A Combinatorial Optimization Model for Emergency Resource Allocation after Disasters

Sehej Jain (✉ sehejjain@outlook.com)

PDPM IITDM: PDPM Indian Institute of Information Technology Design and Manufacturing Jabalpur
<https://orcid.org/0000-0002-7265-7648>

Kusum Kumari Bharti

PDPM IITDM: PDPM Indian Institute of Information Technology Design and Manufacturing Jabalpur

Research Article

Keywords: Disaster Management, Emergency Resource Allocation, Metaheuristics, Combinatorial Optimization

Posted Date: December 9th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1140300/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

A Combinatorial Optimization Model for Emergency Resource Allocation after Disasters

Sehej Jain and Kusum Kumari Bharti[†]

PDPM Indian Institute of Information Technology Design and Manufacturing, Jabalpur,
Madhya Pradesh, India.

*Corresponding author(s). E-mail(s): sehejjain@outlook.com;

Contributing authors: kusum@iiitdmj.ac.in;

[†]These authors contributed equally to this work.

Abstract

Disasters occur over a short or long period of time and cause large-scale harm to humans, infrastructure, as well as the ecosystem every year. Immediate response after a disaster helps minimize its impact on life and property. Therefore, it is crucial to have an emergency response system ready to handle any emergency that may come up after a disaster. In this paper, a model is proposed to optimize the distribution of emergency services at disaster-struck points. Due to the *NP-hardness* of the problem, two metaheuristic algorithms, Particle Swarm Optimization and Cuckoo Search Optimization have been used to dynamically allocate the available resources based on the given situation. The proposed model uses the distance between the emergency location and the emergency service provider, and the severity of the emergency as the main metrics for scoring any considered solution. The conducted experiments demonstrate that the model provides effective, efficient, and dynamic allocation service at emergency locations in simulated disaster situations.

Keywords: Disaster Management, Emergency Resource Allocation, Metaheuristics, Combinatorial Optimization

1 Introduction

Natural Disasters are large-scale harm-causing events that occur due to the natural processes of the earth. Every year, disasters cause harm to human life as well as nature itself. They cause a huge number of casualties along with economic losses measured in billions of dollars annually. In 1998-2017, the economic loss caused by natural disasters was US\$ 2,908 billion¹. Substantial loss of human life and infrastructure raises the urgent

need of addressing such issues caused by natural disasters. Disaster management can be considered as a four-phase process: Mitigation, Preparedness, Response, and Recovery [1].

Activities and Challenges in Natural Disaster Management are classified into[1–3]:

1. Preparation Phase: Before the Disaster
2. Response Phase: During and immediately after a disaster
3. Recovery Phase: Long time after a disaster

During the preparation stage, activities such as reinforcing existing infrastructure and setting

¹Source: United Nations Office for Disaster Risk Reduction
<https://www.unisdr.org/2016/iddr/IDDR2018.Economic%20Losses.pdf>

up emergency warning systems, protocols, and services are done[4–8].

During the response stage, activities like evacuation, relief operations, emergency shelter, and health operations are carried out. Research in this phase is aimed at handling the situation to minimize loss of life and damage to infrastructure [9–12].

During the recovery stage, activities like repair work, finding missing people, and providing emergency services are carried out. Research in this phase attempts to aid this recovery phase and get the disaster-struck area back to normalcy as quickly as possible[13–16].

After these three phases, we come back to the Mitigation Phase in which we learn from past disasters and make sure that appropriate policies and plans are in place so that the next time a disaster strikes, its effect on human life, infrastructure, and ecosystem is minimal.

In this paper, the Response phase of Disaster Management has been addressed. The emergencies that arise after a disaster could be medical or fire emergencies, crowd control, infrastructure collapse, urgent resource requirements, etc. Primarily, this phase attempts to allocate resources to emergency sites to reduce the losses as much as possible caused due to a disaster. It is a post-disaster activity that works on the allocation of resources just after the occurrence of a disaster. Dealing with emergencies as they come up and optimally using the resources available on hand reduces the damage caused by the disaster and is a critical step in disaster management.

