A Scatter Search Algorithm for Multi-Criteria Inventory Classification with Maximize the Satisfaction Level

Ilkay Saracoglu (ilkays@sbbdanismanlik.com)
İstanbul Atlas Üniversitesi: Istanbul Atlas Universitesi https://orcid.org/0000-0003-3338-4912

Research Article

Keywords: Multiple criteria analysis, ABC inventory classification, Scatter Search Algorithm, Satisfaction level

Posted Date: August 24th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-490598/v1

Version of Record: A version of this preprint was published at Soft Computing on July 4th, 2022. See the published version at https://doi.org/10.1007/s00500-022-07227-0.

A Scatter Search Algorithm for Multi-Criteria Inventory Classification with Maximize the Satisfaction Level

Ilkay Saracoglu

Faculty of Engineering and Natural Sciences, Istanbul Atlas University, 34403 Istanbul, Turkey
Email: ilkays@sbbdanismanlik.com, Phone: 00902123660201

Abstract

Inventory management requires thousands or millions of individual transactions each year. Classification of the items influences the results of inventory management. Traditionally, this is usually classified with considering an annual dollar usage criterion but maybe other criterias such as lead time, criticality, perishability, inventory cost, and demand type can be affected on that classification. The objective of this study is to determine the multi-criteria inventory classification (MCIC) of the inventory items to minimize the total inventory cost and also dissimilarity of classes. Because of the two objectives is considered to solve with together, the maximization of satisfaction level is described to solve the multi-objective problem. This study introduces a Mixed Integer Nonlinear Programming (MINLP) model of the MCIC problem by giving two objectives. A Scatter Search Algorithm (SSA) is used to solve the MINLP model for obtaining high-quality solutions within reasonable computation times. Finally, we illustrate an example and compare our results with other studies in previous literature.

Keywords: Multiple criteria analysis, ABC inventory classification, Scatter Search Algorithm, Satisfaction level

1.Introduction

Inventory managers have to focus on important topics by avoiding unnecessary details in order to perform the inventory related tasks efficiently. Providing a more efficient inventory management can be possible through classifying the inventories and deciding which inventory will be counted more frequently and needs close follow-up. For a company beginning inventory management studies, the first procedure is to classify items with ABC analysis. Among the aims of classification of items, there is an important place for responses to questions like how many items should be ordered according to class, how much needs to be held as stock in stores, what should the safe stock level be, how often do items need to be counted, and how often items need to be checked. ABC analysis is important to calculate the days of supply, effective in determining purchasing requirements in MRP studies with ERP programs which have begun to be used by the majority of companies (e.g., SAP, etc.). Most inventory situations involve controlling many different kinds of items, with each having a different impact on cost. The physical characteristics of items can be a major concern because of short shelf life, special storage, space, and materials handling. ABC analysis is a tool that management uses to categorize materials and components into workable classes. Items that are classified in terms of total usage value are ordered from the most valuable to the least valuable. The small A item class contains the items that have the most significant impact on the inventory investment. In contrast, the large C item class consists of items that usually have little consequence on the investment. In the ABC analysis, the A item class is where 80 percent of the investment is concentrated in 20 percent of the different inventory items. The A items should receive the most attention from management. Management should frequently review the inventory function of this class. The B and C item classes represent 20 percent of the investment but 80 percent of the inventory items. Management decides which items go into the A, B, and C classes. After adding any special inventory categories, management should have a good profile of the inventory investment.

Beginning with traditional ABC analysis, the MCIC problem continues to be attempted using various methods to date. The traditional ABC classification had weighted criteria applied with the Analytic Hierarchic Process (AHP) by Flores and Clay Whybark (1986) and items were listed according to the score values obtained as they began the first multicriteria classification studies. In their study, inventory items were classified with annual dollar usage (ADU), average unit cost (ACU), criticality (CF) and lead time (LT), and they determined the weights of the criteria valus by using AHP method. The score values for each item calculated with these weights were listed from largest to smallest and items were classified according to classic ABC classification percentages. Classification studies were again developed using the AHP method (Partovi & Burton, 1993; Partovi & Hopton, 1993).

A similar study by Ramanathan (2006) and this model were later used as the R-model in the literature. (Ramanathan, 2006) used a weighted linear optimization model for classification. It was the study first attempting to solve the multicriteria classification problem with a mathematical model after Flores, Olson, and Dorai (1992). Ng (2007) (called hereafter Ng-model) proposed a linear optimization model similar to the R-model. Differences in the model included the automatic updating of criteria weights in the model. The decision-maker was required to enter the weights. Zhou and Fan (2007) developed an extended version of the R-model. In addition to the R-model, they calculated good index and bad index values by calculating the maximum and minimum objective function values and combined these values to obtain a composite index. Hadi-Vencheh (2010) (called with HV-model) stated the weights for each item in the Ng model did not have an effect when deciding total score and added weights to the Ng model to develop it. Park, Bae, & Bae (2014) developed a new model called cross-evaluation-based weighted linear optimization (CE-WLO) to find the optimal efficiency score for inventory items. Hatefi, Torabi, & Bagheri (2014) presented linear optimization model that include qualitative and quantitative criteria to classify inventory items. When all these studies are examined, different classifications are obtained by applying various methods to a 47-item dataset proposed by Reid (1987). Some studies decide the criteria weights when giving classification weights, while some studies focus on methods to minimize total inventory costs. Hadi-Vencheh and Mohamadghasemi (2011) suggested a fuzzy AHP method for MCIC problem.

In the first classification studies by Guvenir and Erel (1998) they used the genetic algorithm (GA) method. Mohammaditabar, Ghodsypour, and O'brien (2012) proposed simulated annealing (SA) minimizing total inventory costs and dissimilarity index value. They took the total weights of the two objective functions to determine classification.

Lolli, Ishizaka, and Gamberini (2014) described a model using AHP and K-means algorithms. The aim of the K-means algorithm is to minimize the distance between the centers of each cluster. Firstly, items are listed according to AHP, and then divided into classes with K-means. This method was called AHP-K in the literature. The method named AHP-K-Veto determines the class for each criterion and prevents a piece placed in a class for one or more than one criteria from being moved to another class in the final classification. Soylu and Akyol (2014) proposed determination of reference items with the assumption that criteria will have different weights in different industries. The aim of the studies was to determine classifications with minimized total classification error using linear utility and piece-wise linear function with reference items.

Chen et al. (2008) studied MCIC problem by using the case-based distance model. Kaabi, Jabeur, and Enneifar (2015) decided on weights with the continuous variable neighborhood search technique and then applied TOPSIS to calculate the score for each item. According to the determined score, inventory items were classified and classification was assessed according to inventory cost. Inventory cost was compared the traditional ABC, AHP model, R-model and

Mohammaditabar (SA model) in previous studies to reach the best inventory cost. Ghorabaee, Zavadskas, Olfat, & Turskis (2015) developed a new MCIC method that is called EDAS (Evaluation based on Distance from average Soluiton) by using appraisal score for all inventory items. Zhang, Zhao, and Li (2018) proposed fuzzy clustering-means (FCM) method in the newest of all MCIC studies. They targeted increased speed with GA and local search with simulated annealing (SA). Calculating total annual inventory costs, they compared with the previous 9 studies. They found classifications achieving inventory cost with variable holding cost (hi=0.2 and si=0.5) and ordering cost levels $1232.30.

Kaabi, Jabeur, and Ladhari (2018) used GA to decide on criteria weights and weighted sum (WS) and TOPSIS methods to find the weighted score of items and developed a hybrid model. In their study, they made decisions according to two performance values. One of the objectives was to minimize total inventory cost, while the other was to maximize inventory cycle rate. They studied 4 different forms of GA-WS and GA-TOPSIS for each aim separately. The best result was obtained with GA-WS. Additionally, in this study, the MCIC studies are summarized in detail as tables. (Kheybari, Naji, Rezaie, & Salehpour, 2019) suggested goal programming for multi-objective decision anaylsis. The main purpose of their study is to minimize deviation from the targets. Douissa & Jabeur (2019) proposed ELECTRE III method to compute the global score of each inventory items and then applied Continuous Variable Neighborhood Search (CVNS) to estimate the required parameters. Some of the previous literature used to evaluate the performance of their algorithms with Reid dataset is summarized as in Table 1.

