Journal of Optimization in Industrial Engineering
Vol.14, Issue 1, Winter & Spring 2021, 10-18
DOI: 10.22094/joie.2020.511.21

Research Article

Solving a Bi-objective Model for Hotel Revenue Management Considering Customer Choice Behavior Using Meta-heuristic Algorithms

Surur Yaghobi Harzandi, Amir Abbas Najafi

Department of Industrial Engineering, Science & Research Branch, Islamic Azad University, Qazvin Branch, Iran
Faculty of Industrial Engineering, K.N. Toosi University of Technology, Tehran, Iran

Received 28 March 2020; Revised 09 May 2020; Accepted 12 May 2020

Abstract

The problem of maximizing the benefit from a specified number of a particular product with respect to the behavior of customer choices is regarded as revenue management. This managerial technique was first adopted by the airline industries before being widely used by many others such as hotel industries. The scope of this research is mainly focused on hotel revenue management, regarding which a bi-objective model is proposed. The suggested method aims at increasing the revenue of hotels by assigning the same rooms to different customers. Maximization of hotel revenue is a network management problem aiming to manage several resources simultaneously. Accordingly, a model is proposed in this paper based on the customer choice behavior in which the customers are divided into two groups of business and leisure. Customers of the business group prefer products with full price, whereas products with discounts are most desirable for leisure customers. The model consists of two objectives, the first one of which maximizes the means of revenue, and the second one minimizes the dispersion of revenue. Since the problem under consideration is Non-deterministic Polynomial-time hard (NP-hard), two meta-heuristic algorithms of Non-dominated Sorting Genetic Algorithm-II (NSGA-II) and Multiple Objective Particle Swarm Optimization (MOPSO) are proposed to solve the problem. Moreover, the tuned algorithms are compared via the statistical analysis method. The results show that the NSGA-II is more efficient in comparison with MOPSO.

Keywords**:** Hotel Revenue management, Bi-objective model, Meta-heuristic algorithms, Customer choices.

1. Introduction

Revenue management, hereinafter regarded as RM, is a managerial technique which makes use of a number of strategies to manage the allocation of capacity to different fare classes over time in order that revenue would be maximized for different industries (Philips, 2005). This method, which was first suggested by Littelewood in 1987, was initially employed for time and fare control in Civil Aeronautics Board (CAB). It was, then, the airline industry which pioneered in revenue management in 1980, following the success of which the technique was utilized by many other industries including hotels, rental cars, freight transportation, and cruise line (Philips, 2005).
As shown by many scholars, hotel industries can potentially make use of the managerial techniques of airline industries due to their similar characteristics, making revenue management a significant subject of interest for hotel managers. The following characteristics are among the common characteristics of airline industries and hotels: a) products of both industries (hotel rooms and airline seats) are mortal and cannot be stocked; b) the volume of the products are stabilized; c) reserving in advance is permissible in both (Lai and Ng, 2005).
Goldman et al. studied the decision-making rules of multiple-day reservation in hotels based on stochastic and definitive mathematical programming methods in a paper in 2002, where they considered a flexible reserving schedule rather than a fixed schedule. A network optimization model in a stochastic environment was proposed by Lai and Ng in 2005 for hotel revenue management. Their proposed optimization approach presents a stochastic programming in order to obtain randomness of pathless demands. More recently, some researchers have been interested in deadly modeling of customer behavior in revenue management problems (Van Ryzin and Vulcano, 2008a). A single-leg model of revenue management with distinct elected model of order was introduced by Taluri and Van Ryzin (2004a). Gallego et al. developed elastic products in networks (2004). Moreover, a dynamic programming was introduced by Liue and Ryzin in 2008 based on decay heuristic. Based on dynamic planning,

*Corresponding author Email address: aanajafi@kntu.ac.ir

Adelman presented a model for time-dependent pricing of base prices. Since the problem was complicated due to large dimensions, a column generation method was used to solve the problem. Tong and Topaloglu further developed Adelman's network revenue management for an airline network in 2012 by making use of a linear model in which the number of constraints exponentially increases with increase in the number of flight legs.

Several exact methods have been proposed to solve this problem, most of which are considerably difficult and time consuming. Although the results of these exact methods are close to the optimal solution, they are not desirable due to their unreasonable computation time and complexity. In order to address the difficulties of the exact solution methods, other techniques such as simulation were proposed by the scholars which are more convenient but less accurate. More recently, meta-heuristic methods have been used as alternative solution techniques. Gosavi (2002) has made use of artificial intelligence methods for optimizing revenue in airline industries. The column generation algorithm has been used by Bront et al. in a research work in 2009 for solving choice-based linear programming models, where the greedy heuristic algorithm has been used to solve each column. It was proved by Etebari et al. in 2011 that this problem is an NP-hard, for which they applied the genetic algorithm to solve each column in a choice-based manner. The proposed model has two objectives. The first objective maximizes expected revenue (Liue and Ryzin, 2008) and the second one minimizes dispersion of revenue. In other words, a second objective function is proposed which minimizes the deviation from the mean value while maximizing the income via the first objective function. This approach has two real plus points; firstly, the determination rate changes in both objective functions, and it is dependent on the selected scenarios. Secondly, the aversion risk factor of revenue resulted from random demands under different scenarios is reduced which was one of the major difficulties of the problems of the proposed model of Liue and Ryzin (2008).