To deal with the situation, different response units are assigned to different emergency locations. As there exists no unit that is capable of dealing with all kinds of emergencies, the problem to allocate these units becomes a tedious task. Units must be allocated to all emergencies on the condition that the unit must be capable of dealing with the emergency allotted to it. In traditional models of disaster management, a simple severity factor is used and units are allocated simply based on these severity values. This process was done manually and without any decision support model (DSM). [12]

The past few years have been seen the development of Decision Support Systems (DSS) which are aimed at helping to make better decisions at

the time of difficulty especially in case of emergency. Kondaveti et al.[17] proposed a decision support system for resource allocation in disaster management built on rapid information collection and resource tracking functionalities. This was intended to be used by emergency managers. Fikar et al. [18] proposed a simulation and optimization-based DSS for coordinated disaster relief distribution. The goal of this paper was to have a DSS that facilitates disaster relief coordination between private and relief organizations. Hadigunaa et al. [19] created a web based DSS for disaster logistics based on a case study in Indonesia. The DSS aimed to assess the extent to which public facilities can be used as evacuation centers for the victims of an earthquake and/or tsunami. Jung et al. [20] proposed an intelligent DSS for smart city disaster management. They proposed a new conceptual framework of an intelligent DSS for disaster management, with particular attention paid to wildfires and cold and heatwaves. Turgut et al.[21] proposed a fuzzy AHP based DSS for disaster center location selection. They based this DSS on the analytic hierarchy/-fuzzy analytic hierarchy process method. Rakes et al. [22] proposed a DSS for post disaster housing. This DSS assigned families to interim housing units. Mete et al. [23] proposed a methodology for optimization of medical supply location and distribution in disaster management. Nayeri et al. [12] proposed a fatigue effect-based model for the scheduling of rescue units in case of disasters. The authors compared the rescue operations that happen immediately after a disaster to a job scheduling optimization problem and used severity and time taken to reach and attend to the emergency as factors affecting the optimum solution. Researchers have also developed multicriteria approaches, i.e., competitive and cooperative mechanisms. Fiedrich et al. introduced optimization models for the NDM problem [24]. Another method that has been explored is the application of artificial intelligence in decision support systems [25, 26].

In literature, the use of different optimization algorithms has been suggested to solve combinatorial optimization problems such as Genome Sequencing [27], Knapsack Problem [28, 29], Travelling Salesman Problem [30], and the Minimum Spanning Tree problem [31], etc. The disaster

Fig. 1 Relationship between the 4 phases of disaster management

management problem also comes under this category. In past literature, the use of metaheuristic algorithms has been suggested to solve various problems in disaster management [32–34].

Multiple disaster support systems have been proposed to manage the challenges and activities. These systems model the problem in different ways and use applied statistical methods, probability theory, and mathematical programming approaches to solve them. In this paper, the proposed model uses multiple factors to determine which response unit is suitable for a particular emergency situation. For this, a fitness score is allotted to a possible allocation of units. The objective of the problem then becomes to minimize this fitness value. This can be considered as an NP-hard problem, which is why metaheuristic algorithms are suited for this problem.

The remainder of this paper is organized as follows. Section 2 represents the formulation of the problem statement. Section 3 represents the formulation of the proposed mathematical model. Section 4 represents the results and analysis of the proposed model. Section 5 represents the conclusion and future work.

2 Problem Statement

As dealing with emergencies that come up immediately after a disaster is crucially important, the

designed model should be as close as possible to a real-life situation. To model the situation, there are a few entities that must be translated into objects in the model. The entities are the emergency itself, the emergency service location, the unit that deals with the emergency, and the optimum allocation of these units. Detailed description of each object is given below:

2.1 Objects in the Model

2.1.1 Emergency

Each Emergency object represents one emergency that the model must allocate units to. This object consists of the coordinates of the site of emergency, the severity of the emergency, and the type of aid required.

2.1.2 Emergency Service Location

The Emergency Service Location represents the home of the units that would be deployed to deal with an Emergency. This could be a hospital, a fire station, a police station, etc. Each Emergency Service Location Object records the location of the service provider, a specific number of unit objects, and the type of services available.

2.1.3 Unit

A Unit object represents the entity that deals with the emergency. It could be an ambulance, a paramedic vehicle, a police department vehicle, a fire engine, etc. Each object records the type of emergency unit and the maximum severity value that the unit can deal with.

2.1.4 Allocation

Each possible solution is considered as an allocation object. It consists of the mapping of the selected units and emergencies along with the details of which unit is associated with which emergency. The goal of the optimization problem then becomes to achieve an allocation object with the best fitness score (Refer, Section 3.1).

Moreover, a small number of units must be kept reserved for future emergencies. Hence, each of the objects and criteria which have been discussed is formulated in such a way so that we can represent the relationships as closely as possible to real-life.

The proposed representation fits the skeleton of a Combinatorial Optimization Problem that can be subjected to metaheuristic algorithms. Using metaheuristic algorithms, near-optimum solutions can be obtained for this model.

2.2 Smallest Position Value (SPV) Rule

The proposed model is a permutation-based problem and thus is discrete. By converting this problem to a continuous one, any optimization algorithm can be used to solve it. Here, the SPV rule is used to do this. The Smallest Position Value (SPV) rule is used to represent one permutation of the given position vector through an array of numbers. In this problem, $X_{id} = x_1, x_2 \dots x_n$ represents a potential candidate solution for the considered problem. Here, $x_1, x_2 \dots x_n$ are entities and not numbers. The SPV Rule converts it into a position vector. Figure 2 shows an example of the SPV rule.