This paper proposed a novel approach to make an ABC classsification with two objectives. The first objective is to minimize the Total Relevant Cost and the second one is to minimize the dissimilarity. We will utilize the concept of satisfaction level to combine the manager's decisions. We proposed SSA that is not used before for MCIC problem. This model seeks to obtain the best classsification that considers simultaneously conflicting objectives: TRC and dissimilarity index.

The outline of this paper is organized as follows. In Section 2, we defined the proposed mathematical model. We explain the scatter search algorithm in Sections 3. We present computational results in Section 4. We conclude with a discussion and further research directions in Section 5.

Table 1
The published literature based on the ABC analysis

Author, year	Model Formulation	Obective	Criteria	Benchmark
Reid, (1987)	AHP	weight of criteria	ADU	no
Flores et al., (1992)	AHP	weight of criteria	ADU, ACU, LT, CF	ADU
Ramanathan, (2006)	DEA-like weighted linear optimization	weight of criteria	ADU, ACU, LT, CF	ADU, AHP
Ng, (2007)	DEA-like weighted linear optimization	weight of criteria	ADU, ACU, LT	ADU, R
Zhou and Fan, (2007)	DEA-like weighted linear optimization	weight of criteria	ADU, ACU, LT	R
Chen et al., (2008)	case-based distance model	weight of criteria	ADU, ACU, LT, CF	AHP
Hadi-Vencheh, (2010)	DEA-like weighted linear optimization	weight of criteria	ADU, ACU, LT	Ng, ZF
Chen, (2012)	TOPSIS-DEA based	weight of criteria	ADU, ACU, LT, CF	RC, R, ZF, Ng, ADU
Mohammaditabar et al., (2012)	SA	minimize TRC and dissimilarity	ADU, ACU, LT	ADU, Zhang, AHP, R
Park et al., (2014)	cross-evaluation-based weighted linear optimization (CE-WLO) model, DEA-based and simulation	importance index, minimize the weights	ADU, ACU, LT	R, RC, ZF, Ng
Soylu and Akyol, (2014)	analytical, heuristic	minimize the total classification error	ADU, ACU, LT	AHP, R, ZF, RC
Lolli et al., (2014)	AHP, K-means	minimize the sum of squared distance	ADU, ACU, LT, CF	AHP, R, Ng, ZF, HV, RC, SA
Hatefi et al., (2014)	DEA-like weighted linear optimization	weight of criteria	ADU, ACU, LT, CF	AHP, RC, ZF
Kaabi et al., (2015)	TOPSIS, VNS	euclidean distance	ADU, ACU, LT	ADU, AHP, R, SA
Ghorabaee et al., (2015)	Evaluation Based on Distance from Average Solution (EDAS)	euclidean distance	ADU, ACU, LT	R, Ng, ZF, HV, RC, HT
Kaabi et al., (2018)	TOPSIS, WS, GA	Minimizing the Total Relevant Cost (TRC) and maximizing the Inventory Turnover Ratio (ITR).	ADU, ACU, LT	ADU, AHP, R, Ng, HV
Zhang et al., (2018)	GSSA -FCM (GA and SA)	minimize distance	ADU, LT, CF	AHP, R, Ng, ZF, HV, RC, FCM, AHP-K, AHP-K-Veto
Douissa and Jabeur, (2019)	Electre III, Variable neighborhood search	minimize TRC	ADU, ACU, LT	R, ZF, RC, Ng, HV, TOPSIS
Kheybari et al. (2019)	Shannon's entropy, TOPSIS, Goal programming	weight of criteria, minimize the deviation of goals	ADU, ACU, LT, CF	ADU, AHP, R, Ng
Proposed approach	Scatter Search Algorithm (SSA)	Minimize TRC, minimize dissimilarity, maximize satisfaction level	ADU, ACU, LT	SA, AHP-K-VETO; Kaabi (2015); Zhang; Kaabi (2018); DJ

ADU: Traditional ABC; AHP: Flores et al (1992); AHP-K, AHP-K-VETO: Lolli et al.(2014); ZF: Zhou and Fan (2007); SA: Mohammaditabar (2012); RC: Chen (2011); TOPSIS: Chen (2012); DJ: Douissa and Jabeur (2019)

2. Model description

In this study, our aim is to find the best classification using satisfaction functions first defined by Martel and Aouni (1996) utilizing two objective functions. In this situation, we developed a model for maximizing satisfaction levels to provide convenience to decision-makers due to having two objective functions. The Total Relevant Cost (TRC), which is one of the objective functions of our algorithm used in studies by Mohammaditabar et al. (2012), Ramanathan (2006), and Kaabi et al. (2018) to minimize inventory costs. However, materials with more than one criterion are effective in classification (Cohen, M. A., Ernst, 1988; B. E. Flores et al., 1992). Like the other item classification methods of K-means clustering (Jain, 2010; Keskin & Ozkan, 2013; Smet & Guzmán, 2004; Sun, Wang, & Jiang, 2004) the aim is to place items with close criteria values in the same class. The second objective function is defined to minimize dissimilarity index value between items in the same class. It was noted that MCIC problems are not studies noting the choices of managers to satisfy the two obejctives with together. In this study, we divided into three models type: minimize the TRC, minimize the dissimilarity index, and maximize satisfaction level function, respectively. There are some notations to explain the model as given below:

Parameters:

N	number of inventory items, $n=1, \ldots, N$
M	number of criteria, $m=1, \ldots, M$
K	number of classification groups, $k=1,2,3$
s_i	ordering cost of item i
T_k	order interval of classification class k
a_i	total annual demand of item i
p_i	unit purchasing price of item i
I_e	interest rate
h_i	holding cost of per unit i for per unit of time; $h_i=p_i*I_e$
x_{ik}	1, if item i and item j is classified in class k; 0, otherwise
y_{im}	criteria value of item i for criteria m
w_m	weights of the *mth* criteria

2.1.1. Minimize total relevant cost

The Total Relevant Cost (TRC), which is one of the objective functions of our algorithm, can be written as in Eq. (2). TRC might also be referred to as ordering cost and inventory holding cost. The larger the quantity of inventory is held, the more the inventory holding cost is incurred. Inventory holding cost incorporates storage cost, capital cost, obsolescence, damage, shrinkage, deterioration and spoilage cost, and insurance and tax costs. It is not easy to estimate the ratio of the inventory holding cost to the total value of inventory. Although various ratios are to be found in the literature, in practice, the annual holding cost is assumed to be, in accordance with the company structure, within the interval of 15-30% of the inventory investment value. Ordering cost depends on the number of orders placed in a planning period. This cost may be reduced by placing fewer orders through placing higher volume order at a time (Tersine, 1994).

Model 1:

Objective Function:

$$TRC^{min} = Minimize \sum_{k} \left(\frac{\sum_{i \in class(k)} s_i}{T_k} + {}^{1}/_{2}\, T_k \sum_{i \in class(k)} a_i h_i \right) \tag{1}$$

where T_k is explained by (Chakravarty, 1985), the optimal cycle length for class k can be calculated as given in Eq. (2).

$$T_k = \sqrt{\frac{2 \sum_{i \in class(k)} s_i}{\sum_{i \in class(k)} a_i h_i}} \tag{2}$$

Subject to:

$$\sum_{k=1}^{K} x_{ik} = 1, \forall i \tag{3}$$

$$x_{ik} \in \{0,1\} \ \forall i, k \tag{4}$$

Eq. (3) ensures that every item is assigned to the one class. Eq. (4) defines the binary variable: equal to one if an item i is assigned to class k; otherwise, zero

2.1.2. Minimize dissimilarity index

The objective is to minimize the dissimilarity index between the items i and j of the same class in k as given in Eq. (5). Min ds^{min} defines the minimize dissimilarity index as shown in notations with ds.

Model 2:

Objective Function:

$$ds^{min} = Minimize \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{K} d_{ij} x_{ik} x_{jk} \tag{5}$$

where d_{ij} shows that distance value between the item i and item j by considering the weights of each criteria and calculates with Eq. (6).

$$d_{ij} = \left[\sum_{m=1}^{M} w_m \left(\tilde{y}_{im} - \tilde{y}_{jm} \right)^2 \right]^{1/2} \tag{6}$$

In Eq. (6), $\tilde{y}_{im}$ defines the normalized value of the *ith* item for *mth* criteria to covert the criteria values in a 0-1 scale because of the difference measurement values. Normalized value for each item is calculated as given in Eq. (7).

$$\tilde{y}_{im} = \frac{y_{im} - min_{i=1,\dots N}\{y_{im}\}}{max_{i=1,\dots N}\{y_{im}\} - min_{i=1,\dots N}\{y_{im}\}} \tag{7}$$

Subject to:

Eqs. (3) and (4).