2. Customer Stochastic Behavior Programming

Renting the same room to different customers at different prices, which is one of the contributors that increase the revenue of the hotels, is one of the major challenges of hotel managers. In order to accomplish this, the product (the room) should be reserved in advance, especially when the supply exceeds the demand (Liu and Ryzin, 2008). This paper proposes a choice-based revenue management scheme in which the costumer priorities are taken into consideration such that in a multi-nominal logit model, each costumer can be assigned to more than one sector, and the booking horizon is divided into several intervals (Etebari at al., 2011). It is also assumed in the model that the hotel has only one type of room; however, the unit rate of the same room is different based on different booking periods. Moreover, it is worth mentioning that each reservation might cover several days in this model (Liu and Ryzin, 2008).

The major parameters of the proposed model are as follow:

- C is the room capacity of the hotel
- T represents the problem's time horizon in which booking in advance takes place
- R_{m} shows the revenue resulting from renting a room on one day, with regards to capacity price of m.
- α represents the percentage of the allocated rooms to costumers of business type.
- β shows percentage of leisure customers who have rented a room at full price due to lack of enough rooms.
- $D^1_{i,j,s}$ refers to the amount of room demand by business customers in scenario S, who intend to check-in on day i and check-out on day j.
- $D^2_{i,j,s}$ is room demand from leisure customers in scenario S, who would like to check-in on day i and check-out on day j.

Parameter $D_{i,j,s}$ is uncertain in most real situations. In order to address this problem, this uncertain parameter can be replaced by the expected value E $(D_{i,j})$. However, this sometimes does not result in acceptable answers. Accordingly, assuming that a decision-maker encounters a set of scenarios as s ∈ Ω= {1, s} with unknown parameters, and the corresponding probability for each scenario is P_s such that $P_s \geq 0$ and $\sum_{s=1}^{S} P_s = 1$, the following variables are employed in this model:

- $x_{i,i}$ is the number of business customers accepted for check-in on day *i* and check-out on day *j* (Integer decision variables)
- $y_{i,j}$ is the number of leisure customers accepted for check-in one day *i* and check-out on day *j* (integer decision variables)
- $y^2_{i,j}$ is the number of leisure customers, accepted to use second grade rooms for check-in on day *i* and check-out on day *j* (Integer decision variables)
- $y^1_{i,j}$ is the number of leisure customers, accepted to use business class rooms, because second-grade rooms did not have enough capacity for check-in on day *i* and check-out on day *j* (Integer decision variables)

Where 0< $i < j \leq T, i = \{0, 1, 2, \dots, T-1\}$, is the check-in time and j= (1, 2, 3, T) is the check-out time, and z_p is the binary decision variable.

$$z_p = \begin{cases} 1 & \text{if hotel rooms are full,} \\ 0 & \text{otherwise} \end{cases}$$

The first objective function is defined as the following:

$$\max\ Z = \sum_{s=1}^{S} P_s \sum_{i=0}^{T-1} \sum_{j=1}^{T-i} j(R_{1,s} x_{i,j} + R_1 y_{i,j}^1 + R_{2,s} y_{i,j}^2) - \sum_{s=1}^{S} P_s \sum_{i=0}^{T-1} \sum_{j=1}^{T-i} \left(w1_{i,j} \max\{0, x_{i,j} - D1_{i,j,s}\} + w2_{i,j} \max\{0, y_{i,j} - D2_{i,j,s}\}\right) \quad (1)$$

The first term of the objective function defined by equation (1) is the expected revenue of the hotel, and the second term refers to the semi-variance of the revenue. Semi-variance is used in order to measure the robustness of the model, and parameters $w_{i,j}$ are penalty factors in case the constraints are violated.

