[†] 978	926
	std	33	19	25	22	18	35	17
31	minimum	1074	1028	993	1068	1021	966	926
	median	[†] 1133	[†] 1089	[†] 1052	[†] 1108	[†] 1034	[†] 983	936
	std	34	35	41	31	9	7	8
32	minimum	1246	1142	1041	1209	1066	1004	965
	median	[†] 1335	[†] 1264	[†] 1179	[†] 1278	[†] 1216	[†] 1138	995
	std	49	86	64	31	131	112	13
33	minimum	1404	1396	1244	1317	1285	1212	1133
	median	[†] 1492	[†] 1441	[†] 1348	[†] 1498	[†] 1333	[†] 1297	1172
	std	38	39	80	77	38	61	32
34	minimum	1507	1445	1369	1507	1403	1330	1226
	median	[†] 1591	[†] 1528	[†] 1460	[†] 1582	[†] 1478	[†] 1374	1276
	std	62	66	69	56	54	26	22
35	minimum	1630	1560	1453	1529	1534	1386	1349
	median	[†] 1734	[†] 1675	[†] 1597	[†] 1703	[†] 1604	[†] 1541	1423
	std	82	78	82	144	42	75	61
36	minimum	1881	1722	1551	1894	1726	1622	1463
	median	[†] 1965	[†] 1880	[†] 1746	[†] 1957	[†] 1834	[†] 1667	1542
	std	38	86	175	47	85	30	46
37	minimum	896	896	894	895	893	893	891
	median	[†] 906	[†] 906	900	[†] 906	902	897	896
	std	7	7	4	7	7	2	4
38	minimum	985	970	962	969	959	955	936
	median	[†] 101	[†] 995	[†] 976	[†] 1001	[†] 983	[†] 975	951
	std	12	18	7	200	16	11	6
39	minimum	1019	1015	995	998	998	984	960
	median	[†] 1044	[†] 1028	[†] 1004	[†] 1033	[†] 1005	[†] 993	965
	std	13	8	7	14	6	7	3

Table 3. Cont.

