Intelligent Automation And Soft Computing, 2020
Vol. 26, no. 4, 703–713
DOI: 10.32604/iasc.2020.010104

A Generalized Museum Visitor Routing Problem Considering the Preferences and Congestion for Multiple Groups: An Example in Taiwan

Yi-Chih Hsieh[1] and Peng-Sheng You[2]

[1] Department of Industrial Management, National Formosa University, Yunlin, Taiwan
[2] Department of Business Administration, National ChiaYi University, Chia-Yi, Taiwan

ABSTRACT
In this study we present a generalized museum visitor routing problem considering the preferences and congestion of multiple groups with each group having its own must-see and select-see exhibition rooms based on their preferences in exhibits. The problem aims to minimize the makespan of all groups. An effective encoding scheme is proposed to simultaneously determine the scheduling of exhibition rooms for all groups and an immune-based algorithm (IBA) is developed. Numerical results, compared with those of a genetic algorithm and particle swarm optimization, on a museum in Taiwan are reported and discussed to show the performance of the IBA.

KEY WORDS: Generalized museum visitor routing problem, Preference, Congestion, Algorithm, Optimization.

1 INTRODUCTION

VISITING museums is a popular activity among student groups. In Taiwan, students are usually divided into multiple groups based on their preferences and arrive at the museum at the same time. With an advanced appointment, a guide will be scheduled for each exhibition room to show the groups around exhibits. To enhance the visiting quality, the visiting routes of groups are commonly arranged to avoid congestion or queuing such that all groups can depart from the museum together.

In 2007, Chou and Lin (2007) first studied a museum visitor routing problem (MVRP) for multiple groups in which each group had to visit all exhibition rooms with a minimized makespan. They showed that MVRP is NP-hard and solved it by using a simulated annealing (SA) approach. Moreover, Yu et al. (2010) solved the MVRP through a SA approach with a neighborhood search and compared numerical results with those by other evolutionary approaches, including SA, genetic algorithm (GA), and ant colony optimization approach.

In this study, we consider a generalized MVRP, which is called Generalized MVRP with the consideration of Preferences and Congestion (herein referred as GMVRP-PC). The new GMVRP-PC is more practical and differs from MVRP because of the following reasons:

(1) GMVRP-PC assumes that each group may have various must-see and select-see rooms. That is, each group has to visit its own must-see exhibition rooms and a given number of its own select-see exhibition rooms only. However, MVRP assumes that all groups have to visit all exhibition rooms.
(2) GMVRP-PC assumes that each group may have different interests for exhibition rooms and can set its own must-see and select-see rooms. However, MVRP assumes that all groups have to visit all exhibition rooms without considering the group's special preference.

As the MVRP, the objective of GMVRP-PC aims to minimize the makespan, that is, to minimize the maximal completion time of groups to avoid congestion. In 2015, Hsieh and You (2017) introduced a multiple-type MVRP (MT-MVRP) in which the exhibition rooms in a museum are classified into must-see and select-see exhibition rooms. However, the new GMVRP-PC generalizes and differs from MT-MVRP because of the following reasons:

(1) GMVRP-PC assumes that all groups may have different must-see exhibition rooms based on their preferences but MT-MVRP assumes that all groups have the same must-see exhibition rooms.
(2) GMVRP-PC assumes that all groups may have multiple various candidate sets of select-see rooms based on their preferences but MT-MVRP assumes that all groups have only one identical candidate set of select-see rooms.

CONTACT Peng-Sheng You psyuu@mail.ncyu.edu.tw

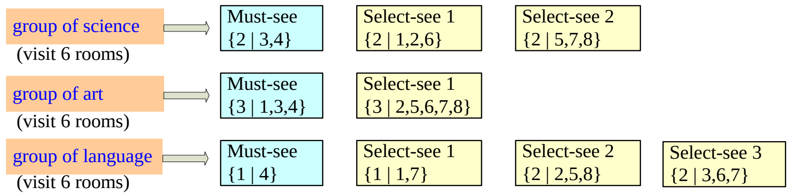

Figure 1. **Example of GMVRP-PC.**

Figure 1 illustrates an example of GMVRP-PC in which three groups in a school, namely, science, art, and language groups, have different must-see exhibition rooms and multiple various candidate sets of select-see rooms. As shown in Figure 1, $\{k \mid x(1), x(2),\ldots, x(n)\}$ denotes selecting k rooms from the candidate set of $\{x(1), x(2),\ldots, x(n)\}$. Therefore, in Figure 1, the science group has to visit two must-see exhibition room {3, 4}, two select-see exhibition rooms from the candidate set of {1, 2, 6} and two select-see exhibition rooms from the candidate set of {5, 7, 8}.

MVRP can be treated as an open-shop scheduling problem (OSSP), which implies that it is NP-hard (Chou and Lin, 2007). In the past, various approaches have been proposed to solve OSSPs, and they are briefly summarized in Table 1. More details of approaches for solving OSSPs are found in the excellent survey paper by Çaliş and Bulkan (2015). For solving MVRP, Chou and Lin (2007) and Yu et al. (2010) proposed SA and neighborhood approaches. However, the main disadvantage of SA is that it may be extremely slow and consumes time for solving a complex problem. Consequently, SA has not been widely applied for complicated engineering problems (Henderson et al., 2003). The considered GMVRP-PC generalizes MVRP and is a complicated optimization problem. Therefore, in this study, we have not adopted SA to solve the GMVRP-PC.

In the past, several artificial intelligence algorithms have been proposed to solve optimization problems. Mirjalili and Lewis (2016) have surveyed more than 10 novel optimization algorithms. Recently, immune-based algorithm (IBA), which was firstly proposed by Jerne (1973), has attracted more attention than most of the other metaheuristic methods due to its several successful applications, for example, Basu (2011). The evolutionary scheme of IBA is similar to that of GA except for the memory mechanisms. With the memory mechanisms, IBA can obtain more distinct optimal solutions. In addition, Particle Swarm Optimization (PSO) and GA can be the most popular evolutionary algorithms in the literature due to their success in different types of optimization problems, for example, Erchiqui (2018), Tasan and Tunali (2008), Srinivas and Patnaik (1994), AlRashidi and El-Hawary (2009), and Lin et al. (2017).

In this study, we focus on the main purpose of presenting a new GMVRP-PC and a novel encoding scheme embedded in IBA to effectively solve GMVRP-PC. The presented encoding scheme can simultaneously determine the scheduling of must-see and select-see exhibition rooms for all the groups. In addition, numerical results by IBA on a museum in Taiwan are reported and compared with those by GA and PSO.