2.1.3. Maximize the satisfaction level

Two objective functions are written in Eq. (1) and Eq. (5) are the one is *TRC* minimization (TRC^{min}) and the second one is to minimize dissimilarity index (ds^{min}) between the criteria values. The main objective of this study is to maximize the two objectives values by considering together. For this reason, we decided to solve the problem with the objective value that calculates the satisfaction level. The objective is to maximize the total satisfaction value of *TRC* and *ds* at the same time and can be written as Eq. (8). The satisfaction level for *TRC* functions is defined by using the δ_1. The satisfaction level for *ds* functions is defined by using the δ_2.

Model 3:

Objective Function:

$$Maximize\ \delta_1 + \delta_2 \tag{8}$$

Subject to:

$$\delta_1 \leq (TRC^{max} - TRC^{*})/(TRC^{max} - TRC^{min}) \tag{9}$$

$$\delta_2 \leq (ds^{max} - ds^{*})/(ds^{max} - ds^{min}) \tag{10}$$

$$\delta_1, \delta_2 \leq 1 \tag{11}$$

Eqs. (3) and (4).

The first objective function in Eq. (9) shows how well the ideal point can be achieved. The second objective function in Eq. (10) shows how well the ideal distance value can be achieved. Eq. (11) ensures δ_1 and δ_2 are smaller than 1.

To calculate the δ_1 value, the minimum value that the *TRC* function can have is found using Eqs. (1, 2, 3, 11, and 12). To calculate the δ_2 value, the minimum value the distance function can have is found using Eqs. (4, 6, 7, 11, and 12). Our aim is to find satisfaction values that ensure both values are at maximum levels (Fig. 1). According to Fig. 1 (a) if the *TRC* minimum value is provided, the satisfaction level will be 1. If the *TRC* maximum value is provided, the satisfaction level will be 0. Similarly, in Fig. 1 (b) if the *ds* value is at minimum levels, maximum satisfaction level will be obtained; in other words, the satisfaction level will be 1. If the *ds* value is at maximum level, the satisfaction level will be 0. According to Mohammaditabar et al. (2012), this model is very difficult to solve optimally since the objective functions are not linear and the value of T_k is dependent on the value of x_{ik}. In this study, SSA is proposed to solve the problem of obtaining optimum or near optimum solutions in reasonable computation time because of the MINLP model.

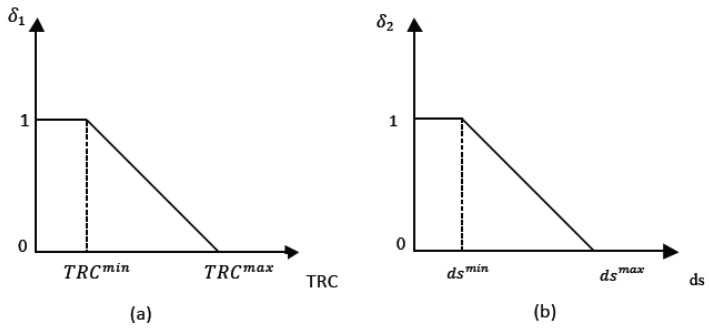

Fig. 1. The satisfaction function for *TRC* (a) and *ds* (b)

3.Scatter Search Algorithm

As the developed model was an MINLP model, there was a polynomial increase in the working duration of the model as the number of items increased. As a result, to attain the optimum result in a more suitable duration, solutions were explored with SSA, one of the evolutionary algorithms. SSA was first developed by Glover (1977). This method, which is very similar to GA, has some differences to GA. One of these is that GA is a stochastic structure, while SSA makes decisions deterministically. The SSA which is an evolutionary approach derives new solutions of the reference set from one set of solutions. Unlike the GA, the SSA tries to find a solution through the smaller reference set. In the GA, he creates new solutions by applying crossover and mutation to two randomly selected solutions from the population. The SSA approach was used for scheduling problems (Fan, Wang, Zhai, & Li, 2019; Riahi, Khorramizadeh, Hakim Newton, & Sattar, 2017), for the uncapacitated facility location problem Hakli and Ortacay (2019), for the economic lot sizing problem (Khojaste Sarakhsi, Fatemi Ghomi, & Karimi, 2016) and for the multiobjective clustering problem (Caballero et al., 2011)). SSA can basically be defined with 5 basic components (Martí, Laguna, & Glover, 2006).

1. *A Diversification Generation Method* (DGM) produces good and diverse individuals to create an initial population.
2. *Improvement Method* (IM) is applied to develop new individuals from the initial solutions.
3. *Reference Set Update Method* (RSM), which is included in the reference set of better individuals, removing the bad individuals from the reference set.
4. *The Subset Generation Method* (SGM) creates subsets from individuals in the reference set.
5. *The Combination Method* (CM) combines the individuals in the subset to create new individuals.

In general, the flow diagram of the SSA is shown in Fig. 2 as follows:

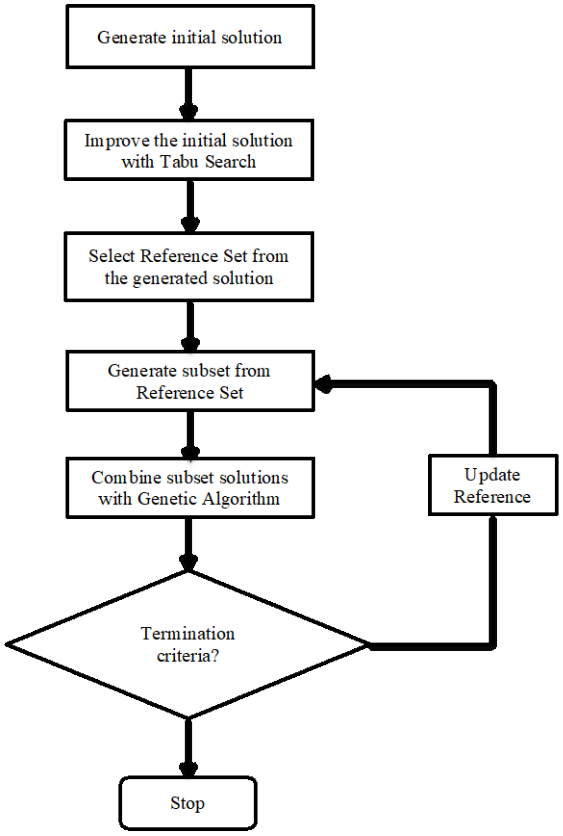

Fig. 2. Flowchart of the SSA

3.1. Diversification Generation Method

Various solution sets are created for decision variables without considering the objective function value. To get new solutions from a seed solution, new vectors are created in the form of the $v(k,s) = (s, s+k, s+2k, \dots, s+pk)$ in the length of the $\sum_{i=1}^{n} t_i$. Here, $k,s,p \in N\backslash\{0\}$ $s \leq k \leq \sum t_i$ and $s + pk \leq \sum t_i$. Position vector $pv(k) = (v(k,k), v(k,k-1), \dots, v(k,1))$ is defined for each positive integer $k \leq \sum t_i$.

For example, if a seed solution is $seed = (1,3,3,3,2,2)$ then $\sum_{i=1}^{3} t_i = 6$ position set can be established. Seed solution shows that the first element of the vector is assigned to class one, the second element is assigned to class three, and so on. If *k=3*, it will be $pv(3) = \big(v(3,3), v(3,2), v(3,1)\big) = pv\ (3,6,2,5,1,4)$. This *pv* vector represents that element 1 in the seed solution should be assigned to position 3 and the second element to position 6. Thus, the relevant new solution vector will be $(2,3,1,2,3,3)$. If *k=4*, $pv(4) = \big(v(4,4), v(4,3), v(4,2), v(4,1)\big) = pv(4,3,2,6,1,5)$ and will be obtained as the new vector solution $(2,3,3,1,2,3)$. Among the new solutions obtained in this way, 2 best solutions are chosen from the solutions with the best goal value.