2.3 Confluence

After representation, the next phase of the model is the optimum allocation of the resource unit

Let the Position Vector of a particular 5-dimension vector be:

2.34	4.63	-8.45	0.64	-2.54
0	1	2	3	4

Let the original Sequence Vector be:

f_1	f_2	f_3	f_4	f_5
0	1	2	3	4

According to the SPV rule, we assign the first fragment to the shortest value in the position vector and so on

So the Sequence that the particle in question represents becomes:

f_3	f_5	f_4	f_2	f_1
0	1	2	3	4

Fig. 2 SPV Rule

to the emergency locations. Broadly, the allocation of units can be done in two ways: units to emergencies and emergencies to units.

2.3.1 Units to Emergencies

In this method, emergencies are considered as constant and the units are allotted to each emergency. Generally, the number of available units for deployment will be greater than the number of emergencies, units equal to the number of emergencies would need to be selected from all the units available. Thus, each allocation object contains an array of units whose length is equal to the number of emergencies. In this, each unit corresponds to one emergency site.

2.3.2 Emergencies to Units

In this method, Units are taken to be constant, and the emergency sites are allocated to the units. Generally, the number of units is greater than the number of emergencies. Hence, this difference in number must be considered while creating the objects. Here, the proposed allocation object is considered as an array. The length of an array is equal to the number of units available. Each element in the array represents the allocation of the corresponding unit.

3 Mathematical Model

The mathematical models for both cases are mentioned in Section 2.3. They have the same inspiration and working principle, however, they differ in terms of the structure of the objects.

3.1 Fitness Function

In both the above-discussed cases, the fitness function is governed by some common factors. The factors are as follows:

- Severity of the Emergency
- Maximum severity that a particular unit can handle.
- Euclidean distance between the emergency service location and emergency site.

The fitness function has been derived as the summation of the score of each allocated emergency and unit pair. The mathematical formulation for the score of each pair is mentioned below:

$$score(e, u) = D_{e,u} \times \exp(S_e - S_u) \quad (1)$$

where, $D_{e,u}$ represents the Euclidean distance between emergency site e and Emergency Service Location of Unit u . S_e represents the severity of the Emergency e and S_u represents the maximum severity that unit u can handle.

The fitness is calculated as the sum of the scores of all emergencies and their allocated units.

$$fitness = \sum_{i=1}^n score(e_i, u_i) \quad (2)$$

where $e_1, e_2 \dots e_n$ are the n emergencies in the input situation and $u_1, u_2 \dots u_n$ are the n selected units out of all the units to be allocated to these n emergencies.

Here, the objective of this proposed model is to minimize the formulated objective function.

3.2 Units to Emergency

In this case, the allocation object consists of:

- The Input situation: This object represents the given situation in the form of a map which

contains the locations of the emergencies, service locations, severity of each emergency, and specifications regarding each unit.

- Allocated Units: It is a fraction of units from all available units that have been chosen for the particular allocation object.
- Solution Vector: It is a vector of the form $X_i = x_1, x_2, x_3 \dots x_n$, where n is the number of emergencies for which units need to be allocated. It represents the solution in its raw form.
- Allocation Map: This is the actual solution that can be obtained by applying the SPV rule on the solution vector. It contains all the emergency-unit pairs corresponding to the selected unit and their destinations.

3.2.1 Constraints

The constraints considered for the proposed model are summarized below:

1. The number of Units selected should be equal to the number of Emergencies.
2. Each unit can be assigned to only one emergency point. This constraint has been added for simulation purposes. In a real-life implementation, it can be replaced with a function of severity. Moreover, the kind of units available and the capabilities of each unit can be also considered.
3. The ratio of the number of units not allocated to the total number of units should be greater than the reserve ratio (This condition is flexible and must be changed as per the location, emergency situation as well as the history of emergency situations in the past).

3.3 Emergencies to Units

In this case, the Allocation object is defined a bit differently from what was done in the previous case.

- Input-Situation: This object represents the given situation in the form of a map which contains the locations of the emergencies, service locations, severity of each emergency, and specifications regarding each unit. This is the same as in the first case.
- Solution Vector: This is a vector of the form $X_i = x_1, x_2, x_3 \dots x_n$, where n is the total number of units. This represents the solution in its

raw form. The length of each vector is n , consisting of m emergencies and $n - m$ 0 values. Each element in this vector represents either the allocation or the absence of a unit. Here, 0 is considered for the absence of the unit and 1 is considered as an allocated unit. The units for which 0 has been assigned are not considered for fitness function calculation.

