Accepted Manuscript

International Journal of Information Technology & Decision Making

Article Title: A Genetic Algorithm Based Classification Approach for Multicriteria ABC Analysis

Author(s): Hadhami Kaabi, Khaled Jabeur, Talel Ladhari

DOI: 10.1142/S0219622018500475

Received: 25 January 2017

Accepted: 02 July 2018

To be cited as: Hadhami Kaabi, Khaled Jabeur, Talel Ladhari, A Genetic Algorithm Based Classification Approach for Multicriteria ABC Analysis, *International Journal of Information Technology & Decision Making*, doi: 10.1142/S0219622018500475

Link to final version: https://doi.org/10.1142/S0219622018500475

International Journal of Information Technology & Decision Making

A GENETIC ALGORITHM BASED CLASSIFICATION APPROACH FOR MULTICRITERIA ABC ANALYSIS

ABC analysis is a widespread classification technique designed to manage inventory items in an effective way by relaxing controls on low valued items and applying more rigorous controls on high valued items. In the literature, many classification models issued from different methodologies such as Mathematical Programming (MP), Metaheuristics, Artificial Intelligence (AI) and Multicriteria Decision Making (MCDM) are proposed to perform the ABC inventory classification. To the best of our knowledge, the cross-fertilization of classification models issued from different methodologies is rarely tackled in the literature. This paper proposes some hybrid classification models based on both Genetic Algorithm (Metaheuristics) and two MCDM methods (Weighted Sum (WS) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)) to carry out the ABC inventory classification. To test the performance of the proposed classification models with respect to some existing models, a benchmark dataset from a Hospital Respiratory Therapy Unit (HRTU) is used. **The computational results show that our proposed models outperformed the existing classification models according to some inventory performance measures. An additional performance analysis has also shown the effectiveness of our proposed models in inventory management.**

Keywords: ABC Analysis; Multicriteria Classification; Genetic Algorithm; TOPSIS; Weighted Sum.

1. Introduction

ABC analysis is one of the most commonly used technique in inventory management to classify inventory items, referred to as Stock-Keeping Units (SKUs), into three predefined and ordered categories: Category A contains the most valuables items which need a tight and rigorous control, Category B contains the moderately valuable items and Category C includes the least valuable ones. The main aim of this technique is to manage with care the critical few items (A-items) and avoid wasting precious resources by managing trivial many items (C-items). In addition, managing inventory items based on their degree of importance will allow managers to keep inventory related costs under control and, therefore, improve the company competitiveness. In the classical ABC analysis, inventory items are categorized according to a single criterion called the Annual Dollar Usage (ADU). However, recent literature has identified many other criteria which may significantly affect the items classification such as Lead Time (LT), Average Unit Cost (AUC), Critical Factor (CF), Demand (D) and Turnover (T), etc. Hence, the ABC analysis is considered as a Multicriteria Inventory Classification (MCIC) technique. Hence, within a Multicriteria framework, the classification of items into categories is based on their weighted scores. The score of each item is the result of an aggregation function that combines the item evaluations on the different criteria and the criteria weights. Inventory items are then ranked in a descending order with respect to their scores. Finally, the ABC categories are built by using the following commonly used distribution[1]: The first ranked items (about 15-20% of total items) are classified in category A (high value items), items with the worst scores (about 40-50% of total items) are classified in category C (low value items) and the remaining items (about 30-40% of total items) are classified in category B (medium value items). It is important to underline that these percentages are not always preset as there are rough estimates and completely depend on the used dataset. In this case, there is no fixed percentage for each category and therefore different proportions may be applied based on the degree of importance associated to each inventory item[2]. This degree of importance may by unicriterion (e.g. Annual Dollar Usage, Lead Time, Sales, Profits, Unit Cost ...) or multicriteria, i.e. an aggregate value involving different criteria. In order to determine the percentage (or the number) of items to be attributed to each category, we should proceed according to the following four steps. First, we assign a degree of importance for each inventory item. Second, we sort the inventory items in the deceasing order of their degree of importance. Third, we compute the percentage and the cumulative percentage of items and their degree of importance. Forth, and instead of forcing categories to fit some predefined percentages, we group the items into A, B and C categories using "natural" breaking points identified from the above cumulative percentages. For instance, to identify the items of category A, we determine first the closest cumulative percentage of the items importance degrees to the generic value of 80% (issued from the Pareto principle or the 80/20 rule) [3]. Then, the items that contribute to the computation of this cumulative percentage will constitute the category A. A similar reasoning is applied to identify the items of categories B and C.

Most MCIC models proposed in the literature to carry out the ABC classification are based on techniques derived from four main methodologies: Mathematical Programming (MP), Metaheuristics, Artificial Intelligence (AI) and Multicriteria Decision Making (MCDM). In this paper, we argue the benefits of cross-fertilization of techniques issued from Metaheuristics and MCDM methodologies to perform the ABC inventory classification. For this purpose, some new hybrid MCIC models based on both Genetic Algorithm (Metaheuristics methodology) and two MCDM methods (Weighted Sum (WS) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)) will be proposed in this paper. The Genetic Algorithm (GA) is used to estimate the criteria weights and the MCDM methods are used to compute the weighted score of each inventory item. In fact, the MCDM methods are used inside of the GA to evaluate the objective functions (see Fig.1.). To measure the performance of our proposed GA based hybrid models with respect to some existing MCIC models, a comparative study - based on a service-cost evaluation of each inventory classification - will be conducted. To do this, a benchmark dataset of items consumed in a Hospital Respiratory Therapy Unit (HRTU) will be used.

This paper makes three main contributions. First, we develop some new GA based hybrid MCIC models that combine the benefits of techniques used in MCDM and metaheuristics methodologies. To the best of our knowledge, the hybridization of MCIC models issued from MCDM and metaheuristics methodologies is rarely tackled in the literature. Some attempts of such hybridization may be found in recent literature[4,5,6,15,33]. Second, each proposed GA based hybrid model produces an ABC classification of inventory items that optimizes one of the two main inventory performance measures: minimizing the Total Relevant Cost (TRC) and maximizing the Inventory Turnover Ratio (ITR). It should be noted that most of the existing MCIC models determine their ABC classifications of inventory items without considering any inventory performance measures. Indeed, other objective functions are used by these MCIC models to determine their ABC classification: maximizing the score of each item, minimizing the divergence between the desired ABC classification and a subjective classification provided by the decision-maker, etc. Third, we propose a comparative study based on a service-cost performance measure - recently proposed by Babai et al.[7] - to evaluate each ABC inventory classification. We believe that this performance measure may be a relevant tool to compare objectively the performance of the different MCIC models. It should be noted that most of the existing comparative studies of MCIC models are limited to a simple listing and elementary interpretations of the obtained inventory classifications (e.g. which items are identically classified by all compared MCIC models).