3.2. Improvement Method

Tabu Search (TS) is a widely used metaheuristic that uses some commen-sense ideas to enable search process to escape from a local optimum. TS algorithm was proposed by Fred Glover et al. (1997). This metod is a metaheuristic metod to local optimization with neighboorhod solutions. The basic approach prevents or punishes repetitions in the next cycle

to prevent circular movements in steps leading to the final solution. Thus, by investigating new solutions, the TS algorithm guides regional heuristic searches to search for solutions which are more advanced than the regional best solution. The basic principle of the TS algorithm aiming to exceed regional optimals is based on selecting the movement with highest assessment value for each iteration of the assessment function to create the next solution. With the aim of ensuring this, a tabu list is created and the original aim of the tabu list was to prevent inversion of a previous movement rather than repetition. The tabu list has chronologic structure and used flexible memory structure. Though the TS algorithm is explained as labelling unwanted points, in practice it is used to label more attractive points. For example, if the first solution vector is $(2,3,3,1,2,3)$ a neighboring relationship of n*(n-1)/2=6*5/2=15 may form for the solution vector created from the 6 elements. As a result of swaps with neighbors, the variation ensuring best development from the target function is added to the tabu list. For example, the variation in the objective function as a result of the swap procedure with neighbors is shown below.

Initial Trial Solution: (2,3,3,1,2,3) TRC= 308.378 $

If we swap the first element and the second element the new solution will be (3,2,3,1,2,3) and the result is 309.562$. If we swap 4th item class and 5th item class, another new solution will be (2,3,3,2,1,3) and TRC is calculated of 306.430$. So, the 4th item class and 5th item class will be the tabu list and not changed in the next iteration because of the better TRC value. TS algorithm continues to get better solutions and generated new solution vectors.

3.3. Reference Set Update Method

The reference set consisting of high quality solutions and the sum of various solutions is used to derive new solutions by applying *Refset*. *RefSet* consists of two sub-sets which are *b1* and *b2*, that is, *RefSet* = *b* = *b1* + *b2*. The best b1 solution is selected from the first *P* population and these solutions are added to RefSet and deleted from *P* (Glover, Laguna, & Martí, 2004).

The euclidean distance value is calculated by comparing each solution in *RefSet1* with the solutions in *P*. Maximum is selected from the solutions with minimum distance and this solution is added to *b2* and deleted from *P*. This process continues for the specified size of *b2*. *Refset* includes *b1* high quality results and *b2* diverse results. Then, the minimum distance is calculated between each *x* solution in *P-Refset* and *y* solutions in *RefSet*. This calculation gives the Euclidean distance between *x* and *y* as follows:

$$d_{min}(x) = \underset{y \in RefSet}{\text{Min}} \{d(x,y)\}$$

For example, we suppose that we have 5 solutions for *P* given in Fig. 3. The best two solutions in Fig. 3 are Refset1= $\{S_4, S_5\}$ as shown in Fig. 4.

S_1	1	1	2	1	3	1	\$314.780
S_2	2	3	3	1	3	2	\$309.088
S_3	3	3	2	3	1	2	\$312.738
S_4	1	3	3	1	3	2	\$306.280
S_5	3	3	1	3	3	2	\$306.212

Fig. 3. Improved solutions.

S_4	1	3	3	1	3	2	\$306.280
S_5	3	3	1	3	3	2	\$306.212

Fig. 4. The best solutions of Refset1

Solutions $\{S_1, S_2, S3\}$ are the other solutions in the *P*. The minimum distance between the solutions S_1 and the solutions S_4 and S_5 in *Refset1* is calculated by using the Eq. (12) and the results are shown as in Fig. 5.

$$d(x,y) = \sqrt{\sum_{i \in P} (x_i - y_i)^2} \tag{12}$$

The minimum distance between the 1st solution and 4th and 5th solutions is between solutions 1 and 4 giving. The minimum distance between the 2nd solution and 4th and 5th solutions occurs between 2 and 4. The minimum distance between the 3rd solution and 4th and 5th solutions is between 3 and 4. This calculation is shown as follows:

$$d_{min}(S_1) = min\{d(S_1,S_4), d(S_1,S_5)\}$$

$$d_{min}(S_1) = min\{2.449, 3.741\} = 2.449$$

These distances are (2.449, 1, 2.236), and it appears the maximum distance is between 1st and 4th solutions. In conclusion, the 1st solution is added to refset2 and deleted from P solution space.

{S_1, S_4}	0	4	1	0	0	1	2.449
{S_1, S_5}	4	4	1	4	0	1	3.741

{S_2, S_4}	1	0	0	0	0	0	1
{S_2, S_5}	1	0	4	4	0	0	3

{S_3, S_4}	4	0	1	4	4	0	3.605
{S_3, S_5}	0	0	1	0	4	0	2.236

Fig. 5. Distance value between Refset1 and the P-Refset1

And update the minimum distance values. The new maximum values between the minimum distance values of 2.236 corresponding with solution 3 in Fig. 3. Finally, diverse Refset2 is as shown in Fig. 6.

S_1	1	1	2	1	3	1	$314.780
S_3	3	3	2	3	1	2	$312.738

Fig. 6. Diverse subset of *Refset2*

3.4. Subset Generation Method

In this step, new solutions are tried to generate for diversification. SGM proposes to select the all possible combinations of solutions in Refset1 with solutions in Refset2. For example, suppose that we have b1=b2=2, i.e., Refset1 = {S4, S5} and Refset2 = {S1, S3}. According to the combinations of two Refset, 4 subsets will be generated as follows: {S4, S1}, {S4, S3}, {S5, S1}, {S5, S3}.

3.5. Combination method

Attempts are made to obtain new solutions with the subset solutions created with this method. For this, various methods are recommended (Fred Glover et al., 1997). Here, the method used in the study by Caballero et al. (2011) was applied. The following methods were applied in an attempt to obtain alternative solutions. Let the selected subsets be 1st and 4th

solutions. Random numbers are derived for each element. According to whether the random numbers are smaller or larger than 0.5, elements are exchanged with each other and new solutions obtained. For example, let the generated random numbers be (0.42, 0.26, 0.18, 0.03, 0.87, 0.67). As the 1st, 2nd, 3rd, and 4th elements are smaller than 0.5, they remain the same. The 5th and 6th elements change places and a new solution given below Fig. 7 is obtained.

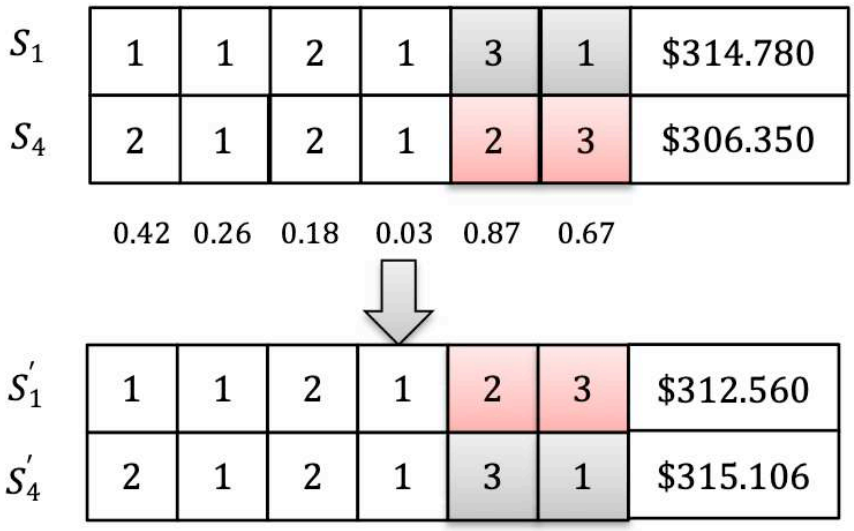

Fig. 7. Combination of two solutions

After ending these steps, finally various stopping conditions are proposed for the termination or stopping criteria. The most widely used ones are to stop the algorithm after a specific number of generations, or after a given time period. Another way is to stop the search when the objective values for several consecutive generations do not improve.