The second objective function is defined as follows:

$$Min\ Q = \sum_{\boldsymbol{s=1}}^{\boldsymbol{S}} \boldsymbol{P_s} \left(max \left\{ 0, \sum_{s=1}^{S} P_s \sum_{i=0}^{T-1} \sum_{j=1}^{T-i} j\left(R_{1,s} x_{i,j} + R_1 y_{i,j}^1 + R_{2,s} y_{i,j}^2\right) - \sum_{i=0}^{T-1} \sum_{j=1}^{T-i} j(R_{1,s} x_{i,j} + R_1 y_{i,j}^1 + R_{2,s} y_{i,j}^2 \right\} \right)^{\boldsymbol{2}} \quad (2)$$

The second objective minimizes the dispersion of different scenarios.

$$\sum_{i=0}^{p} \sum_{j=p-i+1}^{T-i} (x_{i,j} + y_{i,j}^1) \le \alpha C\ ; \forall p = 0, \dots, T-1 \quad (3)$$

Constraint (3) is defined to prevent the sum of business customers and commercial customers who intend to use first class rooms from exceeding the maximum capacity allocated to commercial customers.

$$\sum_{i=0}^{p} \sum_{j=p-i+1}^{T-i} y_{i,j}^2 \le (1-\alpha)\mathrm{C}; \forall \mathrm{p} = 0, \dots, \mathrm{T} \quad (4)$$

Constraint (4) prevents the model from accepting more leisure customers than the available capacity of the hotel.

$$\sum_{i=0}^{p} \sum_{j=p-i+1}^{T-i} y_{i,j} \ge (1-\alpha)\mathrm{C}\ z_p; \forall \mathrm{p} = 0, \dots, \mathrm{T}-1 \quad (5)$$

$$\sum_{i=0}^{p} \sum_{j=p-i+1}^{T-i} y_{i,j}^1 \le \alpha \mathrm{C}\ z_p; \forall \mathrm{p} = 0, \dots, \mathrm{T}-1 \quad (6)$$

$$0 \le y_{i,j}^1 \le \beta\left(max\{D2_{i,j,s}\} - y_{i,j}^2\right), int, \forall 0 \le i \le T-1, \forall j = 1, \dots, T \quad (7)$$

$$y_{i,j} = y_{i,j}^1 + y_{i,j}^2; \forall 0 \le i \le T-1, \forall j = 0, \dots, T \quad (8)$$

Constraint (5) states that if z_p equals to one, then leisure customers will exceed the capacity of the hotel for this type of customers, in which case constraint (7) allows the model to allocate the excessive leisure customers to the business class rooms. Constraint (6), on the other hand, makes sure that the capacity of the hotel for business customers does not exceed its capacity for leisure customers.

$$0 \le x_{i,j} \le D1_{i,j,s}, int, \forall 0 \le i \le T-1, \forall j = 1 \dots \dots T \quad s = 1, \dots, S \quad (9)$$

$$0 \le y_{i,j} \le D2_{i,j,s} \forall 0 \le i \le T-1, \forall j = 1 \dots \dots T \quad s = 1, \dots, S \quad (10)$$

$$z_p \in \{0,1\}, \forall 0 \le p \le T-1 \quad (11)$$

Equations (8) to (10) express limitations on variables.

According to the previous discussions, this problem is NP-hard (Eteberi et al., 2011). Hence, two meta-heuristic algorithms are proposed in this paper to address the problem.

3. Designing NSGA-II and MOPSO for the proposed model

Owing to the fact that the proposed model of this paper is bi-objective, a multi-objective optimization method must be applied. Two algorithms are proposed in this respect to solve the model. One of them is the NSGA-II optimization algorithm which is a popular non-domination based optimization algorithm for multi-objective optimization. This highly efficient algorithm is, however, often criticized for its computational elaboration (Srinivas and Deb, 1994). The second proposed algorithm is the multi-objective version of the particle swarm optimization algorithm (MOPSO) which elects the best local leader (the global best particle), and makes a set of Pareto-optimal solutions for every particle of the population. This algorithm is known to perform effectively in terms of both convergence and solution diversity (Mostaghim and Teich, 2003).

3.1. Common characteristic of these two algorithms

A population is exploited for both proposed algorithms (NSGA-II and MOPSO) in order to obtain an appropriate solution, in which the population matrix is common between the two algorithms.

3.2. Solution representation

This subsection is dedicated to defining a common representation approach between the two proposed algorithms. Accordingly, an initial solution must be defined. Assuming that the considered horizon consists of *m* days, a $2m \times m$ matrix can be defined, in which the first *m* rows (the first 5 rows for the given example of the next section) represent check-in days for business customers, and the second *m* rows show the check-in days of the leisure customers. The columns represent the length of stay in the hotel for customers in terms of days. As an example, Figure 1 illustrates a solution in which g (i, j) =4 indicates that the demands of the business customers that have checked-in on day i and checked-out on day j, considering g $1 \leq i \leq m$, is equal to four. It should be noted that if i is between $m+1$ and $2m$, it is the demands of the leisure customers that equals to four.