The remainder of this paper is organized as follows. Section 2 reviews the relevant literature on ABC inventory classification. Section 3 describes the general framework of the proposed GA-based hybrid MCIC models. In section 4, computational results of the proposed GA-based models are discussed and compared with those obtained by some existing MCIC models. Finally, concluding remarks and directions for further research are reported in section 5.

2. Literature review

Classifying inventory items into ordered and predefined (in term of percentage) ABC categories is the subject of an extensive literature. Despite this abundance, most of the existing MCIC models may be categorized into four main methodologies (or classes), and this, according to the type of techniques used by these models to perform the ABC classification: (i) MCIC models based on Mathematical Programming (MP) methodology, (ii) MCIC models based on Multicriteria Decision Making (MCDM) methodology, (iii) Metaheuristics and (iiii) MCIC models based on Artificial Intelligence (AI) methodology. Whatever the methodology used, two main topics are often addressed by these MCIC models: (i) The specification of the aggregation function used in the computation of the overall item score (e.g.

Weighted Sum), and (ii) The estimation of the aggregation function parameters (e.g. criteria weights, discrimination thresholds, etc.).

Classification models based on Mathematical Programming (MP) methodology propose linear and non linear programming models to generate a weight vector[a] or a weight matrix[b] that maximizes or minimizes the weighted score of each item. Hence, the overall score of each item is automatically generated by the objective function of these mathematical programming models. Models such as the R-model proposed by Ramanathan[8], the ZF-model - which is an extension of the R-model - suggested by Zhou and Fan[9], the Ng-model introduced by Ng[8], the H-model proposed by Vencheh[11] and the F-model proposed by Fu et al.[11], are some typical weighted optimization models of the Mathematical Programming methodology. Ramanathan[8] proposed a linear optimization model - called R-model - to obtain a set of optimal weights which maximizes the score of each item expressed by a weighted additive function. The main drawback of the R-model is that an inventory item with a high evaluation on an irrelevant criterion may be considered as a critical item, i.e. belonging to category A. To address this drawback, Zhou and Fan[9] proposed an extended version of the R-model (called ZF-model), in which two linear programming models are solved for each inventory item: the first one generates a set of weights that are the most favorable for each item whereas the second generates a set of weights that are the least favorable for each item. The outputs of these two linear programming models are then combined to compute the overall score of each item. Ng[10] proposed a linear optimization model (called Ng-model) in which the decision maker may integrate for each item a ranking of the criteria weights. After some mathematical transformations, Ng showed that his model may be solved without any linear optimizer and the optimal score of each item is obtained by a simple calculation mechanism. The main weakness of the Ng-model is that the optimal score of items does not depend on criteria weights. Vencheh[11] provides an improved version of Ng-model by proposing a nonlinear optimization model, called H-model, which maintains the effects of weights in the optimal scores of items. Note that both models - Ng-model and H-model - require that the decision-maker ranks subjectively the criteria weights in a sequence. To avoid this subjectivity, Fu et al.[12], proposed an extended version of the Ng-model that determines the common weights associated with all possible rankings of the criteria weights, and then provides a comprehensive scoring scheme. Ladhari et al.[13] have also developed a new hybrid MCIC model which combines the ZF-model[9] and the Ng-model[10] in order to reduce - through a consensus process - the conflict between some existing MP-based classification models. Kaabi and Jabeur[14] have proposed an hybrid mathematical programming model which combines the usefulness of the ZF-model and the H-model. By using a commonly used benchmark dataset, these authors have shown that their hybrid

[a]When these weights are established for each criterion
[b]When these weights are established for each (criterion, item) pair

model has obtained very promising results since it outperforms the H-model and Ng-model in term of minimizing the inventory cost. Recently, Baykasoglu et al.[15] have proposed a fuzzy linear assignment model to perform the ABC classification of inventory items. The proposed model is able to consider uncertain alternative evaluations with mixed quantitative and qualitative criteria by using various concepts of fuzzy set theory. According to these authors, the proposed model provides - in comparison with some commonly used MCIC benchmark models - effective categories under uncertain environment. **Zheng et al. [16] have also proposed an improvement of the Ng model [10] based upon Shannon entropy. The proposed model improves the Ng model by aggregating all criteria sequences based upon Shannon entropy.**

Classification models based on MCDM methodology proceed in two steps to classify an inventory item into one of the three ABC categories. In the first, a MCDM method - essentially the Analytic Hierarchy Process (AHP)[17] is applied once to determine the criteria weights. In the second step, an aggregation function is used to compute the overall score of each inventory item. The models proposed by Flores et al.[18], Partovi and Burton[19], Bhattacharya et al.[20] and Vencheh and Mohamadghasemi[21] are considered as typical MCDM based classification models. Flores et al.[18] proposed a classification model based on AHP method to determine the criteria weights and the Weighted Sum (WS) rule to compute the score of each inventory item. Partovi and Burton[19] proposed a classification model entirely based on the AHP method since the computation of the criteria weights and the item scores are both determined by this method. The main advantage of Partovi and Burton's model with respect to Flores et al.'s model is that the former considers both qualitative and quantitative criteria whereas the latter considers only quantitative criteria. Bhattacharya et al.[20] proposed a classification model based on AHP to determine the criteria weights and the TOPSIS method (Technique Order Preference by Similarity to Ideal Solution) to compute the score of each item. Vencheh and Mohamadghasemi[21] proposed an integrated Fuzzy AHP-DEA (Data Envelopment Analysis) classification model to estimate the criteria weights and the Weighted Sum (WS) rule to compute an overall score for each item. Recently, Ghorabaee et al.[22] have proposed a new classification model, called EDAS (Evaluation based on Distance from Average Solution), which uses the average solution to compute the overall score of each inventory item. Two measures, called PDA (Positive Distance from Average) and NDA (Negative Distance from Average), are considered to determine this overall score. According to these authors, the EDAS model seems to be more stable than some existing MCIC models when a sensibility analysis is carried out on the criteria weights. A novel approach is also proposed by Arikan and Citak [23], in which they have presented an integrated approach based on TOPSIS (to rank the alternatives) and AHP (to compute the criteria weights). The proposed approach is applied in an Electronics firm.
It is important to note that the MCDM based classification models have the advan-