- **Allocation Map:** This is a map corresponding to the units and their allocated emergencies. This contains all the unit-emergency pairs corresponding to the selected emergency and the unit assigned to it. It represents the object that is evaluated when calculating the fitness.

3.3.1 Constraints

The constraints considered for the proposed model is summarized below:

1. The number of units selected should be equal to the number of emergencies.
2. The length of the solution vector is equal to the number of units available on hand.
3. The number of non-null or non-zero values in the solution vector must be equal to the number of emergencies.
4. The ratio of the number of units not allocated to the total number of units should be greater than reserve ratio (This condition is flexible and must be changed as per the location, emergency situation as well as the history of emergency situations in the past).

3.4 Confluence

The mathematical model of the research problem is formulated as follows:

$$\text{Min} \sum_{i=1}^n (D_{e_i, u_i} \times (S_{e_i} - S_{u_i})) \quad (3)$$

$$n < m \quad (4)$$

$$\frac{a}{m} > r \quad (5)$$

$$D_{e_i, u_i} > 0 \quad (6)$$

$$\sum_{i=0}^m 1[[X_i^{e2u} \in E]] = n \quad (7)$$

$$X_i^{u2e} \in U \quad (8)$$

Equation 4 ensures that the situation does not have more emergencies than the units available. Constraint 5 ensures that units are also kept in reserve to handle upcoming emergencies. Equation 7 is relevant for the Emergencies to Units case and represents that all emergencies must have units allotted to them. Equation 8 ensures that Units are appropriately allocated. In this paper, only one unit is assigned to one emergency, therefore $a = n$. The notations used in the above model are tabulated in Table 1.

3.5 Meta-heuristics Algorithms

In this paper, Particle Swarm Optimization (PSO) and Cuckoo Search meta-heuristics algorithms have been used to solve the defined problem.

3.5.1 Particle Swarm Optimization

Particle swarm optimization is a well-known swarm intelligence optimization algorithm used to address combinatorial optimization problems. It optimizes a problem by iteratively optimizing the candidate solutions using a given fitness function. PSO was originally proposed by Kennedy and Eberhart [35], which simulates the social behavior of bird flocks or fish schools. In PSO, each particle is considered as a potential candidate solution for the given problem, and the set of solutions is referred to as the swarm. The particle's position is represented by X , which is updated in each iteration for all particles. The best particle in the swarm is represented by $gbest$ and each particle's personal best is represented by $pbest$. The position of each particle is updated by computing the velocity v from $pbest$ and $gbest$. The mathematical formulation of the velocity and position update is mentioned below.

$$\begin{aligned} v_i(k+1) = & w \cdot v_i(k) + c_1 \cdot r_1 \cdot (pbest_i - x_i(k)) \\ & + c_2 \cdot r_2 \cdot (gbest - x_i(k)) \end{aligned} \quad (9)$$

$$x_i(k+1) = x_i(k) + v_i(k+1) \quad (10)$$

where $v_i(k)$ represents the velocity of i^{th} particle in k^{th} iteration and $X_i(k)$ represents the position of particle i in iteration k . $gbest$ represents the best particle in the current iteration and

Table 1 Notations

Parameters	
n	number of emergencies
m	number of units available
a	number of units allocated
r	reserve ratio, float between 0 and 1, can be a function of severity
e_i	i^{th} emergency
u_i	unit assigned to i^{th} emergency
D_{e_i, u_i}	Euclidean Distance between location of e_i and u_i
S_{e_i}	Severity of the i^{th} emergency
S_{u_i}	Severity of the unit assigned to deal with the i^{th} emergency
X^{u2e}	allocation vector in the Units to Emergencies case
X_i^{u2e}	unit assigned to i^{th} emergency
X^{e2u}	allocation vector in the Emergencies to Units case
X_i^{e2u}	emergency alloted to i^{th} unit or 0
E	set of all emergencies
U	set of all units

$pbest$ represents the best position of the i^{th} particle. c_1 and c_2 are constants that impact the social and cognitive behavior of the swarm. r_1 and r_2 are random numbers in the range of $[0, 1]$.

The PSO is a widely used and recognized algorithm that has been used to solve many real-life problems. Multiple variants have also been proposed trying to improve various aspects of this algorithm [36]. Hence, in this paper, the PSO algorithm has been used to solve the resource allocation problem in disaster management.