Instance	Metrics	GA	PSO	DE	BBO	EBO	WWO	EWWO
40	minimum	1064	1051	1035	1054	1046	1031	1024
	median	[†] 1093	[†] 1077	[†] 1054	[†] 1074	[†] 1066	1054	1034
	std	23	20	14	15	15	20	7
41	minimum	1305	1228	1188	1239	1167	1111	1039
	median	[†] 1428	[†] 1368	[†] 1282	[†] 1410	[†] 1232	[†] 1247	1069
	std	86	105	51	83	37	77	19
42	minimum	1497	1494	1339	1481	1286	1253	1167
	median	[†] 1571	[†] 1546	[†] 1397	[†] 1571	[†] 1383	[†] 1347	1197
	std	35	27	44	80	85	56	25
43	minimum	1556	1524	1441	1459	1406	1373	1261
	median	[†] 1645	[†] 1572	[†] 1523	[†] 1661	[†] 1514	[†] 1460	1325
	std	54	37	33	99	87	49	51
44	minimum	1847	1847	1595	1821	1582	1469	1413
	median	[†] 1985	[†] 1895	[†] 1789	[†] 2004	[†] 1741	[†] 1636	1507
	std	118	39	89	92	91	73	42
45	minimum	2029	1898	1679	2040	1827	1537	1482
	median	[†] 2197	[†] 2083	[†] 1907	[†] 2191	[†] 2001	[†] 1696	1571
	std	141	98	119	107	110	94	75
46	minimum	749	749	749	749	749	749	749
	median	[†] 754	[†] 754	[†] 752	[†] 753	751	750	749
	std	4	4	2	3	1	0	0
47	minimum	788	787	783	788	785	781	778
	median	[†] 818	[†] 816	[†] 811	[†] 816	806	805	803
	std	14	14	23	13	14	12	20
48	minimum	906	902	874	896	863	847	837
	median	[†] 960	[†] 956	[†] 901	[†] 950	[†] 904	[†] 914	877
	std	31	45	19	28	21	37	28
49	minimum	1182	1169	1102	1142	1082	1047	1000
	median	[†] 1271	[†] 1223	[†] 1194	[†] 1236	[†] 1150	[†] 1130	1069
	std	78	48	81	79	57	73	62
50	minimum	1458	1393	1325	1433	1257	1204	1148
	median	[†] 1566	[†] 1527	[†] 1417	[†] 1580	[†] 1372	[†] 1282	1207
	std	64	66	40	69	54	53	28
51	minimum	1566	1456	1420	1529	1417	1301	1261
	median	[†] 1694	[†] 1631	[†] 1464	[†] 1694	[†] 1500	[†] 1476	1325
	std	99	121	23	147	5	152	53
52	minimum	1660	1600	1529	1632	1585	1540	1473
	median	[†] 1773	[†] 1695	[†] 1627	[†] 1731	[†] 1701	[†] 1610	1556
	std	84	74	52	46	73	39	73
53	minimum	1872	1798	1604	1795	1681	1597	1537
	median	[†] 1990	[†] 1983	[†] 1704	[†] 1986	[†] 1755	[†] 1765	1615
	std	55	75	44	140	53	104	63
54	minimum	2049	1919	1903	2041	1864	1694	1655
	median	[†] 2157	[†] 2057	[†] 2007	[†] 2104	[†] 1993	[†] 1875	1748
	std	84	99	90	36	68	147	62
55	minimum	857	855	836	851	837	822	818
	median	[†] 882	[†] 867	[†] 850	[†] 882	[†] 865	[†] 848	832
	std	15	9	7	23	14	15	6
56	minimum	965	946	911	940	937	925	896
	median	[†] 995	[†] 973	[†] 970	[†] 981	[†] 941	[†] 940	926
	std	14	14	41	21	2	10	18
57	minimum	1251	1243	1181	1256	1171	1122	1098
	median	[†] 1310	[†] 1276	[†] 1224	[†] 1302	[†] 1227	[†] 1172	1133
	std	28	21	32	23	47	21	18

Table 3. Cont.

Instance	Metrics	GA	PSO	DE	BBO	EBO	WWO	EWWO
58	minimum	1522	1442	1280	1485	1378	1286	1216
	median	[†] 1591	[†] 1518	[†] 1348	[†] 1604	[†] 1430	[†] 1358	1276
	std	37	50	47	92	45	57	29
59	minimum	1660	1647	1449	1540	1504	1416	1335
	median	[†] 1783	[†] 1727	[†] 1567	[†] 1794	[†] 1618	[†] 1570	1423
	std	87	45	68	197	80	127	69
60	minimum	1995	1937	1883	1886	1825	1729	1675
	median	[†] 2142	[†] 2066	[†] 2004	[†] 2056	[†] 1942	[†] 1918	1768
	std	104	74	62	92	67	116	71
61	minimum	2197	2083	2092	2194	2037	1932	1896
	median	[†] 2325	[†] 2275	[†] 2129	[†] 2278	[†] 2165	[†] 2023	1970
	std	54	149	25	69	77	81	44
62	minimum	2448	2332	2217	2336	2300	2141	2049
	median	[†] 2610	[†] 2489	[†] 2382	[†] 2579	[†] 2384	[†] 2333	2162
	std	138	80	119	166	67	103	60
63	minimum	2837	2652	2363	2819	2584	2369	2221
	median	[†] 3058	[†] 2860	[†] 2648	[†] 3043	[†] 2774	[†] 2486	2325
	std	157	171	252	99	161	85	57
64	minimum	650	650	650	650	650	650	650
	median	[†] 660	[†] 659	[†] 654	[†] 659	[†] 654	[†] 654	650
	std	7	6	3	7	3	2	0
65	minimum	773	772	771	772	771	770	768
	median	[†] 803	[†] 790	[†] 791	[†] 801	[†] 786	[†] 773	768
	std	26	11	17	13	7	3	0
66	minimum	975	952	878	948	911	872	852
	median	[†] 1024	[†] 1019	[†] 967	[†] 1024	[†] 956	[†] 925	867
	std	21	46	63	48	30	35	13
67	minimum	1281	1254	1187	1254	1238	1164	1152
	median	[†] 1345	[†] 1325	[†] 1252	[†] 1336	[†] 1293	1215	1212
	std	56	59	27	42	27	43	46
68	minimum	1409	1392	1314	1392	1307	1284	1236
	median	[†] 1502	[†] 1463	[†] 1361	[†] 1487	[†] 1376	[†] 1376	1285
	std	55	29	30	79	60	45	38
69	minimum	1670	1595	1552	1631	1544	1533	1463
	median	[†] 1788	[†] 1730	[†] 1632	[†] 1788	[†] 1655	[†] 1599	1532
	std	101	87	53	114	84	46	34
70	minimum	1975	1853	1818	1940	1769	1741	1625
	median	[†] 2108	[†] 2026	[†] 1937	[†] 2036	[†] 1894	[†] 1884	1704
	std	111	110	98	47	76	78	33
71	minimum	2098	2032	1978	2047	1938	1893	1832
	median	[†] 2231	[†] 2153	[†] 2089	[†] 2227	[†] 2064	[†] 1964	1916
	std	104	107	66	135	60	33	56
72	minimum	2197	2128	1996	2197	2042	1965	1921
	median	[†] 2369	[†] 2275	[†] 2067	[†] 2326	[†] 2146	[†] 2044	2004
	std	126	130	48	52	54	46	43