tage of incorporating explicitly human judgments, considering conflicting criteria and dealing with data obtained on heterogeneous measurement scales (i.e. quantitative and qualitative). Moreover, MCDM based models avoid the black box situation, i.e. we may easily explain the computation details of the overall items scores produced by these models to carry out the ABC classification. Although, the AHP method is widely used in the MCDM based classification models, two main criticisms may be addressed to this method: (i) the number of pairwise comparisons increases rapidly when the number of criteria and/or items is large [25] and (ii) it is very demanding for the decision maker to specify by using Saaty scale[17] how much a criterion is more important than another criterion.

Classification models issued from both metaheuristics and AI methodologies propose a learning process that estimates the criteria weights in order to produce a classification of items that optimizes some objective functions (e.g. inventory cost function, misclassified items rate, correct classified items rate, etc.). The assessment of the criteria weights is carried out by some well-known metaheuristics such as Genetic Algorithm (GA)[26], Particle Swarm Optimization (PSO)[28] and Simulated Annealing (SA)[29]. Once the criteria weights are determined, a score is computed for each item by using some aggregation rules (e.g. weighted sum). For instance, Guvenir and Erel[26] use the genetic algorithm to learn the criteria weights and two cut-off thresholds (to delimit the frontier of the ABC categories) by maximizing the rate of correct classified items (fitness function). Then, they use a normalized weighted sum rule to compute the score of each item.
Yu[30] proposed a comparative study in which three (AI) based classification models (Support Vector Machines (SVMs), Back Propagation Networks (BPNs) and the K-Nearest Neighbor (K-NN)) and the traditional Multiple Discriminant Analysis (MDA) are tested. The results showed that the AI based classification models outperformed the MDA in term of prediction accuracy. Lopez et al.[31] have developed a pattern based classification model using a Logical Analysis of Data (LAD). The main aim of the proposed model is to detect and correct familiarity bias that may exist in the ABC item classification done by the decision makers and/or inventory experts. In their model, familiarity bias detection is performed with two supervised classification techniques, LAD and K-NN, by using cross validation. The experimental results show that both techniques (LAD and K-NN) are capable of correcting familiarity bias in experts classification, and they obtained, after the bias correction, substantial improvements on the averages prediction accuracies.

Recently, an emerging research direction consists in hybridizing classification models issued from different methodologies[4,5,6,15,36]. For instance, Lolli et al.[4] proposed an hybrid classification model based on AHP and the K-means algorithm. The particularity of their model is that it prevents an item with a high evaluation on an irrelevant criterion to be top ranked in the final classification. Kaabi et al.[5] have also proposed an hybrid classification model based on TOPSIS (MCDM methodology) method to compute the score of each item and the CVNS (Continuous Variable Neighborhood Search) (AI methodology) to estimate the criteria weights. These authors have shown that their model outperformed some existing classification models in terms of minimizing inventory cost. Liu et al.[6] proposed a new MCIC model based on the outranking method ELECTRE III[32] in order to consider the non-compensation concept in the computation of the overall score of each inventory item. In their proposed model, the clustering analysis and the simulated annealing algorithm are combined to find the optimal classification of inventory items. The clustering analysis groups similar inventory items together and builds up the hierarchy of clusters. The simulated annealing algorithm searches for the 'optimal' classification on different levels of the hierarchy.

The relevant literature is enhanced by a table which summarizes all relevant papers dealing with multicriteria ABC inventory classification (see Table 1). In this Table, the main features of the existing inventory classification models are presented: Model formulation (Linear/ Non Linear), whether the model formulation is linear or not in case of mathematical programming and whether the evaluation function is linear or not in other cases (AI, MCDM and metaheuristics), Evaluation of the inventory classification (single/multiple) indicates if the model uses a single evaluation function or multiple evaluation functions, Nature of criteria weights (Objective/ Subjective) indicates whether the criteria weights are determined by the decision maker (Subjective weights) or by using a process (Objective weights), Type of criteria (Quantitative/Qualitative), Inventory classification based on (Scores/ Inventory costs) indicates whether the used evaluation function is based on just optimizing the weighted scores or on optimizing inventory costs, whether the Service cost analysis is performed or not and last, the criteria used to perform the ABC classification.

In this paper, we propose some new hybrid GA based MCIC models that combine

[c]Annual Demand
[d]Unit Cost
[e]Annual Cost
[f]Consumption Rate
[g]Perishability of Items
[h]Storage Cost
[i]Limitation of Warehouse Space
[j]Unit Price
[k]Number of Requests
[l]Scarcity
[m]Substitutability
[n]Repearability
[o]Order Size Requirement
[p]Ordering Cost
[q]Set Up Cost
[r]Holding Cost
[s]Supplier Ordering Cost
[t]Unit Size

the usefulness of both MCDM and AI methodologies. GA (as an AI technique) is a general algorithmic framework, nature inspired, designed to solve complex and hard optimization problems. On the other hand, MCDM are designed essentially to integrate decision-maker knowledge and preferences and to rigorously manipulate heterogeneous, conflicting and non commensurable information. In each proposed GA based model, a new variant of the GA will be combined with a MCDM method (WS or TOPSIS). The Genetic Algorithm (GA) is used to estimate the criteria weights and the MCDM methods are used to compute the weighted score of each item. In the proposed GA based models the process of weight estimation is guided by two inventory performance measures: The minimization of the Total Relevant Cost (TRC) and the maximization of the Inventory Turnover Ratio (ITR). The main advantage of our proposed GA based models with respect to the GA proposed by Guvenir and Erel[26], is that this latter produces a classification of inventory items without considering any inventory control policy. In fact, the process of weight estimation is guided by a subjective function which consists of maximizing the rate of correct classified items with respect to a reference classification provided by the decision maker[26]. We believe that this way of proceeding may produce a biased classification of items, i.e. based on biased item scores, since it is built by using a subjective measure (similarity with respect to a reference classification) and not by using an objective measure such as inventory cost.