3.5.2 Cuckoo Search

Cuckoo Search(CS) is a well-known metaheuristic algorithm that was developed by Yang and Deb in 2009 inspired by the breeding behavior of Cuckoo Birds. Cuckoo Birds do not lay their eggs in their nests. They find other birds and lay their eggs in their nests. It leads to two possibilities. Either the host bird discovers the egg or it does not. If the host bird discovers the egg, it will either throw the egg or it will abandon its nest.

This behavior of the Cuckoo Birds inspired this algorithm.

The three ideal rules for CS which was proposed in the original paper are described as follows:

1. Each Cuckoo lays one egg at a time and deposits it in a randomly chosen nest.

2. The best nests with high-quality eggs will carry over to the next generations.
3. The number of available host nests is fixed and there is a probability that a host can discover an alien egg. In this case, the host bird can either throw the egg away or abandon the nest to build a completely new nest in a new location.

The Cuckoo Search algorithm is controlled using a combination of local random walk and a global random walk algorithm. The value of the random walk is controlled by the parameter p_a . p_a represents the probability that the egg is discovered by the host bird. Higher the value of p_a , higher is the exploitation in that specific iteration. Low values of p_a manes that the algorithm will favor exploration over exploitation. The mathematical formulation of the local random walk is

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \beta s \otimes H(p_a - \epsilon) \otimes (\mathbf{x}_j^t - \mathbf{x}_k^t) \quad (11)$$

where x_j^t and x_k^t are two different solutions selected randomly from all nests in the algorithm. $H(u)$ is the Heavyside Function and ϵ is a random number drawn between 0 and 1. $\otimes$ represents entry-wise multiplication operation and β is a scaling factor.

The global random walk is represented by

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \alpha \otimes L(\lambda) \quad (12)$$

Fig. 3 Situation 1: 15 emergencies, 2 eTypes

where α is a random number between 0 and 1, and $L(s, \lambda)$ represents the Levy Function.

$$L(\lambda) = \left| \frac{\Gamma(1 + \lambda) * \sin(\frac{\pi * \lambda}{2})}{\Gamma(\frac{1 + \lambda}{2}) * \lambda * 2^{\frac{\lambda - 1}{2}}} \right|^{\frac{1}{\lambda}} \quad (13)$$

4 Experimental Results

In this section, the performance of the proposed model is tested with Particle Swarm Optimization and Cuckoo Search. Since this is a novel model for this situation, there is no benchmark to evaluate its performance.

Therefore, we have used Particle Swarm Optimization and Cuckoo Search, two popular meta-heuristics, to assess the proficiency of this model. All algorithms were implemented using Python 3.7.9 and performed on a 3.1 GHz Dual-Core Intel Core i5 with 8GB of memory running Mac OS 11.2.

4.1 Data Generation

For experimental analysis, ten different scenarios have been generated with varying numbers and locations of emergencies, service locations, and units. The parameters of these scenarios are shown in Table 2. For rows 1 – 4, the region in which this situation takes place is $x, y \in [-100, 100]$ and for rows 5 – 10, $x, y \in [-1000, 1000]$.

4.2 Parameters

Table 3 shows the relevant parameters for Cuckoo Search and Particle Swarm Optimization. For both algorithms, the number of iterations has been chosen as the termination criterion and the swarm population has been set constant for both.

Table 2 Situations Specifications

Situation	Emergencies	Service Locations	Emergency Types
1	15	11	2
2	100	50	2
3	100	50	5
4	100	50	4
5	200	75	4
6	200	75	3
7	200	100	3
8	500	150	3
9	500	150	4
10	500	150	5

Table 3 Parameters

Algorithm	Parameter	Value
PSO	c1	2
PSO	c2	2
PSO	Particles	20
CS	p_a	0.25
CS	λ	1.5
CS	Nests	20
PSO, CS	Iterations	100

4.3 Analysis of Results and Discussion

Tables 4 and 5 show the results of Particle Swarm Optimization and Cuckoo Search on the 10 different considered scenarios (Refer Section 4.1). Since there is no benchmark fitness to work with, due to the novelty of the proposed model, we have used two popular meta-heuristics to solve this problem. The fitness that is calculated in each situation is independent of the fitness of other situations, they cannot be compared with each other. Thus, the results of a particular optimization algorithm on a given situation can only be compared with the results of other algorithms on the same situation. This does not impact the effectiveness of the proposed model since, in real-life applications, a situation is fixed and unchangeable, and it is this situation for which an efficient allocation is needed to be found.

In each iteration, the optimization algorithms try to improve their fitness score. Figure 4 shows how the score is improved on Situation 1 from table 2.