Table 4. Summary of the experimental results of the seven algorithms on the test instances.

Metrics	GA	PSO	DE	BBO	EBO	WWO	EWWO
AVG(Minimum)	1348	1303.5	1227.6	1318	1246.7	1187.3	1140.7
RANK(Minimum)	68	54	33	56	37	22	11
AVG(Median)	1436.5	1391.8	1307.3	1420.3	1321.4	1265.9	1189.7
RANK(Median)	68	52	32	60	36	23	10

Although the seven algorithms use the same number of objective function evaluations, there are still slight differences among their running times. Figure 7 presents the median running time of each of the seven algorithms on instances 1–9 (the time variations are similar on the remaining instances). Moreover, Figure 8 presents the median convergence time of each algorithm on the instances (we consider). As we can observe, with the increase in instance size, the convergence time of EWWO becomes more significantly shorter than that of the other algorithms.

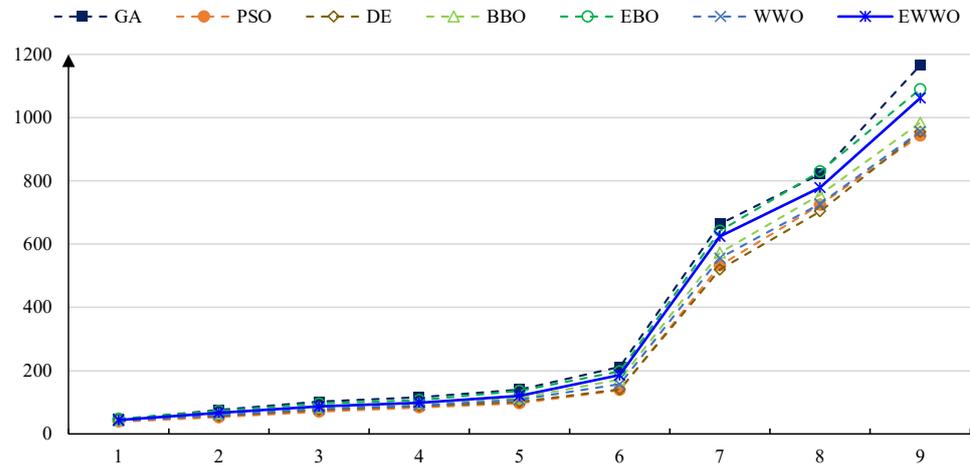


Figure 7. Median running time of each of the seven algorithms on instances 1–9.