4. Illustrative example

In this study, we apply SSA approach to get the maximum satisfaction level of classification of MCIC. A computational study is conducted by running the model for 47 inventory items used in a Hospital Respiratory Therapy Unit (Reid, 1987) to evaluate the performance of the proposed SSA. We consider the ADU, AUC, and LT as criteria, in this case, to be able to compare the results with the results of other algorithms. As the CF criterion is categoric it was not included in this study, just as for the some previous studies (J.-X. Chen, 2012; Hadi-Vencheh & Mohamadghasemi, 2011; Ng, 2007; Zhou & Fan, 2007). The data and normalized data values are given in Table 2.

Table 2
Data for multi criteria classification problem

Item No	ADU ($)	ACU ($)	LT (week)	Normalized ADU	Normalized ACU	Normalized LT
1	5840.64	49.92	2	1.000	0.219	0.167
2	5670.00	210.00	5	0.971	1.000	0.667
3	5037.12	23.76	4	0.862	0.091	0.500
4	4769.56	27.73	1	0.816	0.110	0.000
5	3478.80	57.98	3	0.594	0.258	0.333
6	2936.67	31.24	3	0.501	0.127	0.333
7	2820.00	28.20	3	0.481	0.113	0.333
8	2640.00	55.00	4	0.450	0.243	0.500

9	2423.52	73.44	6	0.412	0.333	0.833
10	2407.50	160.50	4	0.410	0.758	0.500
11	1075.20	5.12	2	0.181	0.000	0.667
12	1043.50	20.87	5	0.175	0.077	0.667
13	1038.00	86.50	7	0.174	0.397	1.000
14	883.20	110.40	5	0.148	0.514	0.667
15	854.40	71.20	3	0.143	0.323	0.333
16	810.00	45.00	3	0.135	0.195	0.333
17	703.68	14.66	4	0.117	0.047	0.500
18	594.00	49.50	6	0.098	0.217	0.833
20	570.00	47.50	5	0.094	0.207	0.667
20	467.60	58.45	4	0.076	0.260	0.500
21	463.60	24.40	4	0.075	0.094	0.500
22	455.00	65.00	4	0.074	0.292	0.500
23	432.50	86.50	4	0.070	0.397	0.500
24	398.40	33.20	3	0.064	0.137	0.333
25	370.50	37.05	1	0.059	0.156	0.000
26	338.40	33.84	3	0.054	0.140	0.333
27	336.12	84.03	1	0.053	0.385	0.000
28	313.60	78.40	6	0.050	0.358	0.833
29	268.68	134.34	7	0.042	0.631	1.000
30	224.00	56.00	1	0.034	0.248	0.000
31	216.00	72.00	5	0.033	0.326	0.667
32	212.08	53.02	2	0.032	0.234	0.167
33	197.92	49.48	5	0.030	0.217	0.667
34	190.89	7.07	7	0.028	0.010	1.000
35	181.80	60.60	3	0.027	0.271	0.333
36	163.28	40.82	3	0.024	0.174	0.333
37	150.00	30.00	5	0.021	0.121	0.667
38	134.80	67.40	3	0.019	0.304	0.333
39	119.20	59.60	5	0.016	0.266	0.667
40	103.36	51.68	6	0.013	0.227	0.833
41	79.20	19.80	2	0.009	0.072	0.167
42	75.40	37.70	2	0.009	0.159	0.167
43	59.78	29.89	5	0.006	0.121	0.667
44	48.30	48.30	3	0.004	0.211	0.333
45	34.40	34.40	7	0.002	0.143	1.000
46	28.80	28.80	3	0.001	0.116	0.333
47	25.38	8.46	5	0.000	0.016	0.667

The ADU, ACU and LT criteria weights used for classification were taken as (0.407, 0.037, 0.556) from the study by Kaabi et al., (2015). We compare our model with other MCIC studies in the literature (Traditional (ADU); (Douissa & Jabeur, 2019; Kaabi et al., 2015; Kaabi et al., 2018; Lolli et al., 2014; Mohammaditabar et al., 2012; Zhang et al., 2018) that is used to benchmark dataset by Reid (1987) and also evalute the TRC functions. To operate the model, firstly the MINLP model was programed with Lingo 2018 and tested with small size problems. The results obtained and durations were assessed in terms of the results and performance obtained with SSA. Due to the difficulty of obtaining a suitable solution in acceptable duration, the decision was made to use SSA, one of the metaheuristic methods not used in classification studies to date.

We implemented SSA with the Frontline Solver Platform and solved the problem on a computer with CPU Intel(R) Core(TM) i5-8365U CPU (1.60 GHz), memory 8 GB, Windows 10. The mathematical model was operated with 5 items, 10 items and 15 items with the Lingo optimization program. Classification with 5 items reached optimum results in a very short duration. For 10 items, global optimum result took 40 minutes to obtain. However, for 15 items the program operation was ended at the end of 3 hours and only local optimum results could be obtained. With SSA, the computational time had acceptable duration of less than 20 minutes.

We assume that the ordering cost is equivalent to the lead time multiplied by a fixed coefficient with 0.5 and the inventory holding cost is assumed to be 20% of the ACU. The demand is calculated by dividing annual dollar usage with the average item cost. With the aim

of being able to determine the best results for solution to our multi objective model, the optimized minimum *TRC* value was $1098.165, and dissimilarity index was 195.026 in Model 1. When we used Model 2 for the second performance criteria of dissimilarity index, the minimum dissimilarity index was calculated as 126.522 with TRC value $1149.241. The maximum values for *TRC* and dissimilarity index were chosen from the largest values obtained in the other compared studies. When we consider each of the two performance criteria together, to operate Model 3, Eq. (12) and (13) were organized as follows;

$$\delta_1 = \begin{cases} 1 & if\ 0 \leq \delta_1 < 1098.165 \\ (1367.586 - TRC^*)/(1367.586 - 1098.165) & if\ 1098.165 \leq \delta_1 < 1367.586 \\ 0 & if\ \delta_1 \geq 1367.586 \end{cases} \quad (12)$$

$$\delta_2 = \begin{cases} 1 & if\ 0 \leq \delta_2 < 126.522 \\ (379.054 - ds^*)/(379.054 - 126.522) & if\ 126.522 \leq \delta_2 < 379.054 \\ 0 & if\ \delta_2 \geq 379.054 \end{cases} \quad (13)$$

Eq. (12) fully meets the targeted TRC in situations when the TRC is lower than $1098.165 and satisfaction level will be 1. If TRC is larger than $1367.586, it will be an unwanted cost for management and satisfaction will be 0. Values between these two values will be calculated with the equation in Eq. (12). Eq. (13) shows that if the targeted dissimilarity index is lower than 126.522, the satisfaction level will be 1. If the dissimilarity index is higher than 379.054, the classification will have an unwanted distance value and satisfaction level will be 0. Values between these two values will be calculated with Eq. (13).

In this study, the ABC classification giving maximum satisfaction level was determined. Table 3 shows the ABC classification results obtained according to Model 3 for 47 items. When our algorithm is compared with other studies, Table 3 shows the best TRC, best dissimilarity index and best satisfaction level obtained.

