Int j simul model 24 (2025) 2, 273-284
Original scientific paper

BARE-BONES PARTICLE SWARM OPTIMIZATION FOR EMERGENCY SCHEDULING IN PUBLIC EVENTS

Zhang, H. K.*; Zheng, Q. M.*; Yang, S.* & Ojstersek, R.**,#
* Business School, Beijing Technology and Business University, Beijing 100048, P. R. China
** University of Maribor, Faculty of Mechanical Engineering, Smetanova 17, 2000 Maribor, Slovenia
E-Mail: robert.ojstersek@um.si (# Corresponding author)

Abstract
Efficient scheduling of emergency resources is of great practical significance to ensure the smooth progress of large-scale events and maintain public safety. Therefore, this paper, firstly, proposes a novel emergency rescue system after public emergencies in large-scale sports events. In order to allocate resources, this paper establishes a single-objective model that minimizes the maximum delivery time (the injured waiting in place and being transported to the hospital) on the basis of considering the rescue effect. This paper proposes an improved bare-bones particle swarm optimization algorithm (IBBPSO) to solve the model. Through case analysis, it is found that compared with the bare-bones particle swarm optimization algorithm (BBPSO) and RBBPSO, the average optimization effect of IBBPSO is 62 % higher than that of RBBPSO and 23 % higher than that of BBPSO. IBBPSO has better performance in solving the resource allocation problem proposed in this paper.
(Received in January 2025, accepted in March 2025. This paper was with the authors 2 weeks for 2 revisions.)

Key Words: Emergency Rescue System, Rescue Effect, Bare-Bones Particle Swarm Optimization, Emergency Medical Resource Scheduling

1. INTRODUCTION

Medical resource scheduling is an important strategic decision, which plays a vital role in emergency medical response.

Large-scale casualty events [1] often lead to a surge in demand for emergency medical services. It is therefore crucial that emergency medical resources are allocated according to priority [2]. Most studies distinguish priorities by assigning different levels of urgency or using different service weights [3]. Another way to describe priority such as Jin et al. [4] is to apply different survival probabilities to reveal different injury severity. Model construction of hospital allocation and vehicle scheduling. Repoussis et al. [5] proposed a mixed integer programming (MIP) formula for hospital selection and treatment ranking. They assume that the patient transport sequence follows the START classification method to minimize the time required to complete the entire patient transport and treatment. Dean and Nair [6] concerned with how to effectively evacuate victims to the different area hospitals in order to provide the greatest good to the greatest number of patients while not overwhelming any single hospital. Wilson et al. [7] modelled the allocation of the injured to the hospital as a single-cycle multi-objective flexible job shop scheduling problem. Each patient is considered as a job, and each response unit is considered as a machine.

In order to deal with the increasingly complex optimization problems effectively, a large number of population-based metaheuristic optimization algorithms have gained rapid development and widespread attention, such as genetic algorithm (GA) [8-10], particle swarm optimization (PSO) [11, 12], ant colony optimization (ACO) [13], hybrid approaches [14-16] etc. Tlili et al. [17] consider the emergency situations in which many of the injured need urgent medical care at the same time, they proposed a cluster-first routing second algorithm based on Petal algorithm and PSO method. Dağlayan and Karakaya [18] proposed a new GA to effectively schedule ambulances after a disaster.

To better understand and analyse population-based metaheuristic algorithms, Kennedy proposed a bare bones mechanism for simplifying these algorithms and applied this mechanism to the original PSO to obtain a modified PSO called bare bones PSO (BBPSO) [19]. In the BBPSO, the velocity vector of an individual was eliminated, and the solution updating equation was revised as a random vector that obeys a normal distribution. Zhang et al. [20] proposed a new BBPSO algorithm that designs an approach based on particle diversity for updating the global particle leaders. The dynamic update rule of particle swarm optimization is formulated as a second-order stochastic difference equation and general relations are derived for search focus, search spread, and swarm stability at stagnation by Blackwell [21]. Guo and Sato [22] proposed the standardized bare bones particle swarm optimization (SBBPSO) algorithm. A standardized converter is used to ensure the dimensions of each particle are integers. Li et al. [23] proposed a new method of feature selection based on an adaptive BBPSO. First, they use the logistic equation of chaotic systems to initialize the particle swarm. Then, the adjacent algorithm (KNN) is used as a classifier to evaluate the achievement of the standard data set. Omran and Sharhan [24] developed a clustering method that is based on barebones particle swarm. The proposed algorithm finds the centroids of a user specified number of clusters, where each cluster groups together similar patterns.

Through the combing of the above literature, we can see that the previous theoretical models are more concerned with the minimum total cost or the shortest total rescue time in the rescue process. It is critical to consider the balance of each person's rescue time and to use a suitable algorithm to solve this practical problem which is NP-hard. Therefore, how to make the rescue time of the injured more balanced considering the different circumstances of the injured and find a stable scheduling scheme in a short time is the most important issue after the public emergency of large-scale sports events.

Therefore, this paper first proposes a novel rescue system and then constructs a mathematical model that balances rescue efficiency and effectiveness based on injury classification. This paper also proposes IBBPSO by introducing the BBPSO into the chaotic mapping randomization population strategy, the particle position update strategy, the particle mutation strategy and the constraint processing function. The improved BBPSO will have stronger adaptability and higher optimization performance in solving NP-hard problems such as emergency medical resource scheduling.

The remainder of this study is organized as follows. Section 2 describes the ideas and methods to solve the problem in this paper. Section 3 introduces the resource scheduling model established. Section 4 shows the specific improvement strategies of IBBPSO and coding examples. Section 5 shows a numerical experiment. Section 6 is the result of the experiment in section 5. Section 7 summarizes the conclusions drawn from this paper.