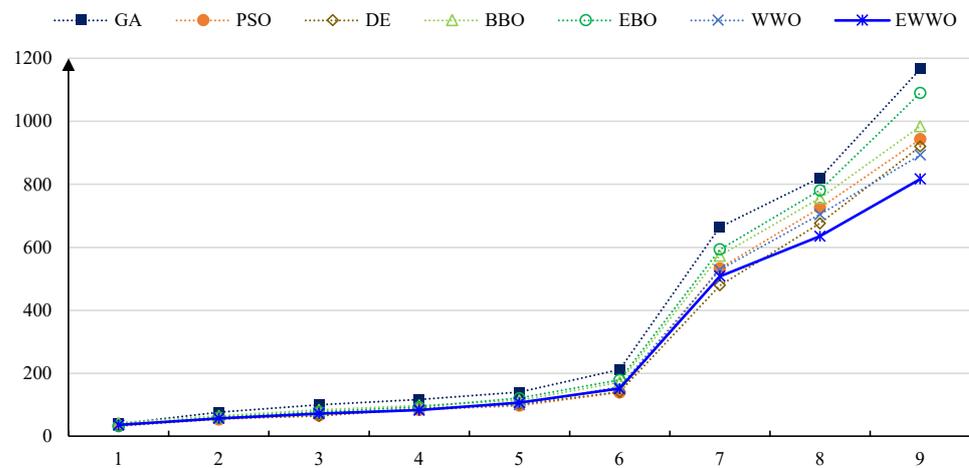


Figure 8. Median convergence time of each of the seven algorithms on instances 1–9.

5. Conclusions

This paper presents a study consisting of a mathematical model that predicts wildfire spread and the corresponding economic losses that work simultaneously; an optimization problem of firefighting drones scheduling to minimize the fire loss based on the wildfire spread and loss estimation; and a metaheuristic optimization method, in particular, the EWWO algorithm for efficiently solving the optimization problem. The results of the model and algorithms can also help us to configure the drones and staff to support efficient firefighting operations and well as prepare a set of plans in response to possible wildfire occurrences. Validation of the fire spread model on the two real-world wildfires showed that the MPAE of full combustion time and decay time were around 7~8%, respectively. Experiments of the scheduling algorithms on a set of 72 test instances demonstrated that the solutions obtained by the proposed algorithm of EWWO could reduce approximately 100,000~600,000 CNY compared to the solutions of other popular algorithms.

The present work has certain limitations. First, we only estimate the losses caused by fire but do not consider the cost of drones in configuration and firefighting. Second, the proposed method limits that the fire in a subarea should be extinguished at a time by a small number of drones, which is not scalable to large areas that need multiple batches of drones for firefighting. Third, the scheduling schema requires that the distance between the fire station (configured with drones) and the utmost subarea cannot exceed the half of the flight range of the drone. Therefore, for large wild areas, we need to set up more fire stations.

In the future work, we will consider the cooperation of drones and ground vehicles [55,56], where ground vehicles carry water and batteries (or support fast battery recharging) for drones to shorten the travel distances of drones and improve the firefighting efficiency. Our ongoing work also considers integrating reinforcement learning into the optimization to improve the problem-solving performance [57].

Author Contributions: Conceptualization, Y.-J.Z.; methodology, R.-Y.W.; software, R.-Y.W.; investigation, X.-C.X.; validation, X.-C.X.; writing—original draft preparation, R.-Y.W.; writing—review and editing, Y.-J.Z.; funding acquisition, Y.-J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (grant numbers 61872123, 62372148) and the Natural Science Foundation of Zhejiang Province (grant number LR20F030002).