Table 3
Comparison results for previous studies and new proposed model

Item No	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]
1	A	A	B	A	A	C	A	A	A	A	A
2	A	A	A	A	A	A	A	A	A	A	A
3	A	A	B	A	A	C	A	A	A	A	A
4	A	A	B	A	A	C	A	A	A	A	A
5	A	A	B	A	B	B	A	A	A	A	A
6	A	A	B	A	C	C	A	A	A	A	A
7	A	A	B	A	B	C	A	A	A	A	A
8	A	A	C	A	B	B	A	A	A	A	A
9	A	A	B	A	B	A	A	B	A	A	A
10	A	A	B	B	B	A	A	A	A	A	A
11	B	A	B	C	C	A	B	A	C	A	A
12	B	B	B	B	C	C	B	B	C	B	B
13	B	B	B	A	C	A	B	B	C	C	B
14	B	B	B	B	C	A	B	B	C	B	B
15	B	B	B	C	C	B	B	B	B	B	B
16	B	B	B	C	C	C	B	B	B	B	B
17	B	B	B	B	C	C	B	B	B	B	B
18	B	B	B	B	C	B	B	B	C	C	C
19	B	B	B	B	C	B	B	B	C	B	B
20	B	B	B	C	C	B	B	B	B	B	B
21	B	B	B	C	C	C	B	B	B	B	B
22	B	B	B	C	C	B	B	B	B	B	B
23	B	B	B	C	C	A	B	B	B	B	B
24	B	B	B	C	C	C	B	B	B	B	B
25	C	C	C	C	C	C	C	B	A	B	B
26	C	C	C	C	C	C	C	B	B	B	B
27	C	C	C	C	C	A	C	B	C	B	A
28	C	C	B	B	C	A	C	C	C	C	C
29	C	C	B	C	C	A	C	C	A	C	C
30	C	C	C	C	C	B	C	B	C	B	B
31	C	B	B	B	C	B	C	C	B	C	C
32	C	B	B	C	C	C	C	B	B	B	B
33	C	C	C	B	C	B	C	C	C	C	C
34	C	C	B	B	C	C	C	C	C	C	C
35	C	C	C	C	C	B	C	C	B	B	B
36	C	B	B	C	C	C	C	C	B	B	C
37	C	C	C	B	C	C	C	C	B	C	C
38	C	B	B	C	C	B	C	C	B	B	C
39	C	C	C	C	C	B	C	C	C	C	C
40	C	C	B	B	C	B	C	C	C	C	C
41	C	C	C	C	C	C	C	C	B	B	B
42	C	C	C	C	C	C	C	C	B	B	B
43	C	C	C	C	C	C	C	C	C	C	C
44	C	C	C	C	C	C	C	C	B	C	C
45	C	C	B	B	C	C	C	C	C	C	C
46	C	C	C	C	C	C	C	C	B	C	C
47	C	C	C	C	C	C	C	C	C	C	C

[1]: Traditional (ADU); [2]: Mohammaditabar (2012); [3]: Lolli (2014)-AHP-K Veto; [4]: Kaabi 2015; [5]: Zhang 2018; [6]: Kaabi 2018; [7]: Douissa (2019); [8]: Proposed SSA-TRC (2020); [9]: Proposed SSA-ds (2020); [10]: Proposed SSA-two objective (2020); [11]: Proposed GA-two objective (2020)

By applying the SSA Frontline Solver Platform, 1.838 total satisfaction level could be obtained in a short duration like 13 minutes. TRC value was calculated as $1138.539 and dissimilarity index was 129.568. With our recommended SSA, not previously used for MCIM problems, better results were provided compared to other studies, as shown in Table 4. Proposed SSA is terminated when the stopping criteria reached for desired solving time applied in order to find the best value of the satisfaction level. The change of the solution values by CPU time (sec) are shown in Fig. 8.

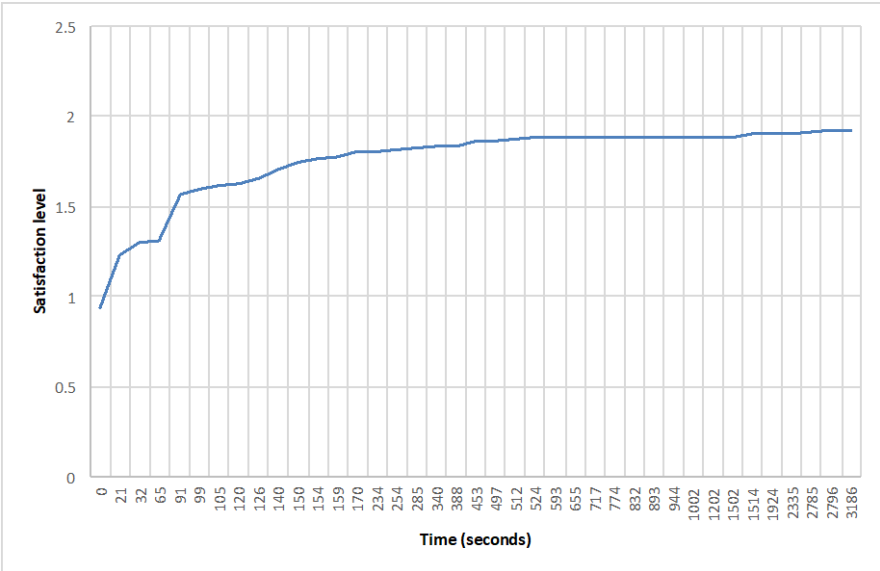

Fig. 8. The SSA solution time

Table 4
Comparison of the results of the proposed model and the other studies published before

	TRC cost ($)	Dissimilarity index	Satisfaction level for TRC	Satisfaction level for dissimilarity	Total satisfaction level
Traditional (ADU)	1119.779	207.389	0.920	0.680	1.600
Mohammaditabar (2012)	1121.536	194.603	0.913	0.730	1.644
Lolli (2014) AHP-K	1356.398	270.532	0.043	0.430	0.472
Kaabi (2015)	1190.228	179.603	0.659	0.790	1.448
Zhang (2018)	1171.209	379.054(4)	0.729	0.000	0.729
Kaabi (2018)	1367.586(3)	251.205	0.001	0.506	0.507
Douissa (2019)	1119.779	207.389	0.920	0.680	1.600
SSA - TRC	1098.165(1)	195.026	1.000	0.720	1.720
SSA - ds	1149.242	126.522(2)	0.803	1.000	1.803
SSA - two obj	1138.539	129.568	0.850	0.988	1.838(5)
GA-two obj	1099.028	191.956	0.997	0.732	1.729

(1): Minimum TRC; (2): Minimum dissimilarity index; (3): Maximum satisfaction level for TRC;
(4) Maximum satisfaction level for dissimilarity; (5) Maximum total satisfaction level

According to the Table 4 and Fig. 9., minimum TRC(1) ($1098.165) and minimum dissimilarity index(2) (126.522) values are obtained by using the our proposed SSA algorithm. The other best result for minimum TRC is calculated by with our proposed GA with two objectives. Traditional ABC analysis and Douissa & Jabeur (2019) calculated very closed to

our minimum TRC value of $1119.779 but dissimilarity index is far from our result. Mohammaditabar et al. (2012) also obtained minimum TRC value of $1121.536. The results shown in Table 4, conduct that the proposed SSA model (SSA-TRC, GA-two objective, SSA-ds, and SSA-two objective) provide the highest satisfaction level (1.720, 1.729, 1.803, and 1.838) respectively.