2. METHODOLOGY

In case of public emergencies in large sports events, the temporary rescue team near the gymnasium will rush to the front line immediately for rescue and reporting the injured situation and location of gymnasium to the emergency rescue dispatch command centre (ERDCC). At the same time, the ambulance centre and hospitals are responsible for receiving and uploading the requests information (location and capacity of available hospitals, location and number of available ambulances, time the ambulance was first available et al.) from ERDCC. Then, ERDCC accesses all the feedback information into the constructed model and gives the best allocation scheme through IBBPSO. See the following Fig. 1 for more details.

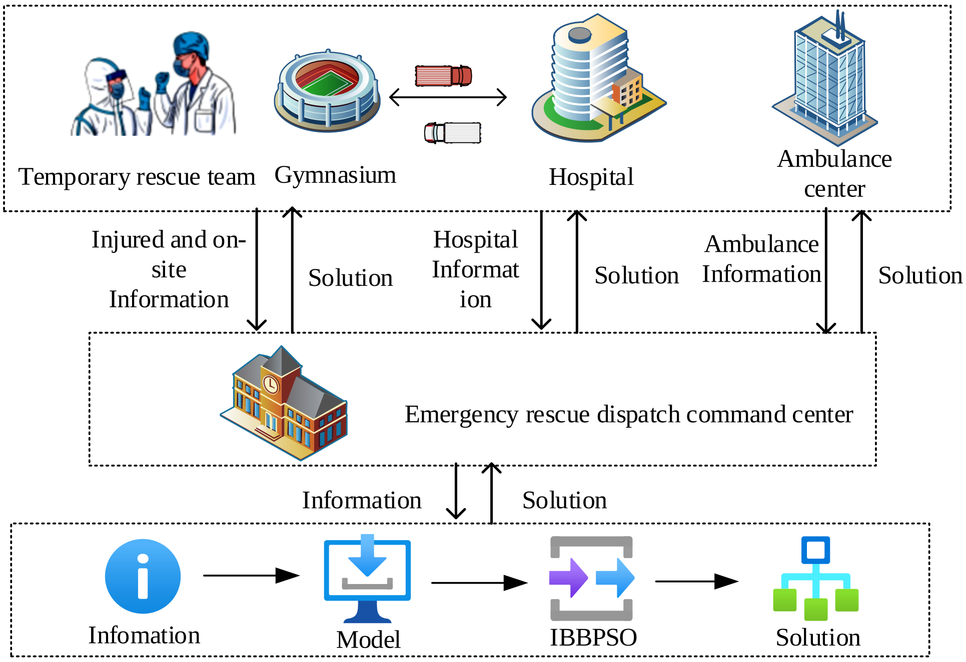

Figure 1: Event object diagram.

3. MATHEMATICAL MODEL

In order to effectively dispatch ambulances and hospitals after public events, this paper established a model that can balance rescue efficiency and rescue effect on the basis of considering the classification and treatment of the injured. The model aims to minimize the maximum transit time of the injured.

3.1 Problem formulation

This paper aims to address the problem of efficiently managing large-scale injury incidents during major sports events, where injured individuals await ambulance transport to hospitals. The cooperative scheduling problem of rescue hospital allocation and transportation network is shown in the Fig. 2 as below.

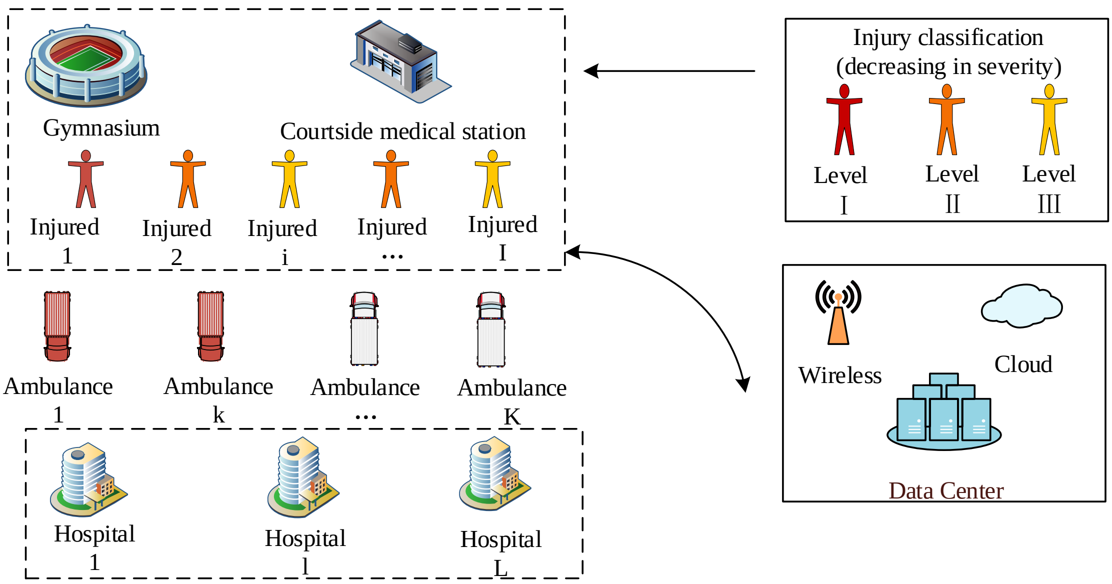

Figure 2: The cooperative scheduling problem.

3.2 Model assumptions

(1) The location of the public incident, available hospitals and available ambulances are known.
(2) The number of injured people, available hospitals and available ambulances are known.
(3) The model does not take into account minor injuries and deaths that do not require medical attention.
(4) Each ambulance can only transport one injured at a time and each injured is transported once during the rescue operation.
(5) The number of people injured in this accident is far greater than the number of available ambulances than the number of hospitals.
(6) The injured people are divided into three categories according to their degree of injury. The first-level injury was the most serious and priority scheduling.
(7) The capacity of each hospital to admit injured people is measured by the number of the injured it can accommodate.
(8) The loading and unloading time of each injured in each ambulance are not considered.
(9) The parameters of the ambulance in this model are all the same.
(10) The time injured has been waiting at dispatch time is known.
(11) How long the ambulance is available at dispatch and where it is available after that is known.