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used in this paper are available at <https://www.compintell.cn/en/dataAndCode.html> (accessed on 26 December 2023)

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

References

1. Gabbert, B. A 9-Year USFS Aerial Firefighting Study Left Many Questions Unanswered. Wildfiretoday. 2021. Available online: <https://wildfiretoday.com/2021/04/16/a-9-year-usfs-aerial-firefighting-study-left-many-questions-unanswered/> (accessed on 16 December 2023).
2. Restas, A. Examining the effectiveness of aerial firefighting with the components of firebreak requirements and footprint geometry—Critics of the present practice. *Fire* **2023**, *6*, 351. [CrossRef]
3. Qin, H.; Cui, J.Q.; Li, J.; Bi, Y.; Lan, M.; Shan, M.; Liu, W.; Wang, K.; Lin, F.; Zhang, Y.F.; et al. Design and implementation of an unmanned aerial vehicle for autonomous firefighting missions. In Proceedings of the IEEE International Conference on Control and Automation, Kathmandu, Nepal, 1–3 June 2016; pp. 62–67. [CrossRef]
4. Alon, O.; Rabinovich, S.; Fyodorov, C.; Cauchard, J.R. Drones in firefighting: A user-centered design perspective. In Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction, Toulouse, France, 27 September–1 October 2021; ACM: New York, NY, USA, 2021; MobileHCI'21. [CrossRef]
5. Roldán-Gómez, J.J.; González-Girona, E.; Barrientos, A. A survey on robotic technologies for forest firefighting: Applying drone swarms to improve firefighters' efficiency and safety. *Appl. Sci.* **2021**, *11*, 363. [CrossRef]
6. Laszlo, B.; Agoston, R.; Xu, Q. Conceptual approach of measuring the professional and economic effectiveness of drone applications supporting forest fire management. In Proceedings of the 8th International Conference on Fire Science and Fire Protection Engineering, Nanjing, China, 27–29 October 2017; Procedia Engineering; 2018, Volume 211, pp. 8–17. [CrossRef]
7. Aydin, B.; Selvi, E.; Tao, J.; Starek, M.J. Use of fire-extinguishing balls for a conceptual system of drone-assisted wildfire fighting. *Drones* **2019**, *3*, 17. [CrossRef]
8. Kataeva, L.Y.; Ilicheva, M.N.; Loshchilov, A.A. Mathematical modeling for extinguishing forest fires using water capsules with a thermoactive shell. *J. Appl. Mech. Tech. Phy.* **2022**, *63*, 1227–1242. [CrossRef]
9. Beijing ZhongHangZhi Technology Co., Ltd. TD220 Coaxial Unmanned Helicopter. Available online: <https://www.zhz.com/en> (accessed on 23 October 2023).
10. Viegas, C.; Chehreh, B.; Andrade, J.; Lourenço, J. Tethered UAV with combined multi-rotor and water jet propulsion for forest fire fighting. *J. Intell. Robot. Syst.* **2021**, *104*. [CrossRef]
11. Śmigielski, G.; Dygdała, R.; Zarzycki, H.; Lewandowski, D. Real-time system of delivering water-capsule for firefighting. In *Hard and Soft Computing for Artificial Intelligence, Multimedia and Security*; Kobayashi, S.Y., Piegat, A., Pejaś, J., El Fray, I., Kacprzyk, J., Eds.; Springer: Cham, Switzerland, 2017; pp. 102–111. [CrossRef]