		The duration of the customer's stay				
Check-in day	Business customers	4	2	4	4	1
		2	4	5	6	0
		3	3	2	0	0
		2	2	0	0	0
		1	0	0	0	0
	Leisure customers	3	3	5	3	7
		2	3	4	0	0
		4	4	0	0	0
		2	3	0	0	0
		2	0	0	0	0

Fig 1. A sample chromosome

3.2.2. Objective function

The objectives of the present paper are defined according to the solution matrix. Accordingly, each solution in the proposed algorithm has a fitness value. However, some solutions might be infeasible due to some constraints, in which case the solution will be placed in the constraints of the model, and the algorithm employs the following approach to make such solutions feasible. Each infeasible solution will be removed, and corresponding values of element g (1,1) in the solution matrix will be selected randomly. The other elements of the matrix will continue to be randomly selected, and the process will continue until the solution becomes feasible.

3.2. Non-dominated sorting genetic algorithm (NSGA-II)

First introduced by Srinivas and Deb in 1994, NSGA is used for solving non-convex non-smooth single and multi-objective optimization problems. Given that the performance of NSGA was strongly dependent on some parameters such as sharing and fitness, Deb et al. proposed a modified version of the algorithm known as NSGA-II which can be summarized in the following steps (2000):

Step (1): the primitive population is created,

Step (2): the value of the fitness function is measured,

Step (3): a rank (fitness) is determined for each solution in each front and the non-dominates are sorted in the fronts,

Step (4): the crowding distance is estimated which is a control parameter that extents how close an individual is to its neighbors,

Step (5): the parents are selected from the population,

Step (6): new offspring are introduced by crossover and mutation operations.

3.2.1. Definitions of the operators

Crossover and mutation operators are utilized in this research. In order to select chromosomes via the crossover operator, three methods are applied, all of which will be combined with random probabilities. Besides, mutation, which is a uniform method, is adopted as a selection approach.

Crossover operator type I

The columns or rows are selected randomly from each parent via this method, and two offspring are created by switching the rows or the columns of the parent.

Crossover operator type II

In this crossover type, which is a uniform crossover, the fixed mixing ratio is used between two parents. To build upon this point a little more, if the mentioned ratio is, for example, 0.7, it means that the offspring inherits approximately 0.7 of its genes from its first parent and 0.3 from its second parent.

Mutation operator

This operation is performed in two ways. The first mutation method, called SWAP, randomly selects two elements of the matrix and switches their places with each other to produce a new offspring. The second mutation method, called mixed mutation. subtracts the lower bound of the chromosomes from the upper bound followed by multiplying the result by a less than unit number; then it sums the result with one and introduces the greatest integer of the resulting matrix as a new offspring.

3.3. Multi-objective particle swarm optimization (MOPSO)

A powerful optimization tool introduced and commonly used by many scholars is the particle swarm optimization (PSO), in which the solution for each generation is the optimum solution of the previous generation. The particles in this algorithm are obtained via equation (12):

$$v_{ij}[t+1] = wv_{ij}[t] + c_1 r_1 \left(Pbest_{ij}[t] - x_{ij}[t]\right) + c_2 r_1 \left(Gbest_{ij}[t] - x_{ij}[t]\right) \quad (12)$$

$$x_{ij}[t+1] = x_{ij}[t] + v_{ij} \quad (13)$$

Where i is the particle index, j is the dimension index, t is the iteration counter, w is the inertia factor, c_1 is the personal learning coefficient, c_2 is the global learning coefficient, r_1 and r_2 are the random numbers uniformly distributed in [0, 1], $v_{i,j}$ is the j-th velocity of particle i, $p_{i,j}$ is the j-th personal for particle i, and g_j shows the j-th element of the best global position.

However, given that the problem under consideration has more than one objective function, the multi-objective version of PSO, known as MOPSO, must be employed. In this version of the algorithm, the non-domination population will be saved in a storage, and Valency's limited number of repositories indicate the Pareto front. The process of MOPSO is summarized as follows:

Step (1): the primitive population is procreated,

Step (2): the non-dominated member is elected and saved in the storage,

Step (3): the solution space is categorized,

Step (4): the leaders are elected in the solution space,

Step (5): the particles are updated by the leader particle, followed by selecting the non-dominated member and adding it to the repository. After a particle finishes the updating process, its best status must be updated too. The update is performed as follows (Mostaghim & Teich, 2003):

$$P_i[t+1] = \begin{cases} P_i[t] & \text{if } p_i[t] \text{ dominates } x_i[t], \\ X_i[t] & \text{if } x_i[t] \text{ dominates } p_i[t], \\ \text{Randomly} & \text{otherwise} \end{cases}$$

The processes of both algorithms continue until the stop criterion is met.

Stop criterion

The adaptive condition is employed in this research as the stop criterion. It should be noted that the maximum number of iterations (generation) is used in this research without improvement.