3. Genetic Algorithm based classification models

The flowchart of the proposed classification models is presented in Fig. 1. Essentially, two main components constitute the proposed models: (i) The MCDM component - which includes the TOPSIS and the WS methods - to compute the weighted score of each inventory item and, (ii) The Genetic Algorithm (GA) component used to estimate the criteria weights. It's important to note that the optimization process of the GA is guided by two objective functions: The Total Relevant Cost (TRC) and the Inventory Turnover Ratio (ITR) (one function at a time). Hence, four variants of GA based classification models will be proposed and tested in this paper: GA-WS-TRC, GA-TOPSIS-TRC, GA-WS-ITR and GA-TOPSIS-ITR.

The flowchart of the GA based classification models includes essentially three main steps. In the first step, an initial population of criteria weights vectors (or chromosomes[u]) is randomly generated. The second step consists first in computing the weighted score of each inventory item by using two MCDM methods, namely TOPSIS and WS methods. Each of these methods proposes an aggregation rule that combines the criteria weights and the item evaluations on the different criteria to compute an overall score for each item. Then, according to their weighted scores,

[u]A chromosome is composed of a set of genes. In our case a gene is the weight of the criterion j proposed by the k^{th} chromosome w_k, i.e. w_{kj}.

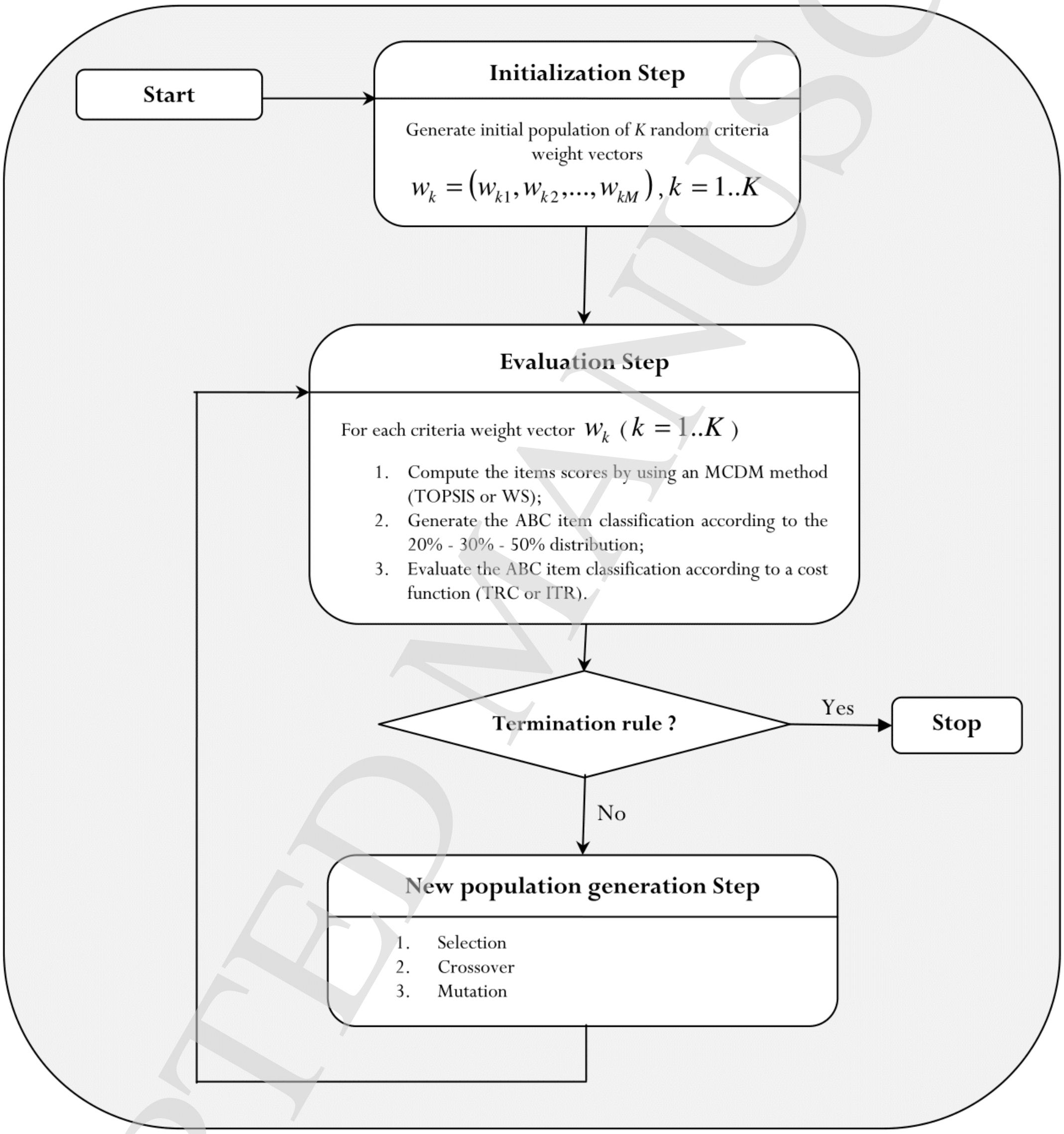

Fig. 1. The flowchart of the proposed GA based classification models.

inventory items are ranked in a descending order and an ABC classification of items is generated by using the commonly used (20% - 30% - 50%) distribution. Finally, the obtained ABC classification of items is evaluated by using two cost functions: Total Relevant Cost (TRC) and the Inventory Turnover Ratio (ITR) (the mathematical formulations of these functions will be detailed in section 3.5). If the termination rule is met, then the classification model is stopped. For this purpose, several terminations rules may be proposed: The maximum number of iterations, the maximum of execution time, no improvement in the fitness function during an interval of time, the average relative change in the fitness function value over a number of consecutive generations is less than a threshold, etc. When the termination rule is not met, a new population of criteria weights vectors is generated (third step) by applying the genetic operators: Selection, crossover and mutation. The selection operator aims to build a new generation of chromosomes by preserving good chromosomes for mating and eliminating the bad ones. The crossover operator combines the features of two "parent" chromosomes to generate two "offspring" chromosomes that may be better than both of the parents. The mutation operator ensures the diversity through successive generations of chromosomes in order to avoid premature convergence to a local optimum. Once the third step is performed, return to the evaluation step (Step 2). In the following subsections, the different components of the GA based classification models will be detailed.