3.3 Model formulation

There are L hospitals, K ambulances, and I injured in the rescue network. $A = \{a_1, a_2, \ldots, a_k, \ldots, a_K\}$ represent the collection of ambulances. $B = \{b_1, b_2, \ldots, b_i, \ldots, b_I\}$ represents the collection of injured. $C = \{c_1, c_2, \ldots, c_l, \ldots, c_L\}$ represents the set of hospitals. $R = \{R_1, R_2, \ldots, R_r, \ldots, R_R\}$ represents the trip of ambulances. S represents the location of the event. w_i represents the time the injured i had waited before the completion of the scheduling. U_k means how long ambulance k will be available at dispatch time before it begins its first journey. Each hospital C_l has a maximum receiving capacity E_l. A travel distance $d(m, n)$ is generated when an ambulance crosses an arc (m, n). We assume that all ambulances are traveling at average speed v. l_{kr} means the destination hospital of the ambulance k on its r trip. It is worth noting that we set l_{k0} as the position where the ambulance k is first available. We define several binary metrics X_{ikr} and Y_{kr}. If the injured i is transported by ambulance k on the r trip, $X_{ikr} = 1$, otherwise, $X_{ikr} = 0$. If ambulance k makes the r trip, $Y_{kr} = 1$, otherwise, $Y_{kr} = 0$. When the destination of the ambulance k is hospital l, $Z_{kl} = 1$, otherwise $Z_{kl} = 0$.

$$f = min\{Max(T_i^{K_r})\} \tag{1}$$

$$T_i^{K_r} = w_i + G_{k_{r-1}} + d[l_{k_{r-1}}, s]/v + d[s, l_{k_r}]/v \tag{2}$$

$$G_{k_{r-1}} = \begin{cases} U_k, r = 1 \\ G_{k_{r-2}} + d[l_{k_{r-2}}, s]/v + d[s, l_{k_{r-1}}]/v, r \geq 2 \end{cases} \tag{3}$$

$$d[l_{k_{r-1}}, s]/v = \begin{cases} d[l_{k_0}, s]/v, r = 1 \\ d[l_{k_{r-1}}, s]/v, r \geq 2 \end{cases} \tag{4}$$

$$\sum_{k=1}^{K}\sum_{r=1}^{R} X_{ikr} = 1, \forall i \in B \tag{5}$$

$$\sum_{k=1}^{K} Y_{kr} \leq K, \forall k \in A \tag{6}$$

$$Y_{kr} - Y_{k(r-1)} \leq 0, \forall k \in A, \forall r \in R \tag{7}$$

$$W_i \geq 0 \tag{8}$$

$$\sum_{i=1}^{I} X_{ikr} \leq Y_{kr}, \forall k \in A, \forall r \in R \tag{9}$$

$$l_{k_r} \in C, \forall k \in A, \forall r \in R \tag{10}$$

$$\sum_{k=1}^{K}\sum_{r=1}^{R}\sum_{i=1}^{I} Z_{kl} X_{ikr} \leq E_l \tag{11}$$

$$X_{ikr} \in \{0, 1\} \tag{12}$$

$$Y_{kr} \in \{0, 1\} \tag{13}$$

$$Z_{kl} \in \{0, 1\} \tag{14}$$

The objective function, Eq. (1), is to minimize the maximum waiting and transport time for an injured. Eq. (2) represents the time taken by an injured person from his injury to the hospital. In Eq. (3), G_{kr-1} represent the time taken from the completion of the algorithm scheduling to the completion of the medical treatment of the injured (transported by the ambulance k on its $r-1$ trip). Eq. (4) describes the time it takes for an ambulance k to travel from the hospital of destination of its $r-1$ trip to the place of discovery. Eq. (5) guarantees that each injured can only be transported once in one ambulance. Eq. (6) ensures that the number of ambulances used must not exceed the total number of ambulances. Constraint in Eq. (7) determines each ambulance must travel continuously. Eq. (8) represents arrangements are only made for the injured whose condition have been confirmed. Eq. (9) guarantee ambulances can transport only one injured per trip. Eq. (10) states that the final journey of each ambulance must reach the hospital. Eq. (11) is explained that because the hospital 's treatment capacity and resources are limited, in order to ensure the quality of medical services and treatment effects, the number of injured acceptable to each hospital is limited. The number of injured allocated to each hospital is not allowed to exceed the maximum number of injured it can bear. Finally, Eqs. (12) to (14) define the domain of the decision variables.

4. IMPROVED BARE-BONES PARTICLE SWARM OPTIMIZATION

BBPSO proposed by Kennedy [19] is a particle swarm optimization algorithm which form is more concise than the PSO, and the theoretical structure is simpler, which can improve the search efficiency and accuracy of the algorithm. It has been successfully applied to constrained optimization problems, data mining, power system regulation and other fields.

In order to overcome the limitations that may be encountered in solving complex problems, such as easy to fall into local optimum, slow convergence speed and low search accuracy, this paper optimizes BBPSO to solve the optimization problems more effectively in practical applications. The general pseudo-code of the IBBPSO is shown in **Algorithm 1**.

4.1 Initialize the population

In the initialization phase, the swarm with size Ns is randomly generated. Each particle in the swarm is assigned random values for each dimension from the respective domain. We use chaotic map to randomly generate the initial position of the particles. Chaotic map is a nonlinear dynamic system. Its pseudo-random property can increase the adaptability of the algorithm to different problems. The initial value for the personal best position of each particle is set to be the particle itself.