4. Computational Results

The algorithm is implemented in this section, and its efficacy is compared. 40 examples have been solved in horizons of 5, 10, and 15 days. Given that customer's behavior is assumed to be choice-based in this model, the customers' entrance pattern follows the Poisson's distribution which assumes that one customer enters per hour. Customers are divided into three sections which are shown in Table 1.

Table 1
Example of horizon

Horizon (day)	The Probability of a customer entering from each section	
5	P1	0.4
	P2	0.5
	P3	0.1
10	P1	0.4
	P2	0.5
	P3	0.1
15	P1	0.4
	P2	0.5
	P3	0.1

The probability of length of stay is described in Table 2, after the determination of which the check-in day must be also determined. Assuming the length of stay as J, the probability of selecting the check-in day is $\frac{1}{t-j+1}$, which presumes that two types of customers are available with two types of price rates.

The first price rate type is applied on business customers. As it was mentioned before, in case the capacity of leisure customers is filled, leisure prices will be applied during their stay in business class suits.

Before solving the examples, the parameters of the algorithms must be tuned.

4.1. Tuning the parameters

Tuning of the parameters directly affects the quality of the solutions of an algorithm. Many techniques exist for tuning the parameters (Salimi and Najafi, 2018; Shahsavar et al., 2011; Amiri et al, 2008). In this research, to determine the best combination of parameters the Taguchi method is applied. This method was recently employed by many researches for algorithm parameters tuning (Rezaei et al, 2020; Arjmand et al, 2020). Taguchi is a statistical method in which some levels of parameters are considered for each algorithm followed by designing several experiments so that optimum parameters would be found by repeating the experiments several times via different parameter levels.

In the present work, four parameters and three parameter levels are considered for NSGA-II. The parameters include population size, crossover probability (p_c), mutation probability (p_m) and maximum number of generations. The results of the experiments obtained via Minitab 17 software are presented in Table 3, where the presented levels are the optimum levels of the parameters.

Furthermore, six parameters including population, repository, personal best (p), inertia factor (w), maximum number of generations, and global best position, and three parameter levels are taken into consideration for MOPSO algorithm. The optimum levels of these parameters are presented in Table 4.

Table 2
The probability of the duration of

duration of stay	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Probability	0.012	0.01	0.03	0.04	0.1	0.03	0.04	0.2	0.1	0.02	0.01	0.01	0.02	0.03	0.1

Table 3
Values of NSGA-II

Parameter	value
Population size	60
Maximum generation	150
Crossover probability(pc)	0.95
Mutation generation(pm)	0.2

Table 4
Values of MOPSO parameters

Parameter	value
Population	40
Repository	30
Personal best(p)	1.5
Inertia factor(w)	0.3
maximum generation	150
global best position(g)	1.5

Figure 2 illustrates an example of a 10-day horizon of running NSGA-II algorithm

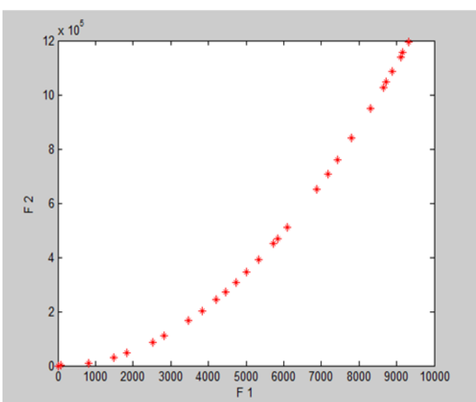

Fig 2. An example of running NSGA-II

And, Figure 3 shows an example of a 10-day horizon of running MOPSO algorithm.
An example of a 10-day horizon ran by MOPSO is shown in Figure 3. F1 and F2 are respectively the first and the second objectives in Figures 2 and 3. As it can be seen from these figures, F1 and F2 increase simultaneously. Hence, it can be concluded that a conflict exists between the two objectives. It should be reminded that this is because the first objective is maximizing the expected revenue, and the second objective is minimizing the dispersion of revenue.

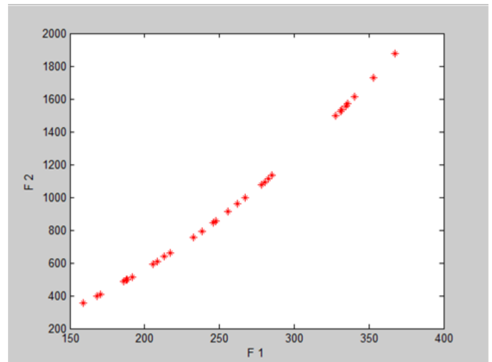

Fig 3. An example of running MOPSO

4.2. Performance measures

Given that the proposed model of this paper is a bi-objective model, evolutionary algorithms are used, the results of which are compared using a statistical analysis. In order to perform this comparison, convergence and dispersion criteria are applied. Convergence criterion includes a number of Pareto optimal solutions and the mean ideal distance (MID), and the dispersion criterion consists of spacing (S) and maximum spread (D). Time is yet another criterion for assessing the qualities of the algorithms.