12. Zheng, L.; Quan, W. Experimental study of explosive water mist extinguishing fire. In Proceedings of the 5th Conference Performance-based Fire and Fire Protection Engineering, Guangzhou, China, 7–9 December 2010; Procedia Engineering; 2011, Volume 11, pp. 258–267. [\[CrossRef\]](#)
13. Śmigielski, G.; Toczek, W.; Dygdała, R.; Stefański, K. Metrological analysis of precision of the system of delivering a water capsule for explosive production of water aerosol. *Metrol. Measure. Syst.* **2016**, *23*, 47–58. [\[CrossRef\]](#)
14. Czerniak, J.; Śmigielski, G.; Ewald, D.; Paprzycki, M.; Dobrosielski, W. New proposed implementation of ABC method to optimization of water capsule flight. In Proceedings of the Federated Conference on Computer Science and Information Systems, Lodz, Poland, 13–16 September 2015; pp. 489–493. [\[CrossRef\]](#)
15. Czerniak, J.M.; Ewald, D.; Śmigielski, G.; Dobrosielski, W.T.; Apiecionek, Ł. Optimization of fuel consumption in firefighting water capsule flights of a helicopter. In *Recent Advances in Computational Optimization: Results of the Workshop on Computational Optimization*; Fidanova, S., Ed.; Springer: Cham, Switzerland, 2016; pp. 39–49. [\[CrossRef\]](#)
16. Wang, X.; Liu, H.; Tian, Y.; Chen, Z.; Cai, Z. A fast optimization method of water-dropping scheme for fixed-wing firefighting aircraft. *IEEE Access* **2021**, *9*, 120815–120832. [\[CrossRef\]](#)
17. Zheng, Y.J.; Du, Y.C.; Sheng, W.G.; Ling, H.F. Collaborative human-UAV search and rescue for missing tourists in nature reserves. *INFORMS J. Appl. Anal.* **2019**, *49*, 371–383. [\[CrossRef\]](#)
18. Du, Y.; Zhang, M.; Ling, H.; Zheng, Y. Evolutionary planning of multi-UAV search for missing tourists. *IEEE Access* **2019**, *7*, 73480–73492. [\[CrossRef\]](#)
19. Kiani, F.; Seyyedabbasi, A.; Aliyev, R.; Gulle, M.U.; Basyildiz, H.; Shah, M.A. Adapted-RRT: Novel hybrid method to solve three-dimensional path planning problem using sampling and metaheuristic-based algorithms. *Neural Comput. Appl.* **2021**, *33*, 15569–15599. [\[CrossRef\]](#)
20. Kiani, F.; Seyyedabbasi, A.; Nematzadeh, S.; Candan, F.; Çevik, T.; Anka, F.A.; Randazzo, G.; Lanza, S.; Muzirafuti, A. Adaptive metaheuristic-based methods for autonomous robot path planning: Sustainable agricultural applications. *Appl. Sci.* **2022**, *12*, 943. [\[CrossRef\]](#)
21. Zheng, Y.J.; Chen, S.Y.; Ling, H.F. Evolutionary optimization for disaster relief operations: A survey. *Appl. Soft Comput.* **2015**, *27*, 553–566. [\[CrossRef\]](#)
22. Luo, Z.; Zhang, Y.; Mu, L.; Huang, J.; Xin, J.; Liu, H.; Jiao, S.; Xie, G.; Yi, Y. A UAV path planning algorithm based on an improved D* Lite algorithm for forest firefighting. In Proceedings of the Chinese Automation Congress, Shanghai, China, 6–8 November 2020; pp. 4233–4237. [\[CrossRef\]](#)
23. Harikumar, K.; Senthilnath, J.; Sundaram, S. Multi-UAV oxyrrhis marina-inspired search and dynamic formation control for forest firefighting. *IEEE Trans. Autom. Sci. Eng.* **2019**, *16*, 863–873. [\[CrossRef\]](#)