2 GMVRP-PC

2.1 Notations

n the number of groups

m the number of exhibition rooms

M_i the set of must-see exhibition rooms for group i, $M_i \subseteq \{1, 2,\ldots, m\}$, $1 \le i \le n$

Table 1. **Related references of OSSP.**

Authors	Year	Approach	Remarks
Adiri and Aizikowitz (1989)	1989	polynomial time algorithm	special structure of OSSP
Fiala (1983)	1983	polynomial time algorithm	special structure of OSSP
Dorndorf et al. (2001)	2001	branch-and-bound method	time consuming
Gueret and Prins (1998)	1998	approximation algorithm	approximation algorithm
Brasel et al. (1993)	1993	insertion algorithm	approximation algorithm
Liaw (1998)	1998	recursive algorithm	approximation algorithm
Taillard (1993)	1993	genetic algorithm	evolutionary algorithm
Fang et al. (1994)	1994	genetic algorithm	evolutionary algorithm
Prin (2000)	2000	genetic algorithm	evolutionary algorithm
Liaw (1999a)	1999	simulated annealing	evolutionary algorithm
Liaw (1999b)	1999	tabu search	evolutionary algorithm
Panahi and Tavakkoli-Moghaddam (2011)	2011	ant colony optimization	evolutionary algorithm
Huang and Lin (2011)	2011	bee colony optimization	evolutionary algorithm
Liu and Ong (2004)	2004	hybrid algorithm	evolutionary algorithm
Roemer (2006)	2006	theory	complexity for special OSSP
Kononov (2012)	2012	theory	complexity for special OSSP

m_i the number of must-see exhibition rooms for group i, $1 \le i \le n$

u_i the number of candidate sets of select-see exhibition rooms for group i, $1 \le i \le n$

S_{ip} the p-th candidate set of select-see exhibition rooms for group i, $S_{ip} \subseteq \{1, 2, \ldots, m\}$, $M_i \cap S_{ip} = \phi, S_{ip} \cap S_{iq} = \phi$, $p \neq q$, $1 \le p \le u_i$, $1 \le q \le u_i$, $1 \le i \le n$

s_{ip} the number of selected exhibition rooms from the p-th candidate set for group i, $s_{ip} \le |S_{ip}| \le m$, $1 \le p \le u_i$, $1 \le i \le n$

v_{ik} the time required to visit exhibition room k for group i, $1 \le i \le n$, $1 \le k \le m$

t_{hk} the time required to move from exhibition room h to exhibition room k, $h \neq k$, $1 \le h \le m$, $1 \le k \le m$

t_{0k} the moving time from the entrance to exhibition room k, $1 \le k \le m$

t_{k0} the moving time from exhibition room k to the exit, $1 \le k \le m$

c_{i0} the completion time of visit for group i, $1 \le i \le n$

2.2 Assumptions

(1) n groups are intended to visit a museum with m exhibition rooms.

(2) All groups arrive at the museum at the same time and have to depart from the museum together.

(3) Group i has to visit m_i (must-see) + s_{i1} (select-see) + s_{i2} (select-see)+…+s_{iui} (select-see) exhibition rooms. For a GMVRP-PC, M_i, m_i, S_{ip} and s_{ip} are given based on the group's preference.

(4) The routing sequence of the exhibition rooms for all groups has no restrictions.

(5) Each exhibition room can contain one group only at a time, that is, one tour guide in each exhibition room.

(6) The visit of a group in each room cannot be interrupted.

(7) The GMVRP-PC aims to simultaneously determine the following:

(a) the select-see exhibition rooms from each corresponding candidate set of select-see rooms for each group,

(b) the routing sequence of the exhibition rooms for all of the groups, including the must-see and select-see rooms.

2.3 Properties

The properties of GMVRP-PC are shown below.

Property 2.1. *The GMVRP-PC generalizes the conventional MVRP.*

Proof. If $m_i+s_{i1}+s_{i2}+\ldots+s_{iui}=m$ for all $1 \le i \le n$, then each group has to visit all of the exhibition rooms. Thus, GMVRP-PC leads to the conventional MVRP, which further implies that GMVRP-PC is also an NP-hard problem.

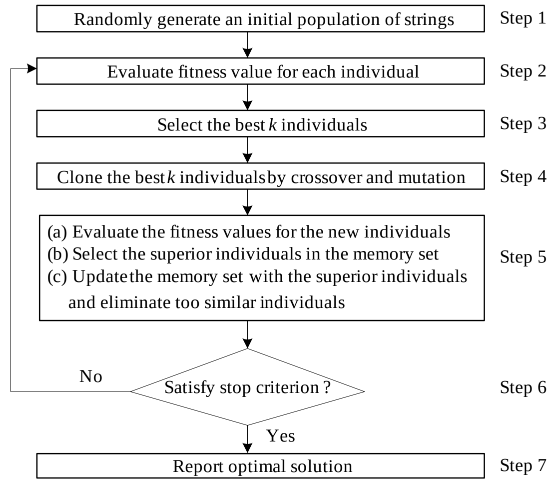

Figure 2. Main steps of IBA.

Property 2.2.*The considered GMVRP-PC generalizes the MT-MVRP.*

Proof. If $M_i = M_j$, $u_i = u_j = 1$, and $s_{i1} = s_{j1}$ for $1 \le i \le n$, $1 \le j \le n$, $i \neq j$, then each group has the same muse-see exhibition rooms and has to select the same number of select-see exhibition rooms from the identical candidate set of select-see rooms. As such, GMVRP-PC leads to MT-MVRP.

3 METHODOLOGY

THE considered GMVRP-PC is an NP-hard problem (Property 2.1), and its objective is to minimize the maximal makespan of all groups, that is., min max c_{i0}, $1 \le i \le n$. Therefore, instead of solving GMVRP-PC through a mathematical programming approach, we focus on the multiple main purposes of this study: (a) present a novel encoding scheme to convert any random permutation of a sequence of integers into a feasible solution of GMVRP-PC, (b) embed the encoding scheme in IBA to solve GMVRP-PC, and (c) compare numerical results of a museum in Taiwan with those by GA and PSO to show the effectiveness of the adopted IBA approach. To shorten the paragraph of this paper, the main steps of IBA are introduced in Section 3.1 and the main steps of GA and PSO are referred to Srinivas and Patnaik (1994), Lim (2014), Michalewicz (1994), AlRashidi and El-Hawary (2009), and Kennedy and Eberhart (1995)

3.1 IBA

The main steps of IBA are illustrated in Figure 2 and they are similar to those of Hsieh and You (2014). IBA is similar to GA except for the memory mechanism, that is, Step 5(c) in Figure 2. Next, to shorten the paragraph, we refer to Weissman and Cooper (1993) and Al-Enezi et al. (2010) for the details of IBA and its mechanism, and we briefly describe the basic operation of memory mechanism of IBA below.

To keep the diversity of solution for GMVRP-PC, IBA updates the memory set of solutions by replacing the inferior ones with the superior ones for each iteration. However, unlike GA, IBA deletes highly similar solutions even though their objectives are good. The similarity of two solutions are computed by $f_{ij}(t)=\|x_i(t)-x_j(t)\|$, where $\|x_i(t)-x_j(t)\|$ is the Euclidean distance of two solutions x_i and x_j at iteration t. Specifically, if $f_{ij}(t)\le\delta$, δ is a given threshold value, then x_i and x_j are too similar. In this study, we set δ=0 in IBA, that is, all solutions in the memory set are different.

3.2 New encoding scheme

In this section, we present the main steps of the new encoding scheme for simultaneously determining the sequence of the exhibition rooms for all groups, including the must-see and select-see exhibition rooms. In the encoding scheme, we convert a random permutation of {1, 2,…, N} with the operator of mod to represent a feasible routing sequence of exhibition rooms for all groups, where N is the total number of exhibition rooms visited for all groups, including the must-see and select-see exhibition rooms.