3.1. *Notations*

In order to present the proposed classification models, some mathematical notations needed to be introduced (see Table 2).

3.2. *Genetic Algorithm*

Genetic Algorithms (GAs) are stochastic algorithms based on the genetic evolution mechanism (selection, crossover and mutation) to solve complex and large optimization problems[37]. The main idea of GAs is to start with an initial population of potential solutions (or chromosomes) arbitrarily generated. Then, the relative performance of each chromosome is evaluated through a fitness function [50]. Note that the fitness score assigned to each chromosome represent its ability to 'compete'. Finally, based on the chromosomes performances, a new population is generated using three evolutionary operators: selection, crossover and mutation. The selection operator identifies both the 'good' chromosomes that will be used to generate the new population and the 'bad' chromosomes that will be removed from the current population. The crossover operator exchanges the structures of two "parent" chromosomes in order to form two "offspring" chromosomes that will be involved in the new population. The mutation operator alters arbitrarily the features of one or more chromosomes in order to ensure population diversity. The above three operators are repeated until a stopping criterion is met[53].

3.3. *Solution (or chromosome) representation*

Each solution to a problem consists of a number of decision variables which are encoded as a chromosome. For many years, binary coded solutions have dominated GA research[38]. However, recent GA literature has shown that real-coded (or floating-point) representations have become a common alternative to the binary coding. Indeed, real-coded representation has several benefits[39]: it is faster to manipulate, it has been shown empirically to be more consistent from run to run, it permits much higher precision and it is intuitively closer to the problem space. This last benefit means that real-coded representation is particularly natural when optimization problems involve real decision variables. Since, in this paper, a solution is a vector of the criteria weights which are real numbers, a real-coded representation will be used to describe a chromosome. Therefore, if the ABC classification of inventory items is based on M criteria, a chromosome w_k (k = 1..K, K is the population size) is then defined as $w_k = (w_{k1}, w_{k2}, ..., w_{kM})$, where w_{kj} is a real number and represents the weight of the criterion j according to the k^{th} chromosome. In addition, for each chromosome w_k ($k = 1..$K), the weights w_{kj} ($j = 1..M$) should verify the following condition: $\sum_{j=1}^{M} w_{kj} = 1$.

3.4. *MCDM methods*

Once the chromosomes w_k ($k = 1..K$) are randomly generated (Initialization Step), two MCDM methods (WS and TOPSIS) will be used to compute a score for each item by combining each chromosome w_k with the item evaluations on the different criteria. Based on these weighted scores, an ABC classification of inventory items is performed. Hence, we will have as many ABC classifications of items as the number of chromosomes in the population.

3.4.1. *The weighted Sum (WS) Method*

In this method, the score of each inventory item is computed in two steps. In the first one, the evaluations of items on the different criteria are normalized as follows:

$$\tilde{x}_{ij} = \frac{x_{ij} - \min\limits_{i=1..N} \{x_{ij}\}}{\max\limits_{i=1..N} \{x_{ij}\} - \min\limits_{i=1..N} \{x_{ij}\}} \quad \text{for each benefit criterion } g_j \tag{1}$$

$$\tilde{x}_{ij} = \frac{\max\limits_{i=1..N} \{x_{ij}\} - x_{ij}}{\max\limits_{i=1..N} \{x_{ij}\} - \min\limits_{i=1..N} \{x_{ij}\}} \quad \text{for each cost criterion } g_j \tag{2}$$

where x_{ij} is the evaluation of item a_i according to the criterion g_j, $\tilde{x}_{ij}$ is the normalized evaluation of item a_i according to the criterion g_j and $\max\limits_{i=1..N} \{x_{ij}\}$ (respectively $\min\limits_{i=1..N} \{x_{ij}\}$) is the maximum (respectively the minimum) value of criterion g_j among all inventory items. The second step consists in computing the score of each inventory item a_i according to the chromosome w_k, called S_{ik}^{WS}, as follows:

$$S_{ik}^{WS} = \sum_{j=1}^{M} w_{kj} \ \tilde{x}_{ij} \tag{3}$$

3.4.2. *TOPSIS method*

Technique for Order Preference by Similarity to Ideal Solutions (TOPSIS) was originally developed by Hwang and Youn[40]. This method allows to the decision-maker to rank a set of alternatives evaluated on a set of conflicting and non-commensurable criteria. In practice, TOPSIS has been successfully applied in diverse fields[54] since it is intuitive, easily understood by decision-makers and simple to implement. The principle behind this method is that the 'best' alternative should have the shortest distance from the Positive Ideal Solution (PIS) and the furthest distance from the Negative Ideal Solution (NIS)[20]. The PIS (resp. NIS) is an hypothetical alternative that has the best (resp. worst) evaluations for all considered criteria. Formally, for a problem with N alternatives $a_i(i = 1..N)$ (or inventory items in our case) evaluated on M criteria $g_j(j = 1..M)$, TOPSIS method proceeds according to the following steps[40,41]:

Step 1. Construct the decision matrix X $= (x_{ij})_{N,M}$ in which each item a_i *(i = 1..N)* is evaluated on the criterion g_j *(j = 1..M)*.