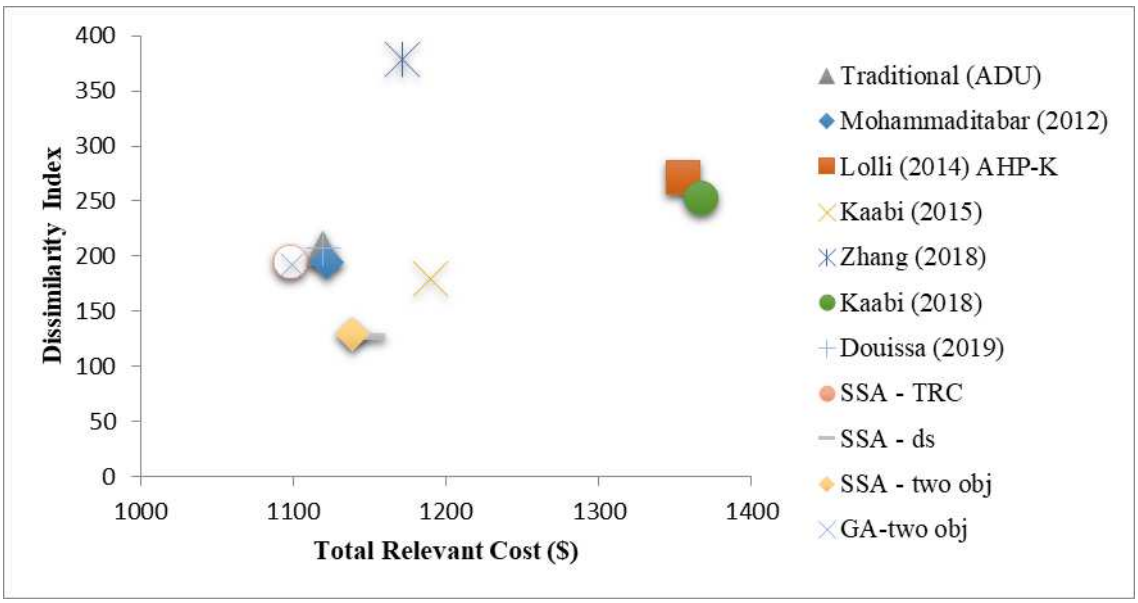

Fig. 9. Cost versus dissimilarity index values of comparison literature

Fig. 9. is shown the graph of the TRC and dissimilarity index values obtained from the other comparison studies. Minimum TRC and dissimilarity index value is calculated by our proposed SSA approach. According to Table 4, the highest similarity to the proposed model was observed for the study of Kaabi et al. (2015). Weight values are very important to calculate the dissimilarity index. According to the weight value, the item class is to change based on the most important criterion. In our study, satisfaction level was recommended ensuring both objective functions approach best values and solutions were found with the SSA approach. In our study, there were 11 items in class A, 16 items in class B and 20 items in class C. It can be seen from Table 5 that, due to having highest the similarity percentage with Kaabi et al. (2015). 8 out of the 47 items do not have the same classification. For example, item 13 was classified as class A by the Kaabi's model. However, it was classified as class B in our proposed model. If we classify this item in class A, TRC value wil be 54.96. But, if we assign this item in class B, TRC value will be 41.56. Thus, the total TRC value will be decrease by percentage of %24.38. We applied three models to get the best satisfaction level of two objectives. The first model is to minimize the TRC. The second is to minimize te dissimilarity index. The third one is two maximize the satisaction level of the two functions. It can be seen in Table 3, item 9 is classified class B for the first model. Item 9 has 58.54 TRC value, while has dissimilarity index 6.93. But, the same item is assigned to the A class in the second model and the TRC values and dissimilarity index changes as 60.28 and 3.94, respectively. Finally, item 9 is assigned to the A class to achive two functions because of the big decreasing in A class.

Table 5
Percentage of similarity results with other methods

Proposed Methods	Traditional			Mohammaditabar			Lolli K veto			Kaabi (2015)			Zhang		
	A	B	C	A	B	C	A	B	C	A	B	C	A	B	C
A	10	1		11			1	9	1	9	1	1	4	5	2
B		5	11		6	10		11	5	1	11	4			16
C		8	12		11	9		11	9		1	19			20
Similarity number	27			26			21			39			24		
similarity %	57,45			55,32			44,68			82,98			51,06		

Proposed Methods	Kaabi (2018)			Douissa			GA-two obj			SSA-TRC			SSA-ds		
	A	B	C	A	B	C	A	B	C	A	B	C	A	B	C
A	4	2	5	10	1		11			10	1		10	1	
B	4	6	6		5	11		4	12		5	11		16	
C	2	6	12		8	12	1	15	4		13	7			20
similarity number	22			27			19			22			46		
similarity %	46,81			57,45			40,43			46,81			97,87		

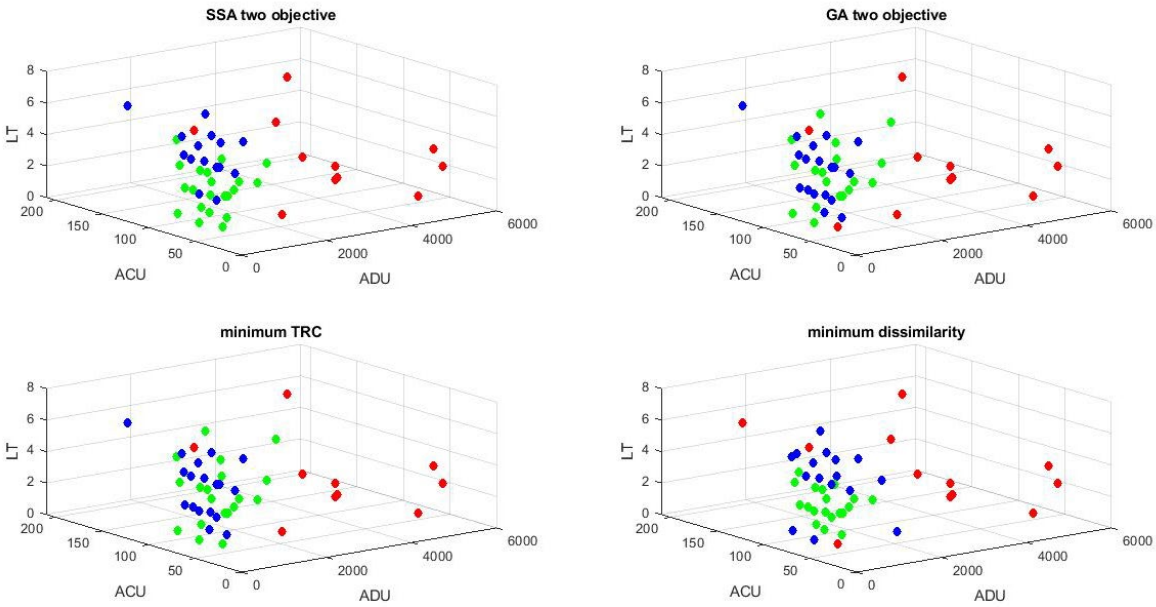

Fig. 10. The classification of items with different objective functions

We applied our model with using the SSA and GA approach for one objective and then for two objective simultaneously. Fig.10. shows that the classification results for four models. In this figure, different colors presented each class of items, i.e. red color shows A class of items, green color shows B class and blue color represents C class of items. It can be see this figure, there is not big changes in the A class items distriution, but B and C class have very big changes in each other.

5.Conclusion

The importance of inventory management problems gradually increases in today's economic and hard competitive conditions. Due to the increasing product range and growing alternative supply channels, reaching the desired product at desired time is one of the most important factors in obtaining customer satisfaction. The main step of the inventory management is started with the classification of the inventory items that consider multi criteria values. We have proposed a very efficient algorithm for helping inventory manager and company directors on how to classify the items about the inventory management.

In this paper, MCIC problem is formulated with the objectives of minimizing total inventory cost and dissimilarity index. The proposed model is developed as MINLP model. We offered three models to success the desired best results. The first model is to minimize the TRC function that includes the inventory holding cost and the ordering cost. The second model is to minimize the dissimilarity index. The aim of the second model is to assign inventory items that have the same criteria values to the same class. The SSA approach applied to solve the model because of the complexity of solving time. A SSA has been proposed and illustrated in this paper for classifying inventory items with multiple criteria and multiple objectives. The proposed model is compared with 7 studies published before. We selected these studies accordig to the used same Reid's dataset and used to compare Total Relevant Cost. It can be clearly seen that the proposed SSA algorithm for the MCIC problem exceeds the other methods. The proposed classification model is applied on the large inventory items that holding in the warehouse for example pharmaceutical items, spare parts, electronical components, and electromechanical items, etc.

In the future, the algorithm can be used to outperform existing methods for different models of classification of items. For solution of multipurpose problems, objectives are weighted and classification is performed. Studies are performed with fuzzy along with different inventory policies. These studies may be tested with real life problems. Different criteria may be added.

Compliance with Ethical Standards

Ethical approval: Humans/Animals are not involved in this work.

Funding details (In case of Funding): No funding was received to assist with the preparation of this manuscript.

Conflict of interest: The Author states that there is no conflict of interest.

References

Caballero, R., Laguna, M., Martí, R., & Molina, J. (2011). Scatter tabu search for multiobjective clustering problems. *Journal of the Operational Research Society*, *62*(11), 2034–2046. https://doi.org/10.1057/jors.2010.180

Chakravarty, A. K. (1985). An Optimal Heuristic for Coordinated Multi-Item Inventory Replenishments. In *The Journal of the Operational Research Society* (Vol. 36).