24. Hong, L.; Wang, Y.; Du, Y.; Chen, X.; Zheng, Y. UAV search-and-rescue planning using an adaptive memetic algorithm. *Front. Inf. Technol. Electron. Eng.* **2021**, *22*, 1477–1491. [\[CrossRef\]](#)
25. Zheng, Y.; Du, Y.; Ling, H.; Sheng, W.; Chen, S. Evolutionary collaborative human-UAV search for escaped criminals. *IEEE Trans. Evol. Comput.* **2020**, *24*, 217–231. [\[CrossRef\]](#)
26. Zheng, Y.J.; Du, Y.C.; Su, Z.L.; Ling, H.F.; Zhang, M.X.; Chen, S.Y. Evolutionary human-UAV cooperation for transmission network restoration. *IEEE Trans. Ind. Informat.* **2021**, *17*, 1648–1657. [\[CrossRef\]](#)
27. Wang, C.; Liu, P.; Zhang, T.; Sun, J. The adaptive vortex search algorithm of optimal path planning for forest fire rescue UAV. In Proceedings of the IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference, Chongqing China, 12–14 October 2018; pp. 400–403. [\[CrossRef\]](#)
28. Xiang, A.; Wang, L. Research on path planning of UAV forest fire fighting based on improved ant colony algorithm. In Proceedings of the 2021 7th International Conference on Computing and Artificial Intelligence, Tianjin, China, 23–26 April 2021; ACM: New York, NY, USA, 2021; pp. 289–295. [\[CrossRef\]](#)
29. Xu, Y.; Li, J.; Zhang, F. A UAV-based forest fire patrol path planning strategy. *Forests* **2022**, *13*, 1952. [\[CrossRef\]](#)
30. Alsammak, I.L.H.; Mahmoud, M.A.; Gunasekaran, S.S.; Ahmed, A.N.; AlKilabi, M. Nature-inspired drone swarming for wildfires suppression considering distributed fire spots and energy consumption. *IEEE Access* **2023**, *11*, 50962–50983. [\[CrossRef\]](#)
31. Xavier Viegas, D. Forest fire propagation. *Philos. Trans. Royal Soc. London. Series A* **1998**, *356*, 2907–2928. [\[CrossRef\]](#)
32. Li, J.; Li, X.; Chen, C.; Zheng, H.; Liu, N. Three-dimensional dynamic simulation system for forest surface fire spreading prediction. *Int. J. Pattern Recogn. Artif. Intell.* **2018**, *32*, 1850026. [\[CrossRef\]](#)
33. Alexandridis, A.; Vakalis, D.; Siettos, C.; Bafas, G. A cellular automata model for forest fire spread prediction: The case of the wildfire that swept through Spetses Island in 1990. *Appl. Math. Comput.* **2008**, *204*, 191–201. [\[CrossRef\]](#)
34. Rui, X.; Hui, S.; Yu, X.; Zhang, G.; Wu, B. Forest fire spread simulation algorithm based on cellular automata. *Nat. Hazards* **2018**, *91*, 309–319. [\[CrossRef\]](#)
35. Mutthulakshmi, K.; Wee, M.R.E.; Wong, Y.C.K.; Lai, J.W.; Koh, J.M.; Acharya, U.R.; Cheong, K.H. Simulating forest fire spread and fire-fighting using cellular automata. *Chin. J. Physics* **2020**, *65*, 642–650. [\[CrossRef\]](#)
36. Wu, Z.; Wang, B.; Li, M.; Tian, Y.; Quan, Y.; Liu, J. Simulation of forest fire spread based on artificial intelligence. *Ecol. Ind.* **2022**, *136*, 108653. [\[CrossRef\]](#)
37. Cheng, H.; Hadjisophocleous, G.V. The modeling of fire spread in buildings by Bayesian network. *Fire Saf. J.* **2009**, *44*, 901–908. [\[CrossRef\]](#)