Step 2. Determine the criteria weights w_j *(j = 1..M)* such that:

$$\sum_{j=1}^{M} w_j = 1; \tag{4}$$

Step 3. Compute the normalized decision matrix R $= (r_{ij})_{N,M}$:

$$r_{ij} = \frac{x_{ij}}{\sqrt{(\sum_{i=1}^{N} x_{ij}^2)}} \quad \text{for all j = 1..M and i = 1..N.} \tag{5}$$

Step 4. Compute the normalized weighted decision matrix V $= (v_{ij})_{N,M}$:

$$v_{ij} = w_j r_{ij} \quad \text{for all j = 1..M and i = 1..N.} \tag{6}$$

Step 5. Determine the PIS (a^+) and NIS (a^-) as follows:

$$a^+ = \{V_1{}^+, V_2{}^+, V_j{}^+, ..., V_M{}^+\} \tag{7}$$

where

$$V_j{}^+ = \begin{cases} \underset{i=1..N}{max}\{v_{ij}\} & \text{if } g_j \text{ is a Benefit Criterion} \\ \\ \underset{i=1..N}{min}\{v_{ij}\} & \text{if } g_j \text{ is a Cost Criterion} \end{cases} \qquad \text{and}$$

.

$$a^- = \{V_1{}^-, V_2{}^-, V_j{}^-, ..., V_M{}^-\} \tag{8}$$

where

$$V_j{}^- = \begin{cases} \underset{i=1..N}{min}\{v_{ij}\} & \text{if } g_j \text{ is a Benefit Criterion} \\ \\ \underset{i=1..N}{max}\{v_{ij}\} & \text{if } g_j \text{ is a Cost Criterion} \end{cases}$$

.

Step 6. Compute the Euclidean distance that separates each item a_i *(i = 1..N)* from both PIS and NIS:

$$S_i^+ = \sqrt{\sum_{j=1}^{M}(v_{ij} - V_j^+)^2} \quad \text{for all i = 1..N,} \tag{9}$$

$$S_i^- = \sqrt{\sum_{j=1}^{M}(v_{ij} - V_j^-)^2} \quad \text{for all i = 1..N.} \tag{10}$$

Step 7. Compute the relative Separation Measure SM_i of each item a_i *(i = 1..N)* as follows:

$$SM_i = \frac{S_i^-}{S_i^+ + S_i^-} \quad \text{for all i = 1..N.} \tag{11}$$

Finally, the inventory items a_i (i = 1..N) are ranked in a descending order of their separation measure SM_i. Based on this ranking, an ABC classification of items is then generated by using the commonly used items distribution (20% - 30% - 50%). The overall score produced by TOPSIS method for each inventory item is more realistic since it combines two important components: The first measures the strengths of the inventory item (distance from the PIS) whereas the second measures the weaknesses of the inventory item (distance from the NIS).

3.5. *Evaluation functions*

In the proposed GA based models, two evaluation functions - proposed by Tsai and Yeh[28]- are used both to evaluate each ABC classification of inventory items and to guide the process of the criteria weight estimation. The first function consists of minimizing the Total Relevant Cost (TRC) whereas the second aims to maximize the Inventory Turnover Ratio (ITR). These evaluation functions are considered as the most important performance measures that can be used to enhance the effectiveness of the inventory management[28,29]. The formulas of these evaluation functions are expressed as follows[28]:

$$TRC = \sum_z \left(\frac{\sum_{i \in category(z)} S_i}{T_z} + \frac{1}{2} T_z \sum_{i \in category(z)} D_i h_i \right) \tag{12}$$

$$T_z = \sqrt{\frac{2(\sum_{i \in category(z)} S_i)}{\sum_{i \in category(z)} D_i h_i}} \tag{13}$$

$$ITR = \frac{2\sum_{i=1}^{N} D_i}{\sum_z \sum_{i \in category(z)} D_i T_z} \tag{14}$$

where T_z represents the optimal joint replenishment cycle of any item belonging to the category z (z = A, B or C), S_i is the setup cost of item a_i, D_i is the demand per unit time of item a_i and h_i is the holding cost of item a_i. The item setup cost (S_i), the unit holding cost (h_i) and the demand per unit time (D_i) are generated randomly in the following respective intervals[28]: [10-1200], [0-1] and [50-6000].

In our proposed GAs, each chromosome (i.e. a criteria weight vector) is evaluated according to a fitness function which is always to be maximized. Hence, when the evaluation function is to maximize (i.e. the ITR function), then, the fitness function is proportional to the evaluation function (or the same). When, the evaluation function is to minimize (i.e. the TRC function), then, the fitness function should be inversely proportional to the evaluation function.

3.6. *Genetic Algorithm operators*

Once the ABC classification of inventory items is evaluated, the standard genetic operators are then applied: selection, crossover and mutation.

3.6.1. *Selection*

The selection mechanism determines which of the chromosomes in the current generation of the population will be preserved to be parents for reproduction with the aim that next generation will have higher fitness. The main principle behind the selection mechanism is as follows: The better (with higher fitness) is a chromosome, the higher is its chance to be selected as parent in the next generation of the population. Hence, two main aspects should be considered in the conception of any selection operator: (i) good chromosomes (with higher fitness) should have a greater probability of being selected for mating or mutate, and (ii) bad chromosomes should still have small probability of being selected in order to maintain the diversity of chromosomes in next generation of the populations. In the GA literature, many selection schemes have been proposed: the Roulette Wheel Selection (RWS)[42], the Stochastic Universal Sampling (SUS)[43], the Linear Rank Selection (LRS)[44], the Tournament Selection (TOS)[45], etc. A wide set of selection schemes are reviewed in Back et al.[46]. In what follows, we detailed only the Roulette Wheel Selection (RWS) since it is the unique selection scheme used in this paper.

Roulette Wheel Selection (RWS)

In the Roulette Wheel Selection (RWS), also known as stochastic sampling with replacement[42], each chromosome w_k is selected with a probability $P(w_k)$ (or shortly P_k) that is proportional to its fitness value. Let $f_1, f_2, ..., f_K$ be respectively the fitness values of chromosomes $w_1, w_2, ..., w_K$. According to the RWS, the selection probability P_k of chromosome w_k is defined as follows:

$$P(w_k) = P_k = \frac{f_k}{\sum_{k=1}^{K} f_k} \quad \text{for all k} = 1..\text{K} \tag{15}$$

Once the probabilities $P_k(k = 1..K)$ are computed, the population of chromosomes is mapped onto a roulette wheel as follows: each chromosome w_k is represented by a portion of the roulette wheel that is proportional to its P_k (see Fig. 2). Hence, the best chromosomes (with higher fitness) are depicted by largest portions, whereas the worst ones are represented by smaller portions within the roulette wheel. The selection of chromosomes is performed by repeatedly spinning the roulette wheel until a desired number of parents (or chromosomes for mating) is reached. Each time the wheel is spun, it will finally stop and its pointer will be over a portion corresponding to the chromosome to be selected (see Fig. 2). Obviously, this selection mechanism means that good chromosomes will be selected more often than the bad ones. The main advantage of the RWS scheme is that it gives a chance to all chromosomes (without exception) to be selected. Fig. 2 shows an illustrative example of the RWS scheme.