It is observed that Cuckoo Search outperforms Particle Swarm Optimization in most cases. In Fig. 4 a line connecting a service location and an emergency represents the allocation of a unit from that specific service location to that emergency. This follows the same legend as in fig 3. It can be observed in Fig 4 that in some cases a unit from a service location farther from the closest service location is allocated. This is done because the score of a farther service location might be higher if the maximum severity that can be handled by the units available is insufficient to handle a given emergency. This is also the reason the fitness function contains the expression $\exp(S_e - S_u)$. As a consequence, the preference is given to a unit that can handle the emergency compared to a unit that has a lower maximum severity than the severity of the given emergency. Each of the two methodologies discussed in this paper has its advantages and disadvantages and would perform better in some cases.

The Units to Emergencies methodology is characterized by having a small memory requirement.

This is because the number of units is greater than the number of emergencies and the array of objects in the model is equal to the number of emergencies. However, this methodology lacks in its exploration ability when compared to the other. Since in each allocation object, the units selected are fixed when the object is initialized, during any meta-heuristic algorithm, the solution represented by that particular object will be restricted to having the selected units only. This can be offset by increasing the population parameter of the meta-heuristic used.

Iteration: 1
Fitness = 1743.4439302768858

Iteration: 15
Fitness = 1604.2419121773996

Iteration: 30
Fitness = 1097.492704665907

Iteration: 60
Fitness = 908.8669880327836

Iteration: 80
Fitness = 894.2451503840399

Iteration: 100
Fitness = 656.6486337272244

Fig. 4 Results on Situation 1

Table 4 Results with PSO

Situation	PSO							
	Units to Emergencies				Emergencies to Units			
	Best Fitness	Mean	Worst	Std	Best Fitness	Mean	Worst	Std
1	5.58E+02	7.44E+02	8.92E+02	1.12E+02	5.96E+02	7.19E+02	8.26E+02	7.01E+01
2	1.63E+05	1.71E+05	1.80E+05	5.50E+03	1.62E+05	1.70E+05	1.80E+05	5.23E+03
3	1.40E+05	1.54E+05	1.64E+05	6.86E+03	1.16E+05	1.37E+05	1.47E+05	1.14E+04
4	1.64E+05	1.76E+05	1.93E+05	9.33E+03	4.35E+05	4.52E+05	4.72E+05	1.39E+04
5	3.59E+05	3.81E+05	4.00E+05	1.05E+04	3.41E+05	3.58E+05	3.77E+05	1.24E+04
6	3.15E+05	3.37E+05	3.51E+05	3.37E+05	3.87E+05	4.03E+05	4.21E+05	1.19E+04
7	3.83E+05	4.06E+05	4.24E+05	1.23E+04	4.50E+05	4.67E+05	4.67E+05	9.68E+03
8	1.20E+06	1.23E+06	1.26E+06	1.94E+04	1.17E+06	1.19E+06	1.22E+06	1.49E+04
9	1.18E+06	1.20E+06	1.22E+06	1.36E+04	1.09E+06	1.15E+06	1.18E+06	2.76E+04
10	1.14E+06	1.19E+06	1.23E+06	2.67E+04	1.25E+06	1.21E+06	1.37E+05	1.99E+04

Table 5 Results with CS

Situation	CS							
	Units to Emergencies				Emergencies to Units			
	Best Fitness	Mean	Worst	Std	Best Fitness	Mean	Worst	Std
1	6.04E+02	6.49E+02	7.42E+02	4.96E+01	5.62E+02	7.16E+02	8.65E+02	1.12E+02
2	1.27E+05	1.44E+05	1.64E+05	9.82E+03	1.30E+05	1.40E+05	1.48E+05	5.55E+03
3	1.49E+05	1.61E+05	1.67E+05	6.00E+03	1.53E+05	1.63E+05	1.69E+05	5.52E+03
4	1.21E+05	1.39E+05	1.52E+05	8.58E+03	1.31E+05	1.41E+05	1.48E+05	5.53E+03
5	4.16E+05	4.35E+05	4.54E+05	1.46E+04	4.03E+05	4.24E+05	4.43E+05	1.27E+04
6	3.93E+05	4.15E+05	4.41E+05	1.62E+04	4.07E+05	4.25E+05	4.42E+05	1.17E+04
7	3.52E+05	3.79E+05	3.92E+05	1.14E+04	3.46E+05	3.72E+05	3.87E+05	1.37E+04
8	1.11E+06	1.13E+06	1.16E+06	1.86E+04	1.09E+06	1.13E+06	1.17E+06	2.19E+04
9	1.09E+06	1.11E+06	1.13E+06	1.29E+04	1.17E+06	1.23E+06	1.27E+06	2.82E+04
10	1.17E+06	1.24E+06	1.26E+06	2.76E+04	1.31E+06	1.34E+06	1.36E+06	1.74E+04

The Emergency to Units methodology requires more memory, however, it offers a better exploration behavior than the previous methodology. Since in this case each solution object is represented by an array that contains relations to all emergency and unit objects. It offers an exploration of the entire samples space by each object in the agent population. Since this methodology requires having all emergency and unit objects related, it requires more memory. This becomes significant if we have a large number of units and emergencies or if we have various types of units and emergencies available on hand. Tables 4 and 5 show the comparison between both these methods. In the conducted experiments, both these methodologies offer similar results in all 10 situations that were solved.