4.2.1. Number of Pareto solutions

This index refers to the number of Pareto optimal solutions (Zitler & Deb, 2000).

4.2.2. Spacing (S)

Introduced by Schott in 1995, this index is used for measuring the extent of spread among the obtained solutions which is formulated by the following equation:

$$S = \sqrt{\frac{1}{|n|}\sum_{i=1}^{n}(d_i - \bar{d})^2} \qquad (14)$$

$$d_i = \min_{k\epsilon n \wedge k\neq 1} \sum_{m=1}^{2} \left| f_m^i - f_m^k \right|$$

The above-mentioned equation indicates that the efficiency of the algorithm increases with decrease in this index.

4.2.3. Maximum spread (D)

This criterion, formulated by the following equation, is employed to calculate the mean length of the hyper box (Zitler, 1999). The higher the amount of the metric, the better the efficacy of the algorithm.

$$D = \sqrt{\sum_{m=1}^{M} \left(\max_{1<i<N} f_m^i - \min_{1<i<N} f_m^i \right)^2} \qquad (15)$$

4.2.4. Mean ideal distance (MID)

Mean ideal distance is a convergence criterion that measures the distance between the Pareto solution and the ideal point (0,0). This index is formulated as below:

$$MID = \frac{\sum_{i=1}^{n} c_i}{n} \qquad (16)$$

Where C_i is the distance between the Pareto solution and (0, 0) (Zitler & Thiele, 1998).

4.3. Results

The two proposed algorithms, NSGA-II and MOPSO, will be compared in this subsection, and four indices presented in section 4.2 will be calculated using 40 examples for each algorithm.

The results of the statistical analysis of NSGA-II and MOPSO are presented in Table 5.

The results are then statistically evaluated via the paired sample t-test, which performs a parametric hypothesis test to assess the qualities of the two population means. The results of the t-test for two populations are given in Table 6. According to Table 5, it is clear that the results of NSGA-II algorithm are far more efficient than those of MOPSO in three criteria including mean ideal distance (MID), spacing (S) and maximum spread. However, MOPSO outperforms NSGA-II in terms of computation. On the other hand, the number of Pareto archive solutions is equal in both algorithms. The differences of the two algorithms are shown in Figure 4.