Chromosome identifier	Fitness value	Relative fitness value (%)	Expected number of copies for each chromosome (6* Relative fitness value)	Number of copies for each chromosome obtained by RWS
1	60	30	1.800	2
2	32	16	0.960	1
3	12	6	0.360	0
4	26	13	0.780	1
5	40	20	1.200	1
6	30	15	0.900	1
Total	**200**	**100**	**6**	**6**

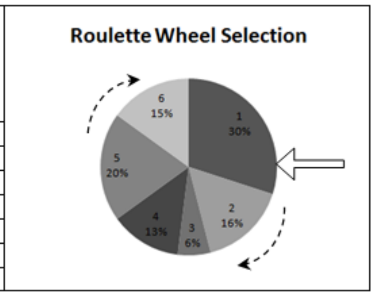

Fig. 2. Application of the RWS scheme.

3.6.2. *The crossover operator*

The crossover operator combines the features of two "parent" chromosomes to produce two new "offspring" chromosomes. The idea behind the crossover operator is that new offsprings may be better that their parents if they take the best characteristics from each of their parents. The crossover operator is not systematically applied to all pairs of chromosomes in the intermediate generation of the population but rather to some randomly selected ones according to a user-definable crossover probability p_c. Once the two "offspring" chromosomes are created, they substitute their parents in the intermediate generation of the population. In real-coded GA many crossover operators have been proposed: Flat Crossover[46], Simple

Crossover[47], Arithmetical Crossover[48], Linear Crossover[47] and many others. A detailed description of crossover operators for RCGAs may be found in Herrera et al.[45]. It is clear that in this paper not all above crossover operators may be applied since they may produce invalid "offspring" chromosomes, i.e. chromosomes $w_k = (w_{k1}, w_{k2}, ..., w_{kM})$ which do not verify the following condition: $\sum_{j=1}^{M} w_{kj} = 1$. Fig. 3 shows that when the simple crossover is applied on two valid parent chromosomes w_1 and w_2, the offspring chromosomes o_1 and o_2 are not necessarily valid.

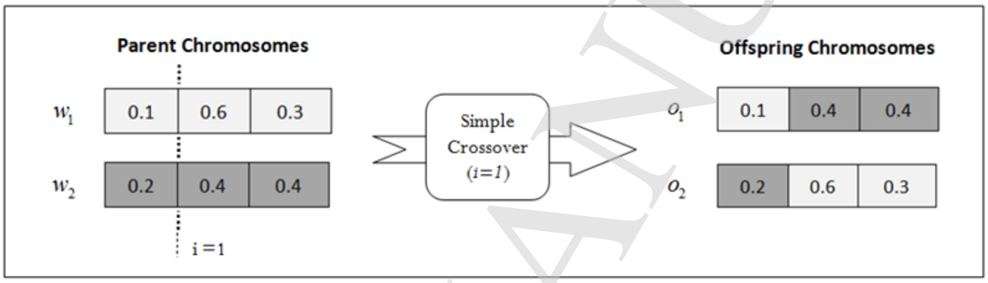

Fig. 3. The simple crossover application.

In this paper, the arithmetical crossover[48] will be used since it ensures the validity of the offspring chromosomes. Based on two parent chromosomes $w_k = (w_{k1}, w_{k2}, ..., w_{kM})$ and $w_{k'} = (w_{k'1}, w_{k'2}, ..., w_{k'M})$ the arithmetical crossover generates the two offspring chromosomes x_1 and x_2 in the following manner:

$$x_1 = (x_{11}, x_{12}, ..., x_{1M}) \text{ where } x_{1j} = \lambda w_{kj} + (1-\lambda) w_{k'j} \text{ for all j = 1..M,} \tag{16}$$

$$x_2 = (x_{21}, x_{22}, ..., x_{2M}) \text{ where } x_{2j} = \lambda w_{k'j} + (1-\lambda) w_{kj} \text{ for all j = 1..M.} \tag{17}$$

where λ is a real number generated randomly in the interval $[0, 1]$. Hence, when the arithmetical crossover is applied on the two parent chromosomes presented in Fig.2, the following valid offspring chromosomes are obtained ($\lambda = 0.2$): $x_1 = (0.18, 0.44, 0.38)$ and $x_2 = (0.12, 0.56, 0.32)$.

3.6.3. *Mutation*

Mutation is an important operator in Genetic Algorithms (GAs) since it preserves and introduces diversity in evolving populations of chromosomes. The mutation operator arbitrarily alters one or more genes of a selected chromosome according to a probability of mutation p_m. As suggested by many authors, this probability should be inversely proportional to the chromosome dimension. Usually, $p_m = \frac{1}{M}$ (M is the criteria number) is used. It is important to note that the mutated chromosome may change entirely from its previous state. Once the mutation step is applied on the intermediate generation of the population, the next generation of the population is finally obtained. In real-coded GA many mutation operators have been proposed: Random mutation[48], Non-uniform mutation [48], Muhlenbeins mutation[50] and many others. A detailed description of mutation operators for RCGAs may be found in Herrera et al.[49]. In this paper, we will use the random mutation[47] for its simplicity and effectiveness. Let us consider a chromosome $w_k = (w_{k1}, w_{k2}, ..., w_{kM})$ and a gene $w_{kj} \in [0, 1]$ to be mutated. The random mutation consists of replacing w_{kj} by a new gene $w_{k'j}$ which is a real number randomly generate in the interval $[0, 1]$. Obviously, the mutated chromosome $w_{k'} = (w_{k1}, w_{k2}, ..., w_{k'j}, ..., w_{kM})$ will be invalid and therefore a normalization step is required, let: $\tilde{w}_{k'} = \left(\frac{w_{k1}}{S}, \frac{w_{k2}}{S}, ..., \frac{w_{k'j}}{S}, ..., \frac{w_{kM}}{S}\right)$ where $S = (w_{k1} + w_{k2} + ... + w_{k'j} + ... + w_{kM})$ and $\tilde{w}_{k'}$ is the normalized mutated chromosome. For example, let $w_k = (0.12, 0.56, 0.32)$ the chromosome to be mutated. Let us assume that the mutation sets the first gene of the chromosome to the value 0.43 (instead of 0.12). Hence, the mutated chromosome $w_{k'} = (0.43, 0.56, 0.32)$ is invalid since the sum of its genes is not equal to 1. Finally, $\tilde{w}_{k'}$ is obtained after a normalization step, let: $\tilde{w}_{k'} = (0.33, 0.43, 0.24)$.