The main steps of the encoding scheme are as follows:

Step 0. $R=\phi$, given M_i, S_{ip} and s_{ip}, $1\le i\le n$, $1\le p\le u_i$.

Step 1. Generate a random permutation of {1, 2,…, N}, termed T, where $N = \Sigma(m_i+s_{i1}+s_{i2}+\ldots+s_{iui})$ is the total number of rooms visited for all groups, including the must-see and select-see exhibition rooms.

Step 2. Based on the total number of exhibition rooms for each group, divide T into groups with $T=(T_{11}\cup T_{12}\cup\ldots\cup T_{1u_i+1})\cup\ldots\cup(T_{n1}\cup T_{n2}\cup\ldots\cup T_{nu_i+1})$.

Step 3. Construct a table as Table 5. Specifically, for each i, $1\le i\le n$, execute the following steps in (a)-(b).

(a) Let w be the order of T_{i1} and $R=T_{i1}(w)$.

(b) (i) For all p, $1\le p\le u_i$, do (ii).

(ii) Let w be the first number of $T_{i,p+1}$. Add the $\{(w \bmod |S_{ip}|)+1\}^{th}$ number in set S_{ip} to its corresponding location in R; delete it from set S_{ip}, that is, R(index) = the $\{(w \bmod |S_{ip}|)+1\}^{th}$ number in set S_{ip}. Delete w from $T_{i,p+1}$, and delete R(index) from S_{ip}. Repeat (ii) until $T_{i,p+1}=\phi$.

Step 4. Follow the sequence of T, from 1 to N, and its corresponding value in R to construct the routing for each group.

The main computation of the above encoding scheme is on the order operation of T_{i1} and the mod operation in Step 3. In Step 3, for each i, $1\le i\le n$, the complexity of the worst case for the order operation of T_{i1} is $O(m^2)$ and the complexity of the worst case for the mod operation is $O(m)$. Therefore, the complexity of the worst case of the proposed encoding scheme is $O(nm^2)$. It further implies that the computational complexity of the scheme is polynomial. An example of the presented encoding scheme is shown in Section 3.3.

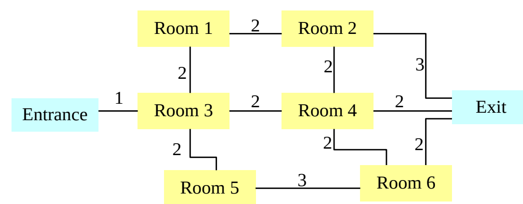

Figure 3. Network for the example of GMVRP-PC.

Table 2. Moving time between exhibition rooms, entrance, and exit (example).

Moving Time (t_{hk})	Room 1	2	3	4	5	6
Room 1	-	2	2	4	4	6
Room 2	2	--	4	2	6	4
Room 3	2	4	--	2	2	4
Room 4	4	2	2	--	4	2
Room 5	4	6	2	4	--	3
Room 6	6	4	4	2	3	--
Entrance (t_{0k})	3	5	1	3	3	5
Exit (t_{k0})	5	3	4	2	5	2

Table 3. Visiting time (v_{ik}) of the group (example).

Visiting time	Room 1	2	3	4	5	6
Group 1	6	8	5	6	4	8
Group 2	6	7	5	4	6	6
Group 3	5	6	7	3	6	7
Group 4	4	6	6	6	5	6
Group 5	7	6	5	8	5	4
Group 6	5	6	6	8	5	7
Group 7	5	4	6	6	4	8
Group 8	5	4	6	6	5	6

Table 4. Exhibition rooms (example).

Room	Must-See	Select-See 1	Select-See 2
Group 1	{2 | 1,2}	{1 | 3,4}	{1 | 5,6}
Group 2	{1 | 2}	{2 | 1,3,4}	{1 | 5,6}
Group 3	{2 | 2,3}	{1 | 1,5}	{1 | 4,6}
Group 4	{2 | 2,5}	{1 | 1,4}	{1 | 3,6}
Group 5	{1 | 2}	{2 | 1,3,4}	{1 | 5,6}
Group 6	{2 | 1,2}	{1 | 3,4}	{1 | 5,6}
Group 7	{1 | 2}	{1 | 1,4}	{2 | 3,5,6}
Group 8	{2 | 2,3}	{1 | 1,5}	{1 | 4,6}

$\{k\,|x(1),\ldots,x(n)\}$ indicates selecting k rooms from $x(1),\ldots,x(n)$.

Table 5. Encoding scheme (example).

Group(G)	1				2				3				4			
Room	M		S_{11}	S_{12}	M	S_{21}		S_{22}	M		S_{31}	S_{32}	M		S_{41}	S_{42}
Type	T_{11}		T_{12}	T_{13}	T_{21}	T_{22}		T_{23}	T_{31}		T_{32}	T_{33}	T_{41}		T_{42}	T_{43}
Index (I)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	26	6	25	10	21	14	11	28	2	23	13	12	15	18	31	24
R	2	1	4	5	2	4	3	5	2	3	5	4	2	5	4	3

Group(G)	5				6				7				8			
Room	M	S_{51}		S_{52}	M		S_{61}	S_{62}	M	S_{71}	S_{72}		M		S_{81}	S_{82}
Type	T_{51}	T_{52}		T_{53}	T_{61}		T_{62}	T_{63}	T_{71}	T_{72}	T_{73}		T_{81}		T_{82}	T_{83}
Index (I)	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
T	22	1	20	4	8	5	19	29	16	27	32	3	30	17	9	7
R	2	3	1	5	2	1	4	6	2	4	6	5	3	2	5	6

M=must-see, S_{i1}=Candidate set 1 of select-see rooms, S_{i2}=Candidate set 2 of select-see rooms for group *i*.

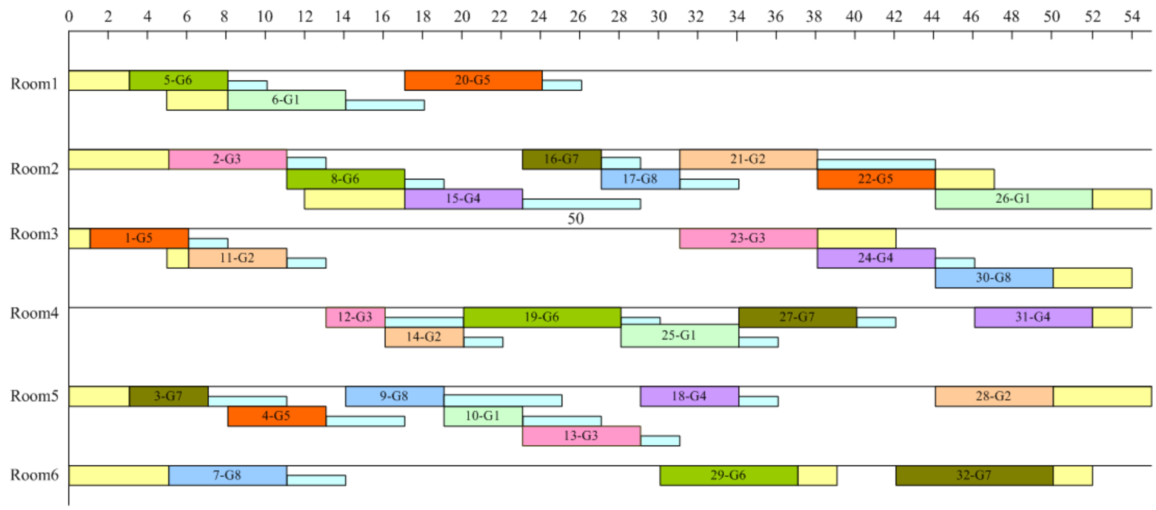

Figure 4. Gantt chart for the example (moving time, entrance/exit time).