Algorithm 1: The general pseudo-code of the IBBPSO.

Step 0: Setting the parameters. particle swarm size, particle dimension and number of iterations etc.
Step 1: Initialize the population.
  Step 1.1: Initialize swarm positions. Using Chaotic Map to randomly generate the initial position of the particles.
  Step 1.2: Evaluate the fitness of each individual.
  Step 1.3: Set personal best positions. The personal best position of each particle is set to be the particle itself.
  Step 1.4: Set global best position. Find the global best position from the personal best position.
Step 2: Iterative population.
for $ite = 1: max\_ite$
  for i = 1: *Ns*
    Step 2.1: Update particle position. Update the position of each particle by the Eq. (15).
    Step 2.2: Apply mutation operator. The mutation strategy based on chaotic map dynamically selects agents and dimensions for mutation in each iteration and gradually increases the mutation range with iteration to balance global and local search.
    Step 2.3: Handle constraints. The algorithm deals with two main types of constraints: boundary constraints and capacity constraints. If the candidate solution exceeds the given upper and lower bounds, the candidate solution will be replaced according to Eq. (16).
    Step 2.4: Evaluate the fitness of each individual.
    Step 2.5: Update personal best positions.
    Step 2.6: Update global best positions.
  End
End

4.2 Update particle position

Here we propose new formula to update the particle 's position:

$$x_{i,j}(t+1) = \begin{cases} N\left(r_1 Pb_{i,j(t)} + (1-r_1)Gb_{j(t)}, \left|Pb_{i,j(t)} - Gb_{j(t)}\right|\right), & r_2 < 0.5 \\ Pb_{i,j(t)}, & 0.5 \le r_2 < 0.75 \\ Gb_{j(t)}, & \text{otherwise} \end{cases} \tag{15}$$

The new position of the particle is determined by the historical personal best position and the global best position. Both r_1 and r_2 are random numbers generated by chaotic maps. If r_2 is less than 0.5, the new position of the particle is generated by the normal distribution. If r_2 is greater than or equal to 0.5 and less than 0.75, the new position of the particle remains its historical personal best position $Pb_{i,j(t)}$ unchanged. Otherwise, the new position of the particle takes the global best position $Gb_{j(t)}$. This mechanism helps the particle, broadens the search range, and explores new potential solutions within the search space.

4.3 Handle constraints

In this paper, there are two kinds of constraints: boundary constraints and capacity constraints. The boundary constraint ensures that the solution is in the feasible solution space and the capacity constraint ensures that the maximum carrying capacity of the system will not be exceeded when allocating resources or tasks.

Border constraints: The algorithm ensures that the candidate solutions remain within the specified boundaries and are adjusted when they exceed these boundaries. At the same time, a certain randomness is introduced, and the global optimal solution and individual optimal solution are used to guide the adjustment process. If the candidate solution exceeds the given upper and lower bounds, the candidate solution is replaced according to Eq. (16):

$$x_{i,j} = \begin{cases} rand\left(L_j, U_j\right), & r_3 < 0.25 \\ Pb_{i,j}, & 0.25 \le r_3 < 0.5 \\ Gb_{i,j,} & 0.5 \le r_3 < 0.75 \\ L_j, & 0.75 \le r_3 \le 1 \cap X_j < L_j \\ U_j, & 0.75 \le r_3 \le 1 \cap X_j < L_j \cap X_j > U_j \end{cases} \tag{16}$$

For $x_{i,j}$, if it exceeds the corresponding lower bound L_j and upper bound U_j, 0.25 probability using the randomly generated solution, 0.25 probability using the personal best solution, 0.25 probability using the global best solution, and 0.25 probability using the boundary value. Specifically, r_3 is a random number generated from uniformly distributed $U(0,1)$.

Hospital capacity constraints: We decode the individuals obtained by IBBPSO to obtain a rescue solution. Once the number of injured assigned to a hospital exceeds the upper limit of the hospital 's capacity, the solution is invalid.

4.4 Apply mutation operator

This algorithm implements a dynamic mutation strategy through chaotic mapping. r_4 is a random number within [0, 1]. This algorithm defines a factor e ($e < r_4$) that increases with the increase of the number of iterations to control the degree of variation, uses chaotic values to select individuals and dimensions for variation, calculates the variation range according to the problem boundary and e, and uses Gaussian distribution to randomly generate individual positions to balance global and local search.

4.5 Examples of encoding

In order to apply IBBPSO to solve the problem, it is necessary to design a coding system to realize the transformation of optimization scheme. We take a large gymnasium with 6 injured people with different degrees of injury, 3 ambulances nearby and 2 hospitals with different capacities as an example. The coding scheme is as Fig. 3.

Subtable1: Injured condition form

Number	Name	Sex	Injured classification	Wating time (h)	Location
1	Ming Li	Male	II	0.07	(116.3959, 39.9934)
2	Hao Wang	Female	I	0.12	(116.3959, 39.9934)
3	Dong Zhang	Male	II	0.08	(116.3959, 39.9934)
4	Guo Li	Male	I	0.16	(116.3959, 39.9934)
5	Kang Yang	Female	III	0.05	(116.3959, 39.9934)
6	Xue Lin	Female	I	0.18	(116.3959, 39.9934)

Subtable2: Hospital condition form

Number	Name	Location	Capacity (person)
1	Haidian Hospital	(116.3155,39.9766)	9
2	PLA General Hospital	(116.2765,39.9045)	15

Subtable3: Ambulance condition form

Number	Velocity (km/h)	Available time	Available position
1	72	0	(116.2026,40.0264)
2	72	0.13	(116.0522,39.9068)
3	72	0.5	(116.1890,39.9612)