Chen, J.-X. (2012). Multiple criteria ABC inventory classification using two virtual items. *International Journal of Production Research*, *50*(6), 1702–1713. https://doi.org/10.1080/00207543.2011.560201

Chen, Y., Li, K. W., Kilgour, D. M., & Hipel, K. W. (2008). A case-based distance model for multiple criteria ABC analysis. *Computers and Operations Research*, *35*(3), 776–796. https://doi.org/10.1016/j.cor.2006.03.024

Cohen, M. A., Ernst, R. (1988). Multi-Item Classification And Generic Inventory Stock Contr ABI/INFORM Global pg. 6. In

Production and Inventory Management Journal; Third Quarter (Vol. 29).
Douissa, M. R., & Jabeur, K. (2019). A non-compensatory classification approach for multi-criteria ABC analysis. *Soft Computing*. https://doi.org/10.1007/s00500-019-04462-w
Fan, K., Wang, M., Zhai, Y., & Li, X. (2019). Scatter search algorithm for the multiprocessor task job-shop scheduling problem. *Computers and Industrial Engineering*, *127*, 677–686. https://doi.org/10.1016/j.cie.2018.11.006
Flores, B. E., Olson, D. L., & Dorai, V. K. (1992). Management of Multicriteria Inventory Classification. In *Mathl. Comput. Modelling* (Vol. 16).
Flores, Benito E., & Clay Whybark, D. (1986). Multiple Criteria ABC Analysis. *International Journal of Operations & Production Management*. https://doi.org/10.1108/eb054765
Frontline Solver (2019). <http://www.solver.com/> Accessed 02.04.19.
Ghorabaee, M. K., Zavadskas, E. K., Olfat, L., & Turskis, Z. (2015). Multi-Criteria Inventory Classification Using a New Method of Evaluation Based on Distance from Average Solution (EDAS). *Informatica (Netherlands)*, *26*(3), 435–451. https://doi.org/10.15388/Informatica.2015.57
Glover, F. (1977). Heuristics For Integer Programming Using Surrogate Constraints. *Decision Sciences,* 8(1), 156-166. https://doi.org/10.1111/j.1540-5915.1977.tb01074.x
Glover, F., Hao, K., Lutton, E., Ronald, E., Schoenauer, M., & Snyers, D. (1997). A Template For Scatter Search And Path Relinking. In *Lecture Notes in Computer Science*. (Vol.1363), 1-51.
Glover, F., Laguna, M., & Martí, R. (2004). Scatter Search and Path Relinking: Foundations and Advanced Designs. *New Optimization Techniques in Engineering.* (Vol.141), 87-99. https://doi.org/10.1007/978-3-540-39930-8_4
Guvenir, H. A., & Erel, E. (1998). Multicriteria inventory classification using a genetic algorithm. In *European Journal of Operational Research* (Vol. 105). https://doi.org/10.1016/s0377-2217(97)00039-8
Hadi-Vencheh, A. (2010). An improvement to multiple criteria ABC inventory classification. *European Journal of Operational Research*, *201*(3), 962–965. https://doi.org/10.1016/j.ejor.2009.04.013
Hadi-Vencheh, A., & Mohamadghasemi, A. (2011). A fuzzy AHP-DEA approach for multiple criteria ABC inventory classification. *Expert Systems with Applications*, *38*(4), 3346–3352. https://doi.org/10.1016/j.eswa.2010.08.119
Hakli, H., & Ortacay, Z. (2019). An improved scatter search algorithm for the uncapacitated facility location problem. *Computers and Industrial Engineering*, *135*, 855–867. https://doi.org/10.1016/j.cie.2019.06.060
Hatefi, S. M., Torabi, S. A., & Bagheri, P. (2014). Multi-criteria ABC inventory classification with mixed quantitative and qualitative criteria. *International Journal of Production Research*, *52*(3), 776–786. https://doi.org/10.1080/00207543.2013.838328
Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, *31*(8), 651–666. https://doi.org/10.1016/j.patrec.2009.09.011
Kaabi, H., Jabeur, K., & Enneifar, L. (2015). Learning criteria weights with TOPSIS method and continuous VNS for multi-criteria inventory classification. *Electronic Notes in Discrete Mathematics*, *47*, 197–204. https://doi.org/10.1016/j.endm.2014.11.026
Kaabi, Hadhami, Jabeur, K., & Ladhari, T. (2018). A genetic algorithm-based classification approach for multicriteria ABC analysis. *International Journal of Information Technology and Decision Making*, *17*(6), 1805–1837. https://doi.org/10.1142/S0219622018500475
Keskin, A. G., & Ozkan, C. (2013). Multiple Criteria ABC Analysis with FCM Clustering. *Journal of Industrial Engineering*, *2013*, 1–7. https://doi.org/10.1155/2013/827274
Kheybari, S., Naji, S. A., Rezaie, F. M., & Salehpour, R. (2019). ABC classification according to Pareto's principle: a hybrid methodology. *Opsearch*, *56*(2), 539–562. https://doi.org/10.1007/s12597-019-00365-4
Khojaste Sarakhsi, M., Fatemi Ghomi, S. M. T., & Karimi, B. (2016). A new hybrid algorithm of scatter search and Nelder-Mead algorithms to optimize joint economic lot sizing problem. *Journal of Computational and Applied Mathematics*, *292*, 387–401. https://doi.org/10.1016/j.cam.2015.07.027
Lolli, F., Ishizaka, A., & Gamberini, R. (2014). New AHP-based approaches for multi-criteria inventory classification. *International Journal of Production Economics*, *156*, 62–74. https://doi.org/10.1016/j.ijpe.2014.05.015
Martel, J.-M., & Aouni, B. (1996). *Incorporating the Decision-Maker's Preferences in the Goal Programming Model with Fuzzy Goal Values: A new Formulation*. https://doi.org/10.1007/978-3-642-87561-8_17
Martí, R., Laguna, M., & Glover, F. (2006). Principles of scatter search. *European Journal of Operational Research*, *169*(2), 359–372. https://doi.org/10.1016/j.ejor.2004.08.004
Mohammaditabar, D., Ghodsypour, H. S., & O'brien, C. (2012). Inventory control system design by integrating inventory classification and policy selection. *International Journal of Production Economics*, *140*(2), 655–659. https://doi.org/10.1016/j.ijpe.2011.03.012
Ng, W. L. (2007). A simple classifier for multiple criteria ABC analysis. *European Journal of Operational Research*, *177*(1), 344–353. https://doi.org/10.1016/j.ejor.2005.11.018
Park, J., Bae, H., & Bae, J. (2014). Cross-evaluation-based weighted linear optimization for multi-criteria ABC inventory classification. *Computers and Industrial Engineering*, *76*(1), 40–48. https://doi.org/10.1016/j.cie.2014.07.020
Partovi, F. Y., & Burton, J. (1993). Using the Analytic Hierarchy Process for ABC Analysis. *International Journal of Operations & Production Management*, *13*(9), 29–44.
Partovi, F. Y., & Hopton, W. E. (1993). The analytic hierarchy process as applied to two types of inventory problems. *Production and Inventory Management Journal*, *35*(1), 13–19.
Ramanathan, R. (2006). ABC inventory classification with multiple-criteria using weighted linear optimization. *Computers and Operations Research*, *33*(3), 695–700. https://doi.org/10.1016/j.cor.2004.07.014

Reid, R. (1987). Abc Method in Hospital Inventory Management: a Practical Approach. *Production and Inventory Management Journal*, *28*(4), 67–70.

Riahi, V., Khorramizadeh, M., Hakim Newton, M. A., & Sattar, A. (2017). Scatter search for mixed blocking flowshop scheduling. *Expert Systems with Applications*, *79*, 20–32. https://doi.org/10.1016/j.eswa.2017.02.027

Smet, Y. D., & Guzmán, L. M. (2004). Towards multicriteria clustering: An extension of the k-means algorithm. *European Journal of Operational Research*, *158*(2), 390–398. https://doi.org/10.1016/j.ejor.2003.06.012

Soylu, B., & Akyol, B. (2014). Multi-criteria inventory classification with reference items. *Computers and Industrial Engineering*, *69*(1), 12–20. https://doi.org/10.1016/j.cie.2013.12.011

Sun, H., Wang, S., & Jiang, Q. (2004). FCM-based model selection algorithms for determining the number of clusters. *Pattern Recognition*, *37*(10), 2027–2037. https://doi.org/10.1016/j.patcog.2004.03.012

Tersine, R. J. (1994). *Principles of inventory and materials management* (4th ed.). Pearson Education, Inc.

Zhang, Q., Zhao, Q., & Li, Y. (2018). Integrating replenishment policy with GSAA-FCM based multi-criteria inventory classification. *International Journal of Computational Intelligence Systems*, *11*(1), 248–255. https://doi.org/10.2991/ijcis.11.1.19

Zhou, P., & Fan, L. (2007). A note on multi-criteria ABC inventory classification using weighted linear optimization. *European Journal of Operational Research*, *182*(3), 1488–1491. https://doi.org/10.1016/j.ejor.2006.08.052