3.3 Example

Assume that eight groups visit six exhibition rooms (i.e., *n*=8 and *m*=6) in a small museum. The exhibition rooms are illustrated in Figure 3, and the moving and visiting times for the groups are shown in Tables 2 and 3. Suppose that all groups have to visit one set of must-see exhibition rooms and two sets of select-see exhibition rooms, which are illustrated in Table 4. For example, Group 1 has to visit must-see rooms {1, 2}, one select-see room from candidate set {3, 4}, and one select-see room from candidate set {5, 6}.

As each group has to visit 4 rooms, N=32(=8×4). Suppose that T=26-6-25-10-21-14-11-28-2-23-13-12-15-18-31-24-22-1-20-4-8-5-19-29-16-27-32-3-30-17-9-7 is a random permutation of 1 to 32. Following the steps in Section 3.2, we have Table 5. On the basis of Table 5, we obtain the routing sequence to construct its Gantt chart:

G_{53}-G_{32}-G_{75}-G_{55}-G_{61}-G_{11}-G_{86}-G_{62}-G_{85}-G_{15}-G_{23}-G_{34}-G_{35}-G_{24}-G_{42}-G_{72}-G_{82}-G_{45}-G_{64}-G_{51}-G_{22}-G_{52}-G_{33}-G_{43}-G_{14}-G_{12}-G_{74}-G_{25}-G_{66}-G_{83}-G_{44}-G_{76}

where G_{ij} denotes that group *i* visits exhibition room *j*. That is, in the Gantt chart, we assign Group 5 to visit Exhibition Room 3 first and then assign Group 3 to visit Exhibition Room 2 next, and so on. Note that:

(1) In this example, Group 1 visits Rooms 2, 1, 4, and 5; Group 2 visits Rooms 2, 4, 3, and 5; Group 3 visits Rooms 2, 3, 5, and 4; Group 4 visits Rooms 2, 5, 4, and 3, and so on. The Gantt chart for this routing sequence is illustrated in Figure 4 in which a-G_i denotes that group *i* is the *a*-th scheduled assignment in the routing sequence.

(2) The solution shown in Figure 4 is the global optimum solution for this example, because (i) Room 2 is a must-see exhibition room for all groups and (ii) the sum of visiting time of all groups for Room 2 and the entrance/exit time is (8 + 7 + 6 + 6 + 6 + 6 + 4 + 4)+(5 + 3)=55, which is identical to that presented in Figure 4.

The step-by-step encoding processes of the example in Section 3.3 are shown in Appendix.

4 NUMERICAL RESULTS AND DISCUSSIONS

4.1 Test problems and experimental results

IN this study, we test the proposed IBA for solving 12 instances of Taipei Fine Arts Museum (TFAM) in Taiwan. The plan of the museum is illustrated in Figure 5. Four types of group are established in terms of size, namely, small group, medium group, large group, and super-large group. The moving time from entrance to the exhibition rooms and exit and the visiting time of each group for exhibition room are listed in Tables 6 and 7. As assumed, the larger group takes longer time to move and visit.

In this study, to analyze the performance of the proposed approach, we vary the number of the must-see and select-see exhibition rooms for the museum, such that the percentages of the rooms visited by the groups range from 50% to 100%. The considered GMVRP-PC leads to the conventional MVRP when the percentage of rooms visited by the groups is 100%. More details of the 12 test instances are summarized in Table 8.

For each test instance, we executed IBA 50 times. Hence, the total number of trials in this study was 600 (12 instances × 50 times) for IBA. To compare the results of the IBA, we also apply GA and PSO to solve each instance. In the experiments, based upon our test experience, the parameters of IBA and GA were set as follows: population=300, iteration=1000, crossover=0.8, and mutation= 0.1; the parameters of PSO were set as follows: population=300, iteration =1000, crossover=0.8, mutation=0.1, c_1=1.49445 and c_2=1.2. In addition, three algorithms will stop in advance if no improvement in objective value is observed up to 500 iterations.

The three algorithms were coded in MATLAB. All numerical results were computed on a PC with an Intel(R) Core(TM) i5-4570 CPU3.2GHz RAM 4GB. Numerical results are reported in Tables 9 to 10, including the best, average, standard deviation, 95% confidence interval and CPU time of solutions for all algorithms.

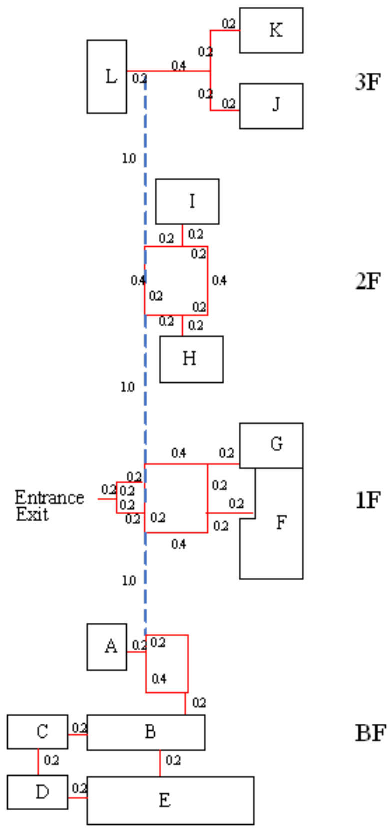

Figure 5. TFAM, Taiwan.

Table 6. Visiting time of the group for Instances 1-6 of TFAM (15 groups, 12 rooms).

Time	Group	A	B	C	D	E	F	G	H	I	J	K	L
Small Group	G1	9.0	12.5	10.3	10.0	13.7	16.6	16.5	14.4	15.9	15.7	14.7	13.6
	G2	8.9	12.1	10.1	10.3	13.5	17.0	16.9	14.6	15.6	15.9	14.5	13.5
	G3	8.8	12.2	10.2	10.3	13.5	16.5	16.8	14.8	15.8	15.4	14.8	13.4
	G4	9.0	12.6	10.4	10.1	13.6	16.8	16.7	14.5	15.8	15.6	14.6	13.7
	G5	9.2	12.3	10.3	10.2	13.3	16.9	16.5	14.3	15.7	15.5	14.7	13.2
Medium Group	G6	10.0	13.5	11.2	11.3	14.4	18.4	18.1	15.6	17.2	16.8	15.8	14.5
	G7	9.9	13.3	10.9	11.0	14.7	18.5	18.2	15.8	17.3	16.7	16.0	14.8
	G8	10.0	13.4	10.8	11.2	14.4	18.5	18.3	15.9	17.1	16.9	15.8	14.7
	G9	9.7	13.6	11.0	10.9	14.6	18.4	18.6	15.8	16.9	17.2	15.7	14.5
	G10	9.8	13.4	10.9	11.2	14.9	18.6	18.1	16.0	17.2	17.1	15.6	14.8
Large Group	G11	10.9	14.5	12.2	12.0	16.0	19.6	19.7	17.0	18.2	18.4	17.4	16.0
	G12	10.6	14.6	11.8	12.2	16.1	19.5	19.9	17.2	18.4	18.5	17.0	15.9
	G13	10.8	14.7	11.9	12.0	15.9	19.9	19.8	17.1	18.4	18.2	17.2	15.7
	G14	10.5	14.5	12.1	11.7	16.0	19.9	19.6	17.3	18.3	18.6	17.3	16.1
	G15	10.8	14.8	12.0	11.9	15.7	19.8	20.0	17.4	18.2	18.3	17.4	15.6