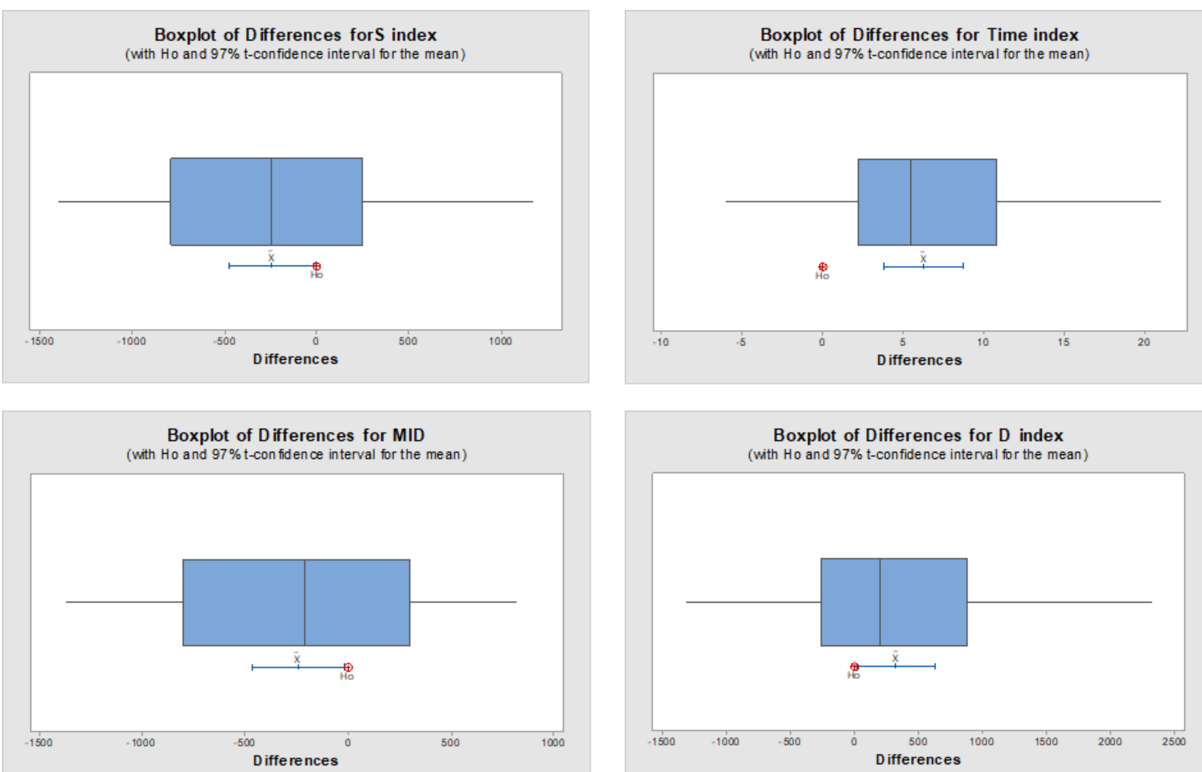

Fig 4. Results of the algorithms

Table 5
Results of running NSGA-II & MOPSO algorithms

problem	NSGA-II					problem	MOPSO				
	NPS	S	MID	T	D		NPS	S	MID	T	D
1	60	990	1124	30	1212	1	30	600	795	16	598
2	60	455	890	36	1323	2	30	959	954	24	1252
3	60	99	580	22	702	3	30	1500	1957	25	1020
4	60	453	656	20	1591	4	30	155	467	17	187
5	60	450	465	19	1324	5	30	1300	1757	21	1595
6	60	352	806	21	566	6	30	450	670	22	1876
7	60	556	382	25	1343	7	30	1005	1305	21	1120
8	60	682	664	35	1121	8	30	800	900	18	877
9	60	674	655	39	1290	9	30	1666	1657	18	1850
10	60	1222	497	32	1840	10	30	145	476	18	195
11	60	97	665	36	203	11	30	177	356	15	345
12	60	145	110	33	148	12	30	1150	1398	25	1023
13	60	756	452	20	203	13	30	1259	1455	16	1423
14	60	650	1655	21	1123	14	30	1450	1767	26	1302
15	60	550	872	23	1250	15	30	1858	1686	26	1201
16	60	541	312	24	1020	16	30	1298	1367	27	1450
17	60	405	995	21	800	17	30	153	585	15	188
18	60	590	349	23	1609	18	30	854	954	16	994
19	60	400	976	20	851	19	30	598	699	16	810
20	60	1232	501	27	2302	20	30	58	115	17	167
21	60	345	579	26	1322	21	30	200	1500	25	1145
22	60	1005	1235	30	2840	22	30	1350	1498	25	1020
23	60	1275	1166	29	2545	23	30	156	459	18	225
24	60	550	1216	26	201	24	30	200	398	16	131
25	60	440	1344	25	1087	25	30	1201	1698	21	1305
26	60	499	1532	26	1034	26	30	1568	1995	23	1444
27	60	485	2150	27	1255	27	30	1305	1450	18	142
28	60	890	1383	26	454	28	30	550	698	19	689
29	60	452	490	35	590	29	30	935	1050	19	810
30	60	1023	1320	33	2454	30	30	890	1198	19	652
31	60	1230	1654	33	2052	31	30	995	1298	18	753
32	60	890	845	27	1204	32	30	142	398	18	166
33	60	330	450	21	1251	33	30	158	267	16	99
34	60	851	695	25	1050	34	30	890	1250	19	450
35	60	498	389	25	1301	35	30	440	1135	19	403
36	60	375	295	25	1023	36	30	598	567	20	530
37	60	435	495	20	525	37	30	180	677	18	195
38	60	299	394	20	685	38	30	1158	352	26	1303
39	60	399	276	21	689	39	30	1088	1455	24	1222
40	60	455	597	22	955	40	30	990	1167	18	188
AVE	60	601	803	26	1158		30	811	1046	20	809

Table 6
Results of the statistical analysis

Efficiency index	Tasting statistics	P_value
T	5.77	0
MID	-2.45	0.019
S	-2.32	0.025
D	2.39	0.022

5. Conclusions and Future Recommendations

The problem of revenue management was investigated in this research considering customer choice behavior. The customers were divided into two groups of business and leisure customers in the proposed model of this paper. Customer preferences are chosen from different price levels. The products (rooms) were presented in two price levels; leisure customers reserve a room at entire price.

If 2^{nd} level price products are not available in recreational customers' first preference, products of the first level price will be demanded in their secondary preference list. The problem has been dealt by a bi-objective function, in which the first objective function increases the mathematical expectation of the scenarios, and the one-way dispersion of each scenario from the average value is obtained in the second objective function.

The model has been solved via two meta-heuristic algorithms, namely NSGA-II and MOPSO. The results indicate that considering increasing revenues in different scenarios in the second objective function will reduce the dispersions. Moreover, the results of the statistical analysis show that NSGA-II has been more efficient than MOPSO.

Several future recommendations are also proposed as the following. Firstly, canceled reservations can be taken into consideration. In this way, additional reserves would be considered for softening the hotel revenue. Secondly, different periods can be considered in future research work to make the problem more realistic.