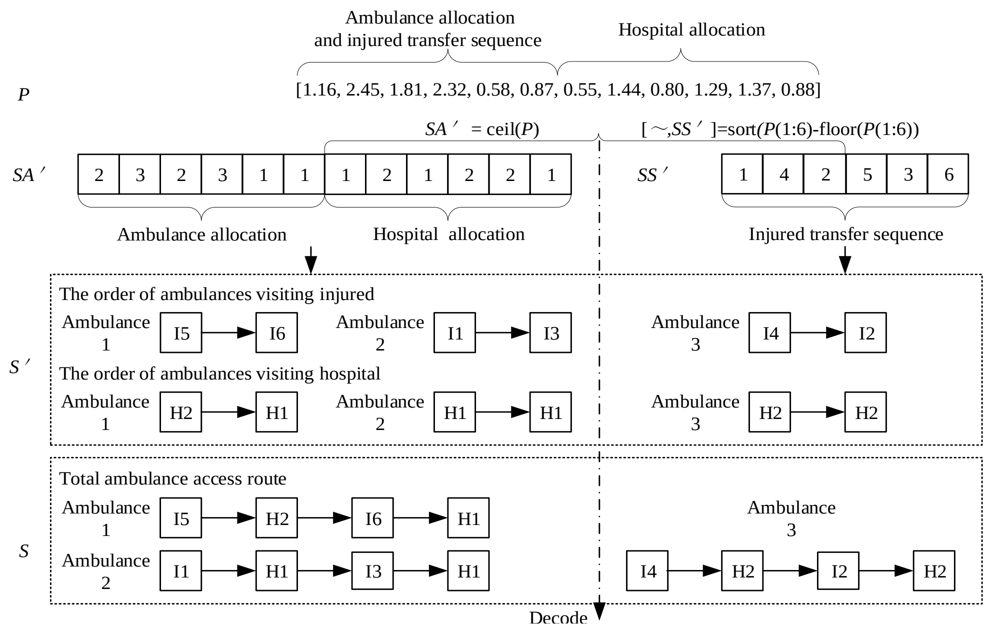

Figure 3: An example of the encoding solution.

We introduce a coding solution based on relative position index to solve the combinatorial optimization problem. In the proposed coding scheme, each particle has $2I$ dimensions, the range of the first 1 to I dimensions of the particle is (0, K), and the range of the $(I+1)$ to $2I$ dimensions is (0, J). The simulation of the coding scheme can be described by the following steps: When initializing the position of the particle, the dimension of the injured ambulance and hospital allocation need to be determined based on a random number, that is, a random number is generated and rounded up, and then it is set to the initial position of the dimension. We need to consider the order in which the injured are transported by ambulance. We sort the decimal parts of the first 1 to I dimensions of the particles in ascending order. Then in this coding demonstration, the relative position index result of this step of treatment order is $(1,4,2,5,3,6)$, and then the transport order result of specific injured is (I1, I4, I2, I5, I3, I6). Where (I1, I4) represents the transport order of the injured coded as 1 before the transport order of the injured coded as 4 when they are transported in the same ambulance. Combined with the injured vehicle scheduling scheme and the hospital allocation of the injured, the final scheduling scheme can be obtained. For example, {Ambulance 1, (I5, H2, I6, H1)} indicates that the ambulance coded 1 transports the injured coded 5 to the hospital coded 2 and then returns to the scene to transport the injured coded 6 to the hospital coded 1.

5. MATHEMATICAL MODELLING CALCULATION EXPERIMENT

In order to verify the effectiveness of our proposed model and optimized algorithm in solving problems, we designed a computational experiment.

5.1 Case description

We hypothesized that a large-scale injury event occurred within the National Stadium (Nest) in Beijing, resulting in many critically ill and non-critically ill injured requiring urgent medical care. Table I shows the injured code (*IC*), injury level (*IL*) and waiting time (*WT*) of 100 injured people.

Table I: Injured condition form.

Location: (116.3959, 39.9934)														
IC	***IL***	***WT*** (h)	***IC***	***IL***	***WT*** (h)	***IC***	***IL***	***WT*** (h)	***IC***	***IL***	***WT*** (h)	***IC***	***IL***	***WT*** (h)
1	II	0.07	21	I	0.12	41	I	0.13	61	I	0.15	81	I	0.05
2	II	0.12	22	I	0.18	42	III	0.10	62	I	0.08	82	III	0.25
3	I	0.08	23	III	0.38	43	II	0.19	63	III	0.19	83	II	0.10
4	III	0.16	24	II	0.55	44	II	0.24	64	II	0.43	84	II	0.15
5	II	0.05	25	II	0.05	45	III	0.22	65	II	0.45	85	III	0.48
6	III	0.18	26	III	0.25	46	I	0.55	66	III	0.25	86	I	0.42
7	I	0.08	27	II	0.10	47	III	0.08	67	II	0.38	87	III	0.04
8	III	0.13	28	III	0.55	48	III	0.56	68	III	0.00	88	III	0.35
9	III	0.15	29	III	0.08	49	II	0.14	69	I	0.05	89	II	0.25
10	III	0.20	30	III	0.02	50	III	0.19	70	III	0.33	90	III	0.33
11	II	0.38	31	III	0.44	51	I	0.05	71	III	0.48	91	I	0.58
12	I	0.00	32	II	0.15	52	III	0.45	72	II	0.26	92	III	0.11
13	III	0.05	33	I	0.25	53	II	0.31	73	III	0.35	93	II	0.55
14	II	0.00	34	II	0.33	54	II	0.33	74	II	0.00	94	II	0.29
15	II	0.08	35	III	0.08	55	III	0.11	75	III	0.16	95	III	0.57
16	III	0.20	36	II	0.11	56	II	0.36	76	II	0.10	96	II	0.42
17	I	0.35	37	I	0.55	57	III	0.35	77	I	0.22	97	III	0.23
18	III	0.28	38	III	0.59	58	I	0.11	78	III	0.28	98	I	0.01
19	III	0.07	39	III	0.07	59	III	0.22	79	III	0.38	99	III	0.29
20	II	0.10	40	II	0.12	60	II	0.26	80	II	0.22	100	III	0.45

Table II shows the conditions of 20 ambulances including the ambulance code (*AC*), available time (*AT*) and available position (*AP*). Table III shows the hospital location (*HL*) and hospital capacity (*HC*) of 10 large hospitals near the National Stadium.