Table 7. **Visiting and moving times of the group for Instances 7-12 of TFAM (20 groups, 12 rooms).**

Time	Group	A	B	C	D	E	F	G	H	I	J	K	L
Small Group	G1	8.9	12.4	10.2	10.4	13.3	16.8	17.0	14.5	15.5	15.7	14.6	13.4
	G2	9.1	12.6	10.1	10.3	13.4	16.5	16.8	14.6	15.6	15.9	14.3	13.6
	G3	9.0	12.2	10.0	9.9	13.2	17.0	16.6	14.7	15.8	15.9	14.5	13.7
	G4	8.8	12.5	10.4	10.2	13.6	16.6	16.5	14.6	15.7	15.6	14.5	13.5
	G5	9.1	12.2	10.1	10.2	13.4	16.5	16.9	14.7	15.6	15.8	14.3	13.4
Medium Group	G6	9.8	13.2	11.1	10.9	14.5	18.2	18.4	15.7	17.0	17.2	15.8	14.4
	G7	10.1	13.5	11.0	11.2	14.9	18.5	18.3	16.1	17.1	17.0	15.7	14.8
	G8	10.0	13.4	11.3	10.8	14.4	18.3	18.1	16.0	16.9	17.3	16.1	14.5
	G9	10.1	13.7	11.2	11.0	14.7	18.0	18.5	15.9	17.2	16.8	15.9	14.8
	G10	9.9	13.5	10.8	11.3	14.8	18.3	18.2	16.0	17.1	16.9	15.6	14.9
Large Group	G11	10.8	14.5	12.0	12.2	15.9	20.0	19.6	17.4	18.2	18.5	17.1	15.8
	G12	10.9	14.8	11.7	11.8	15.8	19.8	20.1	17.3	18.6	18.4	17.4	15.7
	G13	10.9	14.4	11.9	12.1	16.0	19.7	19.8	17.2	18.7	18.4	17.3	15.9
	G14	10.7	14.5	12.1	11.9	15.7	19.6	19.7	17.4	18.4	18.6	17.0	15.8
	G15	10.8	14.3	11.8	12.0	15.6	19.9	19.6	17.2	18.6	18.7	17.3	15.9
Super Large Group	G16	11.6	15.6	12.7	13.0	17.3	21.1	21.3	18.5	19.9	19.6	18.6	17.0
	G17	11.5	15.9	13.1	12.8	17.2	21.0	21.4	18.7	19.8	19.7	18.3	16.9
	G18	11.7	15.7	12.9	12.7	17.1	21.5	21.1	18.4	19.7	19.6	18.5	17.2
	G19	11.6	15.9	12.8	13.0	16.9	21.4	21.2	18.6	20.1	20.0	18.4	16.9
	G20	11.5	15.8	13.0	12.8	17.0	21.2	21.4	18.5	19.8	19.7	18.6	17.3

Table 8. **Test instances.**

<table>
<tr><th>n</th><th>Group-Size</th><th>Instance</th><th>Room (A)</th><th>Must-See NRS(B)</th><th>Room</th><th>Select-See NRS(C)</th><th>Group</th><th>Candidate sets of Room</th><th>% (B+C)/A</th></tr>
<tr><td rowspan="14">15</td><td rowspan="14">5 Small
5 Medium
5 Large</td><td rowspan="3">1</td><td rowspan="3">12</td><td rowspan="3">2</td><td rowspan="3">FG</td><td rowspan="3">4</td><td>1-5</td><td>{1|AE} {2|BCDHI} {1|JKL}</td><td rowspan="3">50%</td></tr>
<tr><td>6-10</td><td>{2|ABCDE} {1|HI} {1|JKL}</td></tr>
<tr><td>11-15</td><td>{1|ABCDE} {1|HI} {2|JKL}</td></tr>
<tr><td rowspan="3">2</td><td rowspan="3">12</td><td rowspan="3">2</td><td rowspan="3">FG</td><td rowspan="3">5</td><td>1-5</td><td>{1|AE} {3|BCDHI} {1|JKL}</td><td rowspan="3">58%</td></tr>
<tr><td>6-10</td><td>{2|ABCDE} {1|HI} {2|JKL}</td></tr>
<tr><td>11-15</td><td>{2|ABCDE} {1|HI} {2|JKL}</td></tr>
<tr><td rowspan="3">3</td><td rowspan="3">12</td><td rowspan="3">3</td><td rowspan="3">FGJ</td><td rowspan="3">5</td><td>1-5</td><td>{1|AE} {2|BCD} {2|HIKL}</td><td rowspan="3">66%</td></tr>
<tr><td>6-10</td><td>{1|AE} {1|BCD} {3|HIKL}</td></tr>
<tr><td>11-15</td><td>{1|AE} {3|BCDHI} {1|KL}</td></tr>
<tr><td rowspan="3">4</td><td rowspan="3">12</td><td rowspan="3">3</td><td rowspan="3">FGJ</td><td rowspan="3">6</td><td>1-5</td><td>{1|AE} {2|BCD} {3|HIKL}</td><td rowspan="3">75%</td></tr>
<tr><td>6-10</td><td>{1|AE} {2|BCD} {3|HIKL}</td></tr>
<tr><td>11-15</td><td>{1|AE} {4|BCDHI} {1|KL}</td></tr>
<tr><td>5</td><td>12</td><td>4</td><td>FGJK</td><td>6</td><td>1-15</td><td>{4|ABCDE} {2|HIL}</td><td>83%</td></tr>
<tr><td>6</td><td>12</td><td>4</td><td>FGJK</td><td>8</td><td>1-15</td><td>{8|ABCDEHIL}</td><td>100%</td></tr>
<tr><td rowspan="18">20</td><td rowspan="18">5 Small
5 Medium
5 Large
5 Super-L</td><td rowspan="4">7</td><td rowspan="4">12</td><td rowspan="4">2</td><td rowspan="4">FG</td><td rowspan="4">4</td><td>1-5</td><td>{1|AE} {2|BCDHI} {1|JKL}</td><td rowspan="4">50%</td></tr>
<tr><td>6-10</td><td>{2|ABCDE} {1|HI} {1|JKL}</td></tr>
<tr><td>11-15</td><td>{1|ABCDE} {1|HI} {2|JKL}</td></tr>
<tr><td>16-20</td><td>{1|AE} {1|BCDHI} {2|JKL}</td></tr>
<tr><td rowspan="4">8</td><td rowspan="4">12</td><td rowspan="4">2</td><td rowspan="4">FG</td><td rowspan="4">5</td><td>1-5</td><td>{1|AE} {3|BCDHI} {1|JKL}</td><td rowspan="4">58%</td></tr>
<tr><td>6-10</td><td>{2|ABCDE} {1|HI} {2|JKL}</td></tr>
<tr><td>11-15</td><td>{2|ABCDE} {1|HI} {2|JKL}</td></tr>
<tr><td>16-20</td><td>{1|AE} {2|BCDHI} {2|JKL}</td></tr>
<tr><td rowspan="4">9</td><td rowspan="4">12</td><td rowspan="4">3</td><td rowspan="4">FGJ</td><td rowspan="4">5</td><td>1-5</td><td>{1|AE} {2|BCD} {2|HIKL}</td><td rowspan="4">66%</td></tr>
<tr><td>6-10</td><td>{1|AE} {1|BCD} {3|HIKL}</td></tr>
<tr><td>11-15</td><td>{1|AE} {3|BCDHI} {1|KL}</td></tr>
<tr><td>16-20</td><td>{1|AE} {3|BCDHI} {1|KL}</td></tr>
<tr><td rowspan="4">10</td><td rowspan="4">12</td><td rowspan="4">3</td><td rowspan="4">FGJ</td><td rowspan="4">6</td><td>1-5</td><td>{1|AE} {2|BCD} {3|HIKL}</td><td rowspan="4">75%</td></tr>
<tr><td>6-10</td><td>{1|AE} {2|BCD} {3|HIKL}</td></tr>
<tr><td>11-15</td><td>{1|AE} {4|BCDHI} {1|KL}</td></tr>
<tr><td>16-20</td><td>{1|AE} {4|BCDHI} {1|KL}</td></tr>
<tr><td>11</td><td>12</td><td>4</td><td>FGJK</td><td>6</td><td>1-20</td><td>{1|AE} {5|BCDHIL}</td><td>83%</td></tr>
<tr><td>12</td><td>12</td><td>4</td><td>FGJK</td><td>8</td><td>1-20</td><td>{8|ABCDEHIL}</td><td>100%</td></tr>
</table>