5 Conclusion

In this paper, a novel model has been proposed for the Emergency Disaster Resource Allocation as a combinatorial optimization problem. The proposed model can be implemented in two ways depending upon the computing resources available at the site. The choice of one of these ways can increase the exploration ability of the algorithm used at the expense of taking more computing resources, whereas, in the other way, the exploration ability of the used algorithm is not enhanced. The model takes the distance of the emergency, the severity of the emergency, the maximum severity that can be handled by a particular unit, and the types of emergency and units as factors affecting the fitness of a particular allocation.

Particle Swarm Optimization and Cuckoo Search have been used to test this model and the results show Cuckoo Search outperforms Particle Swarm Optimization in the time taken and the quality of results. The results show that Cuckoo Search outperforms Particle Swarm Optimization in simulated situations, but since this is a novel model, there are no benchmarks that can be used to evaluate this.

Future work in this direction includes the elaboration of the factors considered. The proposed model can also be modified and tested using real-world data to set up benchmarks for future results. Optimization algorithms can also be modified

to solve this combinatorial optimization problem with more success.

Declarations

The authors did not receive support from any organization for the submitted work. The authors have no competing interests to declare that are relevant to the content of this article.

Data Availability

All data generated or analyzed during this study are included in this published article. Any supplementary data and information related to this research are available on request.

Author Contribution

All authors contributed to the idea conception, implementation, and analysis. The first draft of the manuscript was written by Sehej Jain and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

References

- [1] Altay, N., Green III, W.G.: Or/ms research in disaster operations management. *European journal of operational research* **175**(1), 475–493 (2006)
- [2] Ajami, S., Fattahi, M.: The role of earthquake information management systems (eimss) in reducing destruction: A comparative study of japan, turkey and iran. *Disaster Prevention and Management: An International Journal* (2009)
- [3] Ransikarbum, K., Mason, S.J.: Goal programming-based post-disaster decision making for integrated relief distribution and early-stage network restoration. *International Journal of Production Economics* **182**, 324–341 (2016)
- [4] Gasparini, P., Manfredi, G., Zschau, J., *et al.*: *Earthquake Early Warning Systems*. Springer, ??? (2007)

- [5] Sutton, J., Tierney, K.: Disaster preparedness: Concepts, guidance, and research. Colorado: University of Colorado **3**, 1–41 (2006)
- [6] Erbeyoğlu, G., Bilge, Ü.: A robust disaster preparedness model for effective and fair disaster response. *European Journal of Operational Research* **280**(2), 479–494 (2020)
- [7] Das, R.: Disaster preparedness for better response: Logistics perspectives. *International journal of disaster risk reduction* **31**, 153–159 (2018)
- [8] Acar, M., Kaya, O.: A healthcare network design model with mobile hospitals for disaster preparedness: A case study for istanbul earthquake. *Transportation Research Part E: Logistics and Transportation Review* **130**, 273–292 (2019)
- [9] Abdel-Basset, M., Mohamed, R., Elhoseny, M., Chang, V.: Evaluation framework for smart disaster response systems in uncertainty environment. *Mechanical Systems and Signal Processing* **145**, 106941 (2020)
- [10] Pi, Y., Nath, N.D., Behzadan, A.H.: Convolutional neural networks for object detection in aerial imagery for disaster response and recovery. *Advanced Engineering Informatics* **43**, 101009 (2020)
- [11] Chowdhury, S., Emelogu, A., Marufuzzaman, M., Nurre, S.G., Bian, L.: Drones for disaster response and relief operations: A continuous approximation model. *International Journal of Production Economics* **188**, 167–184 (2017)
- [12] Nayeri, S., Asadi-Gangraj, E., Emami, S.: Metaheuristic algorithms to allocate and schedule of the rescue units in the natural disaster with fatigue effect. *Neural Computing and Applications* **31**(11), 7517–7537 (2019)
- [13] Eid, M.S., El-Adaway, I.H.: Decision-making framework for holistic sustainable disaster recovery: Agent-based approach for decreasing vulnerabilities of the associated communities. *Journal of infrastructure systems* **24**(3), 04018009 (2018)
- [14] Miles, S.B., Burton, H.V., Kang, H.: Community of practice for modeling disaster recovery. *Natural Hazards Review* **20**(1), 04018023 (2019)
- [15] Kanno, T., Koike, S., Suzuki, T., Furuta, K.: Human-centered modeling framework of multiple interdependency in urban systems for simulation of post-disaster recovery processes. *Cognition, Technology & Work* **21**(2), 301–316 (2019)
- [16] Saleem, K., Luis, S., Deng, Y., Chen, S.-C., Hristidis, V., Li, T.: Towards a business continuity information network for rapid disaster recovery. In: *Dg. O*, pp. 107–116 (2008). Citeseer
- [17] Kondaveti, R., Ganz, A.: Decision support system for resource allocation in disaster management. In: *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3425–3428 (2009). IEEE
- [18] Fikar, C., Gronalt, M., Hirsch, P.: A decision support system for coordinated disaster relief distribution. *Expert Systems with Applications* **57**, 104–116 (2016)
- [19] Hadiguna, R.A., Kamil, I., Delati, A., Reed, R.: Implementing a web-based decision support system for disaster logistics: A case study of an evacuation location assessment for indonesia. *International journal of disaster risk reduction* **9**, 38–47 (2014)
- [20] Jung, D., Tran Tuan, V., Dai Tran, Q., Park, M., Park, S., *et al.*: Conceptual framework of an intelligent decision support system for smart city disaster management. *Applied Sciences* **10**(2), 666 (2020)
- [21] Turğut, B.T., Taş, G., Herekoğlu, A., Tozan, H., Vayvay, O.: A fuzzy ahp based decision support system for disaster center location selection and a case study for istanbul. *Disaster Prevention and Management: An International Journal* (2011)
- [22] Rakes, T.R., Deane, J.K., Rees, L.P., Fetter,