Table II: Ambulance condition form.

Velocity:72 km/h					
AC	***AT*** (h)	***AP***	***AC***	***AT*** (h)	***AP***
1	0.00	(116.3808, 39.9320)	11	0.05	(116.1322, 39.9309)
2	0.13	(116.1514, 39.9886)	12	0.08	(116.1905, 40.0100)
3	0.12	(116.2026, 40.0264)	13	0.03	(116.0711, 39.9608)
4	0.31	(116.0522, 39.9068)	14	0.02	(116.2321, 39.9911)
5	0.00	(116.2765, 39.9045)	15	0.00	(116.4199, 40.0289)
6	0.20	(116.2239, 40.0597)	16	0.11	(116.1042, 39.9579)
7	0.00	(116.4033, 39.9731)	17	0.15	(116.1890, 39.9612)
8	0.05	(116.1213, 39.9908)	18	0.10	(116.0832, 39.9287)
9	0.00	(116.4542, 39.9253)	19	0.00	(116.3600, 39.9827)
10	0.14	(116.2114, 39.9716)	20	0.17	(116.1114, 40.0408)

Table III: Hospital condition form.

Code	***HL***	***HC*** (person)	**Code**	***HL***	***HC*** (person)
1	(116.3808, 39.9320)	18	6	(116.4033, 39.9731)	16
2	(116.2765, 39.9045)	15	7	(116.4542, 39.9253)	19
3	(116.2987, 40.0239)	14	8	(116.4199, 40.0289)	11
4	(116.3206, 39.9800)	17	9	(116.4134, 39.9617)	14
5	(116.3155, 39.9766)	9	10	(116.3600, 39.9827)	17

5.2 Experimental design

The RBBPSO proposed in the literature [20] and the original BBPSO [18] are selected for comparison. All selected algorithms are programmed in MATLAB. Each algorithm is applied to solve the problem above and is run 30 times independently. The software environment for numerical experiments is the R2024b version of MATLAB. The hardware environment for numerical experiments is a laptop with a x64-processor Intel® Core™ i7-8550U CPU @ 1.80 GHz 1.99 GHz and 16 GB RAM.

5.3 Parameter settings

The parameters of the algorithm are set according to the literature and experiments. The parameters N_s (value: 100), *MaxIter* (value: 5000) are set according to the experiment, and the remaining parameter *a* (value: 8) is set according to the literature [20].

6. EXPERIMENTAL RESULT

Fig. 4 shows the average convergence of the algorithm for the optimal solution of this example. Compared with BBPSO and the RBBPSO algorithm, IBBPSO has faster convergence speed and higher performance in the early stage, and the convergence speed is reduced in the later stage, which is beneficial to avoid the algorithm falling into the local optimum.

Fig. 5 shows the performance comparison of IBBPSO, RBBPSO and BBPSO in terms of fitness. Whether it is the maximum value, the minimum value, the average value, or the standard deviation between three algorithms, the result of IBBPSO is the smallest. Fig. 6 shows the performance of three algorithms from the perspective of running time in this case. The minimum running time, maximum running time, average running time and standard deviation of IBBPSO are better than RBBPSO and are very close to BBPSO.

Figs. 7 and 8 are the Gantt chart results of the case study simulation. From Fig. 7, we can see the actual waiting time and transportation time of each injured person clearly for the case with IBBPSO. Fig. 8 shows the Gantt chart of the ambulance using the IBBPSO algorithm in the case. For example, I6 represents the injured numbered 6. The part without number indicates that the ambulance is empty and heading to the injured waiting for delivery.

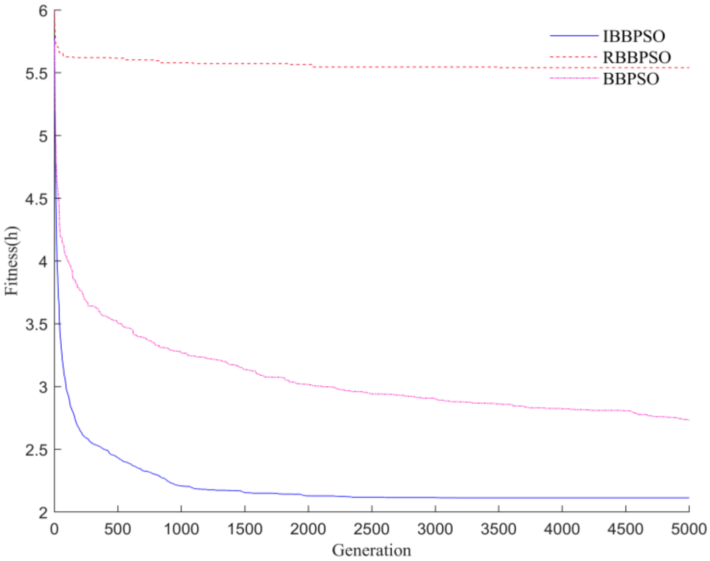

Figure 4: The average convergence of the algorithms for the optimal solution of the case.