NRS=Number of Rooms Selected, where $\{k\,|x(1), x(2),\ldots,x(n)\}$ denotes selecting k rooms from $x(1),x(2),\ldots,x(n)$.

Table 9. Numerical results of three algorithms (50 experiments).

I	% visit	IA (Min	Avg	Std	CPU)	GA (Min	Avg	Std	CPU)	PSO (Min	Avg	Std	CPU)	Gap IA	GA	PSO
1	50%	278.1	278.1	0.00	3116.1	278.1	278.1	0.00	944.6	278.1	278.1	0.00	373.4	0	0	0
2	58%	278.1	278.1	0.00	3337.6	278.1	278.1	0.10	1077.4	278.1	280.2	2.70	610.7	0	0	0
3	66%	278.1	278.1	0.00	3900.3	278.1	279.9	3.41	1445.1	289.1	306.9	9.44	625.4	0	0	0.04
4	75%	278.1	278.3	0.70	4521.0	278.1	281.4	7.60	2197.7	305.6	333.9	11.15	689.7	0	0	0.1
5	83%	278.1	279.3	1.83	5294.2	278.1	286.0	14.84	2416.6	344.6	364.4	11.45	683.7	0	0	0.24
6	100%	304.3	314.3	5.67	4284.2	309.9	322.0	17.06	2926.1	402.2	434.1	14.37	720.3	0	0.02	0.32
7	50%	383.3	383.3	0.00	4010.1	383.3	383.3	0.00	1173.4	383.3	383.3	0.00	467.6	0	0	0
8	58%	383.3	383.3	0.00	4602.2	383.3	383.3	0.00	1458.8	383.3	384.0	1.49	785.7	0	0	0
9	66%	383.3	383.3	0.00	5152.2	383.3	384.6	4.84	1859.5	384.6	412.6	11.79	916.8	0	0	0
10	75%	383.3	383.3	0.00	5847.1	383.3	384.5	3.33	2420.9	407.9	439.0	10.74	789.5	0	0	0.06
11	83%	383.3	383.4	0.50	7369.0	383.3	386.2	7.30	3315.0	446.8	483.1	13.00	978.8	0	0	0.17
12	100%	387.9	403.5	7.56	6076.0	385.8	411.7	18.80	3869.9	530.3	559.6	14.52	1063.7	0.01	0	0.37

I: Instance; Gap: (Min of A – best of three algorithms) / best of three algorithms, A=IBA, GA, PSO.

Table 10. Summarized of times for confidence interval, best solutions, p-values for three algorithms (50 experiments).

I	% visit	95% Confidence Interval IA	GA	PSO	Best Solution (times) IA	GA	PSO	P-value (1)	(2)	(3)
1	50%	(278.10, 278.10)	(278.10, 278.10)	(278.10, 278.10)	278.1(50)	278.1(50)	278.1(50)	*	*	*
2	58%	(278.10, 278.10)	(277.90, 278.30)	(274.91, 285.49)	278.1(50)	278.1(48)	278.1(18)	#	**	**
3	66%	(278.10, 278.10)	(273.22, 286.58)	(288.40, 325.40)	278.1(50)	278.1(30)	289.1(01)	**	**	**
4	75%	(276.93, 279.67)	(266.50, 296.30)	(312.05, 355.75)	278.1(45)	278.1(39)	305.6(01)	**	**	**
5	83%	(275.71, 282.89)	(256.91, 315.09)	(341.96, 386.84)	278.1(25)	278.1(12)	344.6(01)	**	**	**
6	100%	(303.19, 325.41)	(288.56, 355.44)	(405.93, 462.27)	304.3(01)	309.9(01)	402.2(01)	**	**	**
7	50%	(383.30, 383.30)	(383.30, 383.30)	(383.30, 383.30)	383.3(50)	383.3(50)	383.3(50)	*	*	*
8	58%	(383.30, 383.30)	(383.30, 383.30)	(381.08, 386.92)	383.3(50)	383.3(50)	383.3(37)	*	**	**
9	66%	(383.30, 383.30)	(375.11, 394.09)	(389.49, 435.71)	383.3(50)	383.3(43)	384.6(01)	**	**	**
10	75%	(383.30, 383.30)	(377.97, 391.03)	(417.95, 460.05)	383.3(50)	383.3(39)	407.9(01)	**	**	**
11	83%	(382.42, 384.38)	(371.89, 400.51)	(457.62, 508.58)	383.3(45)	383.3(31)	446.8(01)	**	**	**
12	100%	(388.68, 418.32)	(374.85, 448.55)	(531.14, 588.06)	387.9(01)	385.8(01)	530.3(01)	**	**	**

(1)IBA vs GA, (2)IBA vs PSO, (3)GA vs PSO, *All solutions are identical, ** p-value<0.05, # p-value≥0.05.

4.2 Discussions

Table 9 and Table 10 show that:

(1) For each test instance, with an increase in the number of exhibition rooms visited, the best completion time of the groups increases in a stair-type format. For example, in Table 9, the makespan is 278.1 and 304.3 by IBA for Instance 2 and Instance 6, respectively. This finding implies that the completion time does not necessarily increase with the increase of percentage of room visited.