- G.M.: A decision support system for post-disaster interim housing. *Decision Support Systems* **66**, 160–169 (2014)
- [23] Mete, H.O., Zabinsky, Z.B.: Stochastic optimization of medical supply location and distribution in disaster management. *International Journal of Production Economics* **126**(1), 76–84 (2010)
- [24] Fiedrich, F., Gehbauer, F., Rickers, U.: Optimized resource allocation for emergency response after earthquake disasters. *Safety science* **35**(1-3), 41–57 (2000)
- [25] Leifler, O.: Combining technical and human-centered strategies for decision support in command and control-the complan approach. In: 5th International ISCRAM Conference, May 4-7, Washington, DC, USA, pp. 504–515 (2008)
- [26] Van de Walle, B., Turoff, M.: Decision support for emergency situations. In: *Handbook on Decision Support Systems 2*, pp. 39–63. Springer, ??? (2008)
- [27] Indumathy, R., Maheswari, S.U., Subashini, G.: Nature-inspired novel cuckoo search algorithm for genome sequence assembly. *Sadhana* **40**(1), 1–14 (2015)
- [28] Luo, K., Zhao, Q.: A binary grey wolf optimizer for the multidimensional knapsack problem. *Applied Soft Computing* **83**, 105645 (2019)
- [29] Ke, L., Feng, Z., Ren, Z., Wei, X.: An ant colony optimization approach for the multidimensional knapsack problem. *Journal of Heuristics* **16**(1), 65–83 (2010)
- [30] Ouhaarab, A., Ahiod, B., Yang, X.-S.: Discrete cuckoo search algorithm for the travelling salesman problem. *Neural Computing and Applications* **24**(7), 1659–1669 (2014)
- [31] Hu, D., Dai, P., Zhou, K., Ge, S.: Improved particle swarm optimization for minimum spanning tree of length constraint problem. In: 2015 8th International Conference on Intelligent Computation Technology and Automation (ICICTA), pp. 474–477 (2015). IEEE
- [32] Rolland, E., Patterson, R.A., Ward, K., Dodin, B.: Decision support for disaster management. *Operations Management Research* **3**(1-2), 68–79 (2010)
- [33] Yi, W., Kumar, A.: Ant colony optimization for disaster relief operations. *Transportation Research Part E: Logistics and Transportation Review* **43**(6), 660–672 (2007)
- [34] Dávila de León, A., Lalla-Ruiz, E., Melián-Batista, B., Moreno-Vega, J.M.: A simulated annealing-based approach for aid distribution in post-disaster scenarios. In: *International Conference on Computer Aided Systems Theory*, pp. 335–343 (2019). Springer
- [35] Eberhart, R., Kennedy, J.: Particle swarm optimization. In: *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948 (1995). Citeseer
- [36] Zhang, Y.-D., Wang, S., Ji, G.: A comprehensive survey on particle swarm optimization algorithm and its applications. *Mathematical Problems in Engineering* **2015**, 1–38 (2015). <https://doi.org/10.1155/2015/931256>