References

Adelman, D. (2007). Dynamic bid-prices in revenue management, Operations Research, 55(1), 647-661.

Amiri, M., Najafi, A.A., Gheshlaghi, K. (2008). Response surface methodology and genetic algorithm in optimization of cement clinkering process, Journal of Applied Sciences, 8(15), 2732-2738.

Arjmand, M., Najafi, A.A., Ebrahimzadeh, M. (2020). Evolutionary algorithms for multi-objective stochastic resource availability cost problem, OPSEARCH, 57(3), 935-985

Bront, J.J.M., Diaz, I.M., Bront, J .M., Vulcano,G. (2009). A column generation algorithm for the choice-based linear programming model for network revenue management .Operations researches, 57(3), 769-784.

Coello Coello, C.A., Lechuga, M.S. (2002). MOPSO:A propsal for multiple objective particle swarm optimization. Proceedings of the 2002 Congress on Evolutionary Computation, IEEE Press, 1051-1056.

Etebari, F., Aaghaie, A., Khoshalhan, F. (2011). A genetic algorithm for choice –based network revenue management, Iranian Journal of Operations Research,3(1), 89-103.

Gallego, G., Iyengar,G., Phillips, F., Dubey, A. (2004). Managing flexible products on a network, Technical report CORC TR-2004-01, Department of industrial Engineering and operations research, Columbia University.

Goldman, G., Iyengar, G., Phillips, F., Dubey, A. (2002). Models and techniques for hotel revenue management using a rolling horizon, Journal of Revenue and Pricing Management, 1(3), 207-219.

Gosavi, A. (2002). The effect of noise on artificial intelligence and meta-heuristic Techniques, Intelligence Engineering systems through artificial neural network, 12, 981-988.

Kunnumkal, S., Topaloglu, H. (2010). Computing time – dependent bid prices in network revenue management problems, Transportation Science, 44(1), 38-62.

Lai, K.K., Ng, W. (2005). A Stochastic approach to hotel revenue optimization, Computers & Operations Research, 32, 1059-1072.

Liu, Q., Van Ryzin, G. (2008). On the choice-based linear programming model for network revenue management, Manufacturing and Service Operation Management, 10(2), 288-310.

Mostaghim, S., Teich, J. (2003). Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO). Proceedings of the 2003 IEEE Swarm Intelligence Symposium. IEEE Press, 26-33.

Phillips. R. (2005). Pricing and revenue optimization, Stanford University Press.

Ray, W.S. (1960). An introduction to experimental design. Macmillan, New York.

Rezaei, F., Najafi, A.A., Ramezanian, R. (2020). Mean-conditional value at risk model for the stochastic project scheduling problem, Computers and Industrial Engineering, 142(2020), 106356.

Salimi, M., Najafi, A.A. (2018). Modeling and solution procedure for a preemptive multi-objective multi-mode project scheduling model in resource investment problems, Journal of Optimization in Industrial Engineering, 11(1), 169-183

Schott, J.R. (1995). Fault tolerant design using single and multicriteria genetic algorithm optimization, Master’s Thesis, Department of Aeronautics and Astronautics, Institute of Technology, Massachusetts.

Shahsavar M., Najafi A.A. and Niaki, S. T. A. (2011). Statistical Design of Genetic Algorithms for Combinatorial Optimization Problems", Mathematical Problems in Engineering, 2011, 2-17.

Srinivas, N., Deb, K. (1994). Multi-objective optimization using nondominated sorting in genetic algorithm, Evolutionary computation, 2(3), 221-248.

Talluri, K.Y., Van Ryzin, G.J. (2004). Revenue management under a general discrete choice model of consumer behavior, Management Science, 50, 15-33.

Tong, C., Topalogu, H., (2012). On the approximate linear programming approach for network revenue management problem. INFORMS Journal on Computing, 26(1), 1-18.

Van Ryzin .J., Vulcano, G. (2008). Computing virtual nesting controls for network revenue management under customer choice behavior. Manufacturing Service Operations Management, 10(3), 448-467.

Zitzler, E., Deb, K., Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results, Evolutionary Computation, 8(2), 173-195.

Zitzler, E., Thiele, L. (1998). Multiobjective optimization using evolutionary algorithms - A comparative case study, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 292-301.

Zitzler, E., (1999). Evoutionary algorithms for multiobjective optimization: Methods and applications, A dissertation submitted to the Swiss Federal Institute of Technology, Zurich, Swiss.

Yaghobi Harzandi, S., Najafi, A. (2021). Solving Bi-Objective Model Of Hotel Revenue Management Considering Customer Choice Behavior Using Meta-Heuristic Algorithms. *Journal of Optimization in Industrial Engineering*, 14(1), 10-18.

http://www.qjie.ir/article_676308.html
DOI: 10.22094/joie.2020.511.21