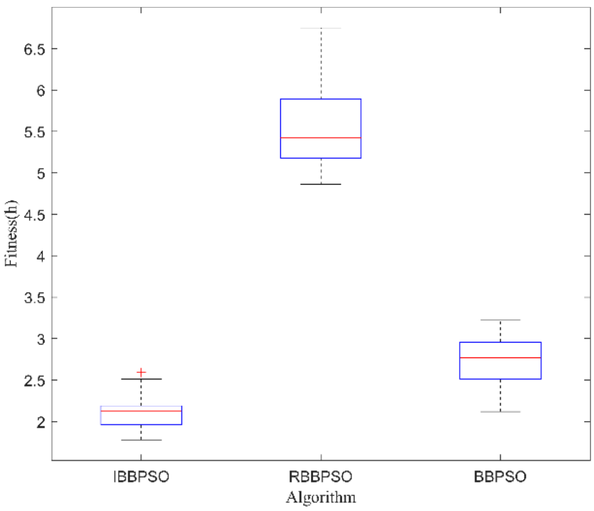

Figure 5: The statistical analysis of the fitness.

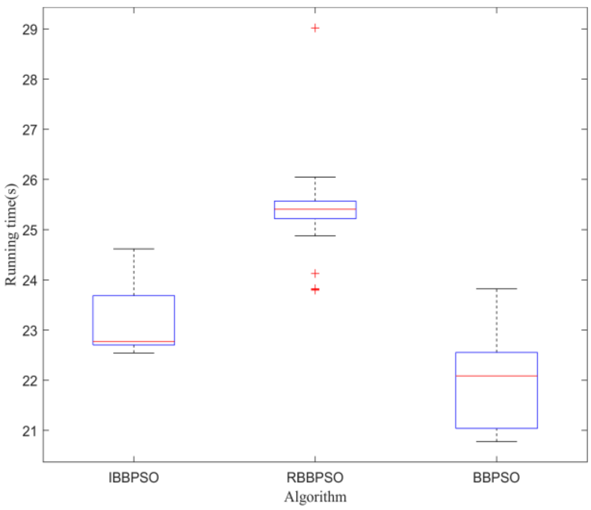

Figure 6: The statistical analysis of the running time.

Figure 7: The Gantt chart of injured.

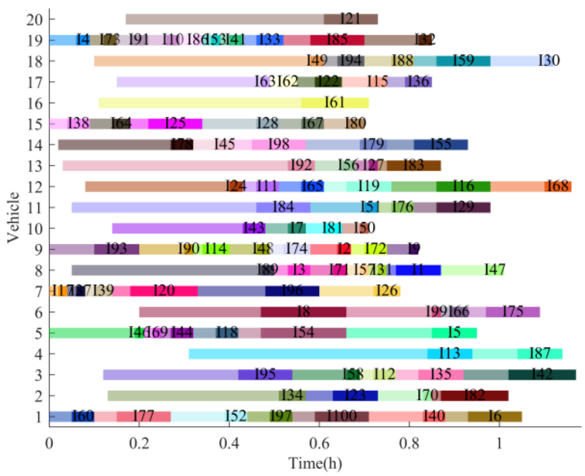

Figure 8: The Gantt chart of vehicles.

In the test example, by improving BBPSO, the running time of the algorithm becomes slightly longer than BBPSO, which can be regarded as a reasonable time increase. IBBPSO also shows significant advantages in finding solution. Compared with RBBPO, the minimum, maximum and average values of the maximum injured transfer time found by IBBPSO are 62 %, 63 % and 62 % less, respectively. Similarly, IBBPSO is 20 %, 16 % and 23 % less than the minimum, maximum and average of the maximum injured transit time found by BBPSO. The standard deviation of IBBPSO is also the smallest, which means that IBBPSO not only has high quality of solution but also has higher stability and consistency.

7. DISCUSSION AND CONCLUSION

Aiming at the optimization problem of emergency resource scheduling for public emergencies in large-scale sports events, the contributions of this paper are:

(1) This paper proposes a novel emergency rescue system after public emergencies in large-scale sports events.

(2) We construct a single-objective model that minimizes the maximum waiting and transportation time of the injured to make the rescue time of the injured more balanced considering the different circumstances of the injured.

(3) To find a stable scheduling scheme in a short time, this paper proposes IBBPSO by introducing the BBPSO into the chaotic mapping randomization population strategy, the particle position update strategy, the particle mutation strategy and the constraint processing function.

The emergency rescue system, scheduling model and optimization algorithm designed in this paper effectively provide a more efficient emergency medical resource scheduling solution for large sports events and provide strong technical support and practical basis for solving emergency rescue problems in society.

Due to the different emphases of the objectives considered in the resource scheduling problem, the model is also different. we suggest that the mathematical model can include consideration of the treatment expertise of different hospitals to determine which hospital is the most suitable for each of the injured in the future.

ACKNOWLEDGEMENT

This work was supported by the Beijing Social Science Foundation [Grant number 21GLC044].

REFERENCES

[1] Aylwin, C. J.; König, T. C.; Brennan, N. W.; Shirley, P. J.; Davies, G.; Walsh, M. S.; Brohi, K. (2006). Reduction in critical mortality in urban mass casualty incidents: analysis of triage, surge, and resource use after the London bombings on July 7, 2005, *The Lancet*, Vol. 368, No. 9554, 2219-2225, doi:10.1016/S0140-6736(06)69896-6

[2] Sung, I.; Lee, T. (2016). Optimal allocation of emergency medical resources in a mass casualty incident: patient prioritization by column generation, *European Journal of Operational Research*, Vol. 252, No. 2, 623-634, doi:10.1016/j.ejor.2016.01.028

[3] Yi, W.; Özdamar, L. (2007). A dynamic logistics coordination model for evacuation and support in disaster response activities, *European Journal of Operational Research*, Vol. 179, No. 3, 1177-1193, doi:10.1016/j.ejor.2005.03.077