38. Shaham, Y.; Benenson, I. Modeling fire spread in cities with non-flammable construction. *Int. J. Disaster Risk Reduct.* **2018**, *31*, 1337–1353. [[CrossRef](#)]
39. Zheng, Y.J. Water wave optimization: A new nature-inspired metaheuristic. *Comput. Oper. Res.* **2015**, *55*, 1–11. [[CrossRef](#)]
40. Weber, R.O. Toward a comprehensive wildfire spread model. *Int. J. Wildland Fire* **1991**, *1*, 245–248. [[CrossRef](#)]
41. Cheng, H.; Hadjisophocleous, G.V. Dynamic modeling of fire spread in building. *Fire Saf. J.* **2011**, *46*, 211–224. [[CrossRef](#)]
42. Wang, L.; Zhang, L.; Zheng, D.Z. A class of order-based genetic algorithm for flow shop scheduling. *Int. J. Adv. Manuf. Technol.* **2003**, *22*, 828–835. [[CrossRef](#)]
43. Liao, C.J.; Tseng, C.T.; Luarn, P. A discrete version of particle swarm optimization for flowshop scheduling problems. *Comput. Oper. Res.* **2007**, *34*, 3099–3111. [[CrossRef](#)]
44. Liang, J.J.; Qin, A.K.; Suganthan, P.; Baskar, S. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. Evol. Comput.* **2006**, *10*, 281–295. [[CrossRef](#)]
45. Zheng, Y.; Ling, H.; Xue, J.; Chen, S. Population classification in fire evacuation: A multiobjective particle swarm optimization approach. *IEEE Trans. Evol. Comput.* **2014**, *18*, 70–81. [[CrossRef](#)]
46. Zheng, Y.J.; Ling, H.F.; Guan, Q. Adaptive Parameters for a Modified Comprehensive Learning Particle Swarm Optimizer. *Math. Prob. Eng.* **2012**, *2012*, 207318. [[CrossRef](#)]
47. Chakraborty, U.K.; Turvey, K.P. Floating-point to integer mapping schemes in differential evolution for permutation flow shop scheduling. *Int. J. Bio-Inspired Comput.* **2010**, *2*, 183–204. [[CrossRef](#)]
48. Zheng, Y.; Lu, X.; Zhang, M.; Chen, S. *Biogeography-Based Optimization: Algorithms and Applications*; Springer: Berlin/Heidelberg, Germany, 2018. [[CrossRef](#)]
49. Zheng, Y.; Ling, H.; Xue, J. Disaster rescue task scheduling: An evolutionary multiobjective optimization approach. *IEEE Trans. Emerg. Top. Comput.* **2018**, *6*, 288–300. [[CrossRef](#)]
50. Zheng, Y.J.; Ling, H.F.; Xue, J.Y. Ecogeography-based optimization: Enhancing biogeography-based optimization with ecogeographic barriers and differentiations. *Comput. Oper. Res.* **2014**, *50*, 115–127. [[CrossRef](#)]
51. Du, Y.C.; Zhang, M.X.; Cai, C.Y.; Zheng, Y.J. Enhanced biogeography-based optimization for flow-shop scheduling. In *Bio-Inspired Computing: Theories and Applications*; Qiao, J., Zhao, X., Pan, L., Zuo, X., Zhang, X., Zhang, Q., Huang, S., Eds.; Springer: Singapore, 2018; pp. 295–306.
52. Zheng, Y.J.; Lu, X.Q.; Du, Y.C.; Xue, Y.; Sheng, W.G. Water wave optimization for combinatorial optimization: Design strategies and applications. *Appl. Soft Comput.* **2019**, *83*, 105611. [[CrossRef](#)]
53. Zheng, Y.J.; Zhang, B. A simplified water wave optimization algorithm. In Proceedings of the IEEE Congress on Evolutionary Computation, Sendai, Japan, 25–28 May 2015; pp. 807–813. [[CrossRef](#)]
54. Nawaz, M.; Enscore, E.E.; Ham, I. A heuristic algorithm for the m -machine, n -job flow-shop sequencing problem. *Omega* **1983**, *11*, 91–95. [[CrossRef](#)]
55. Weng, Y.Y.; Wu, R.Y.; Zheng, Y.J. Cooperative truck-drone delivery path optimization under urban traffic restriction. *Drones* **2023**, *7*, 59. [[CrossRef](#)]
56. Wu, G.; Mao, N.; Luo, Q.; Xu, B.; Shi, J.; Suganthan, P.N. Collaborative truck-drone routing for contactless parcel delivery during the epidemic. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 25077–25091. [[CrossRef](#)]
57. Siddiqui, A.B.; Aqeel, I.; Alkhayyat, A.; Javed, U.; Kaleem, Z. Prioritized user association for sum-rate maximization in UAV-assisted emergency communication: A reinforcement learning approach. *Drones* **2022**, *6*, 45. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.