(2) For each instance, the standard deviation of the 50 solutions by IBA for the 12 instances is low, even reaching to zero. This finding implies that the adopted IBA is stable enough to solve the GMVRP-PC. For example, in Table 9, seven instances with zero standard deviation in solutions are shown.

(3) For each instance, the 95% confidence intervals of solutions for IBA are superior to those of GA, and PSO is worse to GA. For example, Table 10 shows that the 95% confidence intervals of Instance 6 for IBA, GA, and PSO are (303.19, 325.41), (288.56, 355.44), and (405.93, 462.27), respectively. This finding implies that IBA is more stable than GA and PSO for GMVRP-PC.

(4) For each instance, with an increase in the number of exhibition rooms visited, the CPU time of obtaining the solution is stable or slightly increases. For example, in Table 9, the CPU time of Instances 1 to 12 is between 3116.1 and 7369.0 for IBA. Similar results for GA and PSO were obtained. In addition, Table 9 shows that PSO is faster than GA, and IBA is slower than GA.

(5) For Instances 1-5 and 7-11, the best solutions by IBA and GA are all identical. However, for Instance 6, IBA (304.3) is superior to GA (309.9) and PSO (402.2).

(6) The best solutions by IBA and GA are near to the global optimum solution for test instances. For example, in Instance 11, as Room G is a must-see exhibition room for all groups and it is a dominated room (with the largest visiting time), the total visit time and entrance/exit time of all groups for Room G is (17 + 16.8 +…+ 21.4) + 1.4 (entrance) + 1.4 (exit) = 383.3, which is a lower bound of makespan for this instance. If all groups have to visit a specific room and its visiting time is larger than that of other rooms,

then the room is a dominated room. Table 9 shows that the best solution by IBA and GA is 383.3, which implies that this solution is the global optimum solution for the instance. This finding further indicates that IBA and GA are reliable for solving the considered GMVRP-PC.

(7) Table 10 summarizes the number of best solutions achieved by the three algorithms in 50 trials. For example, in Instance 5, IBA has 25 times with an objective value of 278.1, GA achieves 12 times with an objective value of 278.1, and PSO has 1 time with an objective value of 344.6. This finding implies that IBA is superior to GA and GA is superior to PSO.

To further analyze the performance of IBA, GA, and PSO algorithms for each test instance, based on the 50 trials, we used the following statistical hypothesis to test:

$$H_0: V(\mathrm{A})=V(\mathrm{B}),$$
$$H_1: V(\mathrm{A})\neq V(\mathrm{B}),$$

where V(A) and V(B) denote the average makespans by using algorithms A and B, A, B$\in${IBA, GA, PSO}, A$\neq$B. The corresponding p-values of the statistical hypothesis are summarized in Table 10, which also shows the following:

(1) Except for Instance 2 (with p-value 0.0797), IBA outperforms GA for the other instances.
(2) IBA outperforms PSO for all 12 instances.
(3) GA outperforms PSO for all 12 instances.

5 CONCLUSIONS

(1) WE have introduced a new GMVRP-PC that involves a combination of exhibition rooms and the routing of groups. The GMVRP-PC considers the various/multiple interests of groups for must-see and select-see exhibition rooms that MVRP and GMVRP ignored.
(2) We have derived the properties of the GMVRP-PC. That is, the presented GMVRP-PC generalizes OSSP, MVRP, and MT-MVRP.
(3) We have presented a novel encoding scheme to simultaneously determine the scheduling of the must-see and select-see exhibition rooms for all groups. Three algorithms, namely, IBA, GA, andPSO, have also been developed to solve the GMVRP-PC.
(4) We have reported and discussed the numerical results of a museum in Taiwan and shown the effectiveness of the proposed approaches. Among the 12 instances, IBA and GA achieve the best solutions for 11 instances, whereas PSO only 5 instances.
(5) IBA is worse to GA for one instance with 1% and GA is worse to IBA for one instance with 2%.

In the future, other artificial intelligence algorithms may be applied for solving the GMVRP-PC. Other special types of museum routing problem, for example, museum routing problem with multiple guides, may also be considered.

6 APPENDIX

THE step-by-step processes of the encoding procedure for the example in Section 3.3 are shown below.

Step 0. R=ϕ. M_1={1,2}, S_{11}={3,4}, S_{12}={5,6}, m_1=2, s_{11}=1, s_{12}=1, u_1=2; M_2={2}, S_{21}={1,3,4}, S_{22}={5,6}, m_2=1, s_{21}=2, s_{22}=1, u_2=2;M_3={2,3}, S_{31}={1,5}, S_{32}={4,6}, m_3=2, s_{31}=1, s_{32}=1, u_3=2;M_4={2,5}, S_{41}={1,4}, S_{42}={3,6}, m_4=2, s_{41}=1, s_{42}=1, u_4=2;M_5={2}, S_{51}={1,3,4}, S_{52}={5,6}, m_5=1, s_{51}=2, s_{52}=1, u_5=2; M_6={1,2}, S_{61}={3,4}, S_{62}={5,6}, m_6=2, s_{61}=1, s_{62}=1, u_6=2; M_7 ={2}, S_{71}={1,4}, S_{72}={3,5,6}, m_7=1, s_{71}=1, s_{72}=2, u_7=2; M_8={2,3}, S_{81}={1,5}, S_{82}={4,6}, m_8=2, s_{81}=1, s_{82}=1, u_8=2.

Step 1. Since each group has to visit 4 rooms, N=32(=8×4). Suppose that T=26-6-25-10-21-14-11-28-2-23-13-12-15-18-31-24-22-1-20-4-8-5-19-29-16-27-32-3-30-17-9-7 is a random permutation of 1 to 32.

Step 2. As shown in Table 5, divide T into 3×n=24 sub-groups with $T=(T_{11}\cup T_{12}\cup T_{13}) \cup\ldots \cup (T_{81}\cup T_{82} \cup T_{83})$, where, T_{11}={26,6}, T_{12}={25}, T_{13}={10}…, T_{81} ={30,17}, T_{82}={9} and T_{83}={7}.

Step 3. i=1.

(a) Since 26>6, w=order of {26,6}=(2,1). Thus the corresponding R of T_{11} in M_1(={1,2}) is R(1)=2 and R(2)=1.

(b) (i) p=1, (ii) w=the 1[st] number of T_{12}(={25}). Since {(w mod $|S_{ip}|$)+1}={(25 mod 2)+1}=2, the 2[nd] number in S_{11}(={3,4}) is 4. R(3)=4.

(i) p=2, (ii) w= the 1[st] number of T_{13}(={10}). Since {(w mod $|S_{ip}|$)+1}={(10 mod 2)+1}=1, the 1[st] number in S_{12}(={5,6}) is 5. R(4)=5.

Repeat the similar processes for i=2, 3, 4, 5, 6, 7, 8.