[4] Jin, S.; Jeong, S.; Kim, J.; Kim, K. (2015). A logistics model for the transport of disaster victims with various injuries and survival probabilities, *Annals of Operations Research*, Vol. 230, No. 1, 17-33, doi:10.1007/s10479-013-1515-0

[5] Repoussis, P. P.; Paraskevopoulos, D. C.; Vazacopoulos, A.; Hupert, N. (2016). Optimizing emergency preparedness and resource utilization in mass-casualty incidents, *European Journal of Operational Research*, Vol. 255, No. 2, 531-544, doi:10.1016/j.ejor.2016.05.047

[6] Dean, M. D.; Nair, S. K. (2014). Mass-casualty triage: distribution of victims to multiple hospitals using the SAVE model, *European Journal of Operational Research*, Vol. 238, No. 1, 363-373, doi:10.1016/j.ejor.2014.03.028

[7] Wilson, D. T.; Hawe, G. I.; Coates, G.; Crouch, R. S. (2013). A multi-objective combinatorial model of casualty processing in major incident response, *European Journal of Operational Research*, Vol. 230, No. 3, 643-655, doi:10.1016/j.ejor.2013.04.040

[8] Rabbani, M.; Oladzad-Abbasabady, N.; Akbarian-Saravi, N. (2022). Ambulance routing in disaster response considering variable patient condition: NSGA-II and MOPSO algorithms, *Journal of Industrial and Management Optimization*, Vol. 18, No. 2, 1035-1062, doi:10.3934/jimo.2021007

[9] Wang, X. (2023). Analysis of the theory and traffic scheduling for transit network by genetic algorithm-based optimization technique, *Technical Gazette*, Vol. 30, No. 6, 1935-1942, doi:10.17559/TV-20230717000810

[10] Escobar, D.; Chivata, B.; Nino, K. (2024). A bi-objective genetic algorithm for flexible flow shop scheduling: a real-world application in the electrical industry, *Advances in Production Engineering & Management*, Vol. 19, No. 4, 415-434, doi:10.14743/apem2024.4.516

[11] Kennedy, J.; Eberhart, R. (1995). Particle swarm optimization, *Proceedings of ICNN'95 – International Conference on Neural Networks*, Vol. 4, 1942-1948, doi:10.1109/ICNN.1995.488968

[12] Wisittipanich, W.; Wisittipanit, N. (2024). Tri-objective parallel machine with job splitting and sequence dependent setup times using differential evolution and particle swarm optimization, *International Journal of Industrial Engineering and Management*, Vol. 15, No. 4, 264-278, doi:10.24867/IJIEM-2024-4-362

[13] Dorigo, M.; Birattari, M.; Stützle, T. (2006). Ant colony optimization, *IEEE Computational Intelligence Magazine*, Vol. 1, No. 4, 28-39, doi:10.1109/MCI.2006.329691

[14] Nainggolan, N.; Maghsoudlou, E.; AlWadi, B. M.; Atamurotov, F.; Kosov, M. E.; Putra, W. (2024). Advancements in optimization for automotive manufacturing: hybrid approaches and machine learning, *International Journal of Industrial Engineering and Management*, Vol. 15, No. 3, 254-263, doi:10.24867/IJIEM-2024-3-361

[15] Varela, L.; Putnik, G.; Vieira, G.; Manupati, V.; Alves, C. (2023). Group decision making approach for ranking and selecting maintenance tasks for joint scheduling with production orders, *International Journal for Quality Research*, Vol. 18, No. 1, 235-258, doi:10.24874/IJQR18.01-16

[16] Bhuvana, J.; Katariya, J. K.; Bhardwaj, S. (2023). Optimizing production scheduling through hybrid dynamic genetic-adaptive improved gravitational optimization algorithm, *Proceedings on Engineering Sciences*, Vol. 5, No. S1, 79-88, doi:10.24874/PES.SI.01.010

[17] Tlili, T.; Harzi, M.; Krichen, S. (2017). Swarm-based approach for solving the ambulance routing problem, *Procedia Computer Science*, Vol. 112, 350-357, doi:10.1016/j.procs.2017.08.012

[18] Dağlayan, H.; Karakaya, M. (2016). An optimized ambulance dispatching solution for rescuing injures after disaster, *Universal Journal of Engineering Science*, Vol. 4, No. 3, 50-57, doi:10.13189/ujes.2016.040303

[19] Kennedy, J. (2003). Bare bones particle swarms, *Proceedings of the 2003 IEEE Swarm Intelligence Symposium*, 80-87, doi:10.1109/SIS.2003.1202251

[20] Zhang, Y.; Gong, D.-W.; Ding, Z. (2012). A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch, *Information Sciences*, Vol. 192, 213-227, doi:10.1016/j.ins.2011.06.004

[21] Blackwell, T. (2012). A study of collapse in bare bones particle swarm optimization, *IEEE Transactions on Evolutionary Computation*, Vol. 16, No. 3, 354-372, doi:10.1109/TEVC.2011.2136347

[22] Guo, J.; Sato, Y. (2020). A standardized bare bones particle swarm optimization algorithm for traveling salesman problem, *International Journal of Machine Learning*, Vol. 10, No. 3, 477-481, doi:10.18178/ijmlc.2020.10.3.960

[23] Li, C.; Hu, H.; Gao, H.; Wang, B. (2016). Adaptive bare bones particle swarm optimization for feature selection, *2016 Chinese Control and Decision Conference (CCDC)*, 1594-1599, doi:10.1109/CCDC.2016.7531238

[24] Omran, M.; Al-Sharhan, S. (2007). Barebones particle swarm methods for unsupervised image classification, *2007 IEEE Congress on Evolutionary Computation*, 3247-3252, doi:10.1109/CEC.2007.4424888