Step 4: The completed sequence R is shown in Table 5. Following the sequence of T from 1 to 32, we may construct the following routing sequence of rooms by using G and R:

G_{53}-G_{32}-G_{75}-G_{55}-G_{61}-G_{11}-G_{86}-G_{62}-G_{85}-G_{15}-G_{23}-G_{34}-G_{35}-G_{24}-G_{42}-G_{72}-G_{82}-G_{45}-G_{64}-G_{51}-G_{22}-G_{52}-G_{33}-G_{43}-G_{14}-G_{12}-G_{74}-G_{25}-G_{66}-G_{83}-G_{44}-G_{76}

where G_{ij} denotes that group i visits exhibition room j. This routing sequence of rooms can be further used to construct a Gantt chart for all groups. That is, in the Gantt chart of Figure 4, we assign Group 5 to visit Exhibition Room 3 first; then assign Group 3 to visit Exhibition Room 2 next, and so on.

7 ACKNOWLEDGMENT

THIS work was supported in part by Ministry of Science and Technology of Taiwan, under Contracts MOST 104-2221-E-150-022-MY2 and MOST 106-2221-E-150-039.

8 REFERENCES

I. Adiri and N. Aizikowitz, (1989). Open shop scheduling problems with dominated machines, *Naval Research Logistics Quarterly*. 36, 273–281.

J.R. Al-Enezi, M.F. Abbod and S. Alsharhan, (2010). Artificial immune systems – models, algorithms and applications, *International Journal of Research and Reviews in Applied Sciences*. 3(2), 118–131.

M.R. AlRashidi and M.E. El-Hawary, (2009). A survey of particle swarm optimization applications in electric power systems, *IEEE Transactions on Evolutionary Computation*. 13, 913–918.

M. Basu, (2011). Artificial immune system for fixed head hydrothermal power system, *Energy*. 36, 606–612.

H. Brasel, T. Tautenhahn and F. Werner, (1993). Constructive heuristic algorithms for the open shop problem, *Computing*. 51, 95–110.

B. Çaliş and S. Bulkan, (2015). A research survey: review of AI solution strategies of job shop scheduling problem, *Journal of Intelligent Manufacturing*. 26, 961–973.

S.Y. Chou and S.W. Lin, (2007). Museum visitor routing Problem with the balancing of concurrent visitors, *Complex Systems Concurrent Engineering*, book chapter, Springer-Verlag.

U. Dorndorf, E. Pesch and T. Phan-Huy, (2001). Solving the open shop scheduling problem, *Journal of Scheduling*. 4(3), 157–174.

F. Erchiqui, (2018). Application of genetic and simulated annealing algorithms for optimization of infrared heating stage in thermoforming process, *Applied Thermal Engineering*. 128, 1263–1272.

H.L. Fang, P. Ross and D. Corne, (1994). A promising hybrid GA/heuristic approach for open-shop scheduling problems, *Proc. of the 11th European Conference on Artificial Intelligence*, Amsterdam, Netherlands.

T. Fiala, (1983). An algorithm for the open-shop problem, *Mathematics of Operations Research*. 8, 100–109.

C. Gueret and C. Prins, (1998). Classical and new heuristics for the open-shop problem: a computational evaluation, *European Journal of Operational Research*. 107, 306–314.

D. Henderson, S.H. Jacobson and A.W. Johnson, (2003). The Theory and Practice of Simulated Annealing. *Handbook of Metaheuristics* (International Series in Operations Research & Management Science Vol. 57), Springer, 287-319.

Y.C. Hsieh and P.S. You, (2014). An immune evolutionary approach for the label printing problem, *International Journal of Computational Intelligence Systems*. 7, 515–523.

Y.C. Hsieh and P.S. You, (2017). An efficient encoding scheme for a new multiple-type museum visitor routing problem with must-see and select-see exhibition rooms, *International Journal of Computational Intelligence Systems*. 10, 677–689.

Y.M. Huang and J.C. Lin, (2011). A New Bee Colony Optimization Algorithm with Idle-Time-Based Filtering Scheme for Open Shop Scheduling Problems. *Expert Systems with Applications*. 38, 5438–5447.

N.K. Jerne, (1973). The immune system, *Scientific American*. 229, 52–60.

J. Kennedy and R.C. Eberhart, (1995). Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*. 1942–1948, NJ, USA.

A.V. Kononov, (2012). On the routing open shop problem with two machines on a two-vertex network, *Journal of Applied and Industrial Mathematics*. 6, 318-331.

C.F. Liaw, (1998). An iterative improvement approach for the nonpreemptive open shop scheduling problem, *European Journal of Operational Research*. 111, 509–517.

C.F. Liaw, (1999a). Applying simulated annealing to the open shop scheduling problem, *IIE Transactions*. 31, 457–465.

C.F. Liaw, (1999b). A tabu search algorithm for the open shop scheduling problem, *Computers & Operations Research*. 26, 109–126.

T.Y. Lim, (2014). Structured population genetic algorithms: a literature survey, *Artificial Intelligence Review*. 41(3), 385–399.

W.C. Lin, C.C. Wu, K. Yu, Y.H. Zhuang and S.C. Liu, (2018). On the use of genetic algorithm for solving re-entrant flowshop scheduling with sum-of-processing-times-based learning effect to minimize total tardiness, *Journal Intelligent Automation & Soft Computing*. 24, 671-681.

S.Q. Liu and H.L. Ong, (2004). Metaheuristics for the Mixed-Shop Scheduling Problem, *Asia-Pacific Journal of Operational Research*. 21, 97–115.

Z. Michalewicz, (1994). *Genetic Algorithm + Data Structures = Evolution Programs*. Springer-Verlag.

S. Mirjalili and A. Lewis, (2016). The whale optimization algorithm, *Advances in Engineering Software*. 95, 51-67.

H. Panahi and R. Tavakkoli-Moghaddam, (2011). Solving a multi objective open shop scheduling problem by a novel hybrid ant colony optimization, *Expert Systems with Applications*. 38, 2817–2822.

C. Prins, (2000). Competitive genetic algorithms for the open-shop scheduling problem, *Mathematics of Operations Research*. 52, 389–411.

T.A. Roemer, (2006). A note on the complexity of the concurrent open shop problem, *Journal of Scheduling*. 9, 389–396.

M. Srinivas and L.M. Patnaik, (1994). Genetic algorithms: a survey, *Computer*. 27, 17–26.

E. Taillard, (1993). Benchmarks for basic scheduling problems, *European Journal of Operational Research*. 64, 278–285.

S.O. Tasan and S. Tunali, (2008). A review of the current applications of genetic algorithms in assembly line balancing, *Journal of Intelligent Manufacturing*. 19, 49–69.

V.F. Yu, S.W. Lin and S.Y. Chou, (2010). The museum visitor routing problem, *Applied Mathematics and Computation.* 216, 719–729.

I.L. Weissman and M.D. Cooper, (1993) How the immune system develops, *Scientific American*. 269, 33–40.

9 DISCLOSURE STATEMENT

NO potential conflict of interest was reported by the authors.

10 NOTES ON CONTRIBUTORS

Yi-Chih Hsieh received his Ph.D. degree in Industrial Engineering from The University of Iowa, USA, and his current research interests include optimization, operations research, and applications of artificial intelligence algorithms. He is now a professor at the Department of Industrial Management, National Formosa University, Taiwan.

Peng-Sheng You received a Ph.D. degree in Management Science and Engineering from University of Tsukuba, Japan, and his current research interests include inventory management, supply chain management, and Optimization. He is now a professor at the Department of Business Administration, National Chiayi University, Taiwan.