4. Computational Results

To test the performance of the proposed GA based classification models with respect to some existing ABC classification models, a well-known benchmark dataset of inventory items consumed in a Hospital Respiratory Therapy Unit (HRTU), provided by Reid[56], is used. This same data set were used by Ramanathan[8], Zhou and Fan[9], Ng[10], Vencheh[11] and many others. It contains 47 inventory items which are evaluated on three criteria: Annual Dollar Usage (ADU) (\$), Average Unit Cost (AUC) (\$) and Lead Time (LT) (days) (see Table 4). The final ranking of inventory items verifies the commonly used distribution, i.e. the first 10 ranked items are classified in category A (about 21% of total items), the last 23 ranked items are classified in category C (about 49% of total items) and the remaining 14 items are classified in category B (about 30% of total items). Since, in the proposed GA based classification models, two MCDM (TOPSIS and WS methods) are used to compute the items scores and two evaluation functions (TRC and ITR) conduct the GA optimization process to generate the criteria weights, four GA based classification models will be tested in this paper: GA-TOPSIS-TRC, GA-TOPSIS-ITR, GA-WS-TRC and GA-WS-ITR. After several run tests, the best parameter settings of the proposed classification models is as follows: (1) the population size is 50 (2) the crossover probability is 0.9 (3) the mutation probability is 0.7 and (4) the maximum number of iterations, as a termination rule, is 100. The best criteria weight vectors obtained by these four classification models are reported in Table 3. Based on the optimal criteria weight vector generated by the GA-TOPSIS-TRC model, we have reported the computation details of TOPSIS steps in appendix A (see Table 10). It's important to note that the weight vectors generated by our models are in concordance with those obtained by all other existing ABC classification models, i.e. ADU and LT are the most important criteria.

The ABC classification of inventory items obtained by our proposed models and some existing benchmark models such as Traditional-ABC (based only on ADU), AHP-model[18], R-model[8], Ng-model[10], H-model[11] and GA-Classic[26] are reported in Table 4. It's easy to see from Table 4 that 11 of 47 items (about 23% of total items) (items 2, 16, 19, 25, 26, 32, 36, 41, 42, 44 and 46) are identically classified by our proposed GA based models and all other existing benchmark models. Items 21, 24, 33 and 39 are identically classified by all tested models except the Traditional-ABC model. This result may be explained by the fact that Traditional-ABC considers only one criterion (ADU) to classify inventory items whereas all other classification models are based on multiple criteria. Item 40 belongs to: category C by using the Traditional-ABC model, category A by using the AHP-model and category B by using the remaining classification models (i.e. The R-model, the Ng-model, the H-model, the GA-Classic model and our proposed GA based models). This result is expected since the Traditional-ABC is only based on the ADU criterion to classify inventory items (on this criterion item 40 has a low value 103.36) whereas all other classification models consider multiple criteria (item 40 has a high value 6 on an important criterion LT).

We have also computed the percentage of items that are identically classified by our proposed GA based classification models and each benchmark classification model (see Table 5). The lowest percentage is obtained between the Traditional-ABC model and all proposed GA based classification models. This percentage turns around 52.65 % (on average) and it is expected since the Traditional-ABC is only based on the ADU criterion. The highest percentage 91.48 % is obtained with the GA-Classic based model.

In Table 6 we have reported the Total Relevant Cost (TRC) and the Inventory Turnover Ratio (ITR) obtained by all tested classification models. From this table it's is easy to observe that our proposed GA based classification models provide the best values of TRC and ITR: The GA-TOPSIS-TRC model and the GA-WS-TRC have the lowest TRC values in comparison with other models, whereas the GA-TOPSIS-ITR and the GA-WS-ITR have the highest ITR values.

To compare the effectiveness of inventory management of all tested ABC classification models, we perform an additional comparative study based on a Service-Cost performance measure - recently proposed by Babai et al. [7] - to evaluate each ABC inventory classification. For this purpose, we assume that the inventory system is controlled with a reorder point and a reorder quantity, i.e. (s, Q)[7] . Hence, two new functions will be used to perform this comparative study: the Fill Rate (FR) service level and the Total Safety Stock Inventory Cost (C_T).

The total safety stock inventory cost is expressed as:

$$C_T = \sum_{i=1}^{n} h_i sk_i \sigma_i \sqrt{l_i} \tag{18}$$

where the safety factor sk_i for each item i is computed by:

$$sk_i = \phi^{-1}(CSL_i) \tag{19}$$

The fill rate of each item i is given by:

$$FR_i = 1 - \frac{\sigma_i \sqrt{L_i}}{Q_i} G(sk_i) \tag{20}$$

where

$$G(sk_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{{sk_i}^2}{2}} - sk_i[\,1 - \phi(sk_i)] \tag{21}$$

It is important to note that the equation of FR_i is an approximation of the exact equation and this approximation works well for large values of Q_i as it is mentioned in [5].

The Overall fill rate of the inventory system is computed as follows:

$$FR_T = \frac{\sum_{i=1}^{N} FR_i D_i}{\sum_{i=1}^{N} D_i} \tag{22}$$

The results reported in Table 7 show that the proposed GA based classification models (GA-WS-TRC and GA-WS-ITR) provide the lowest total safety stock inventory cost (about 850 in average) whereas the Traditional-ABC classification model obtains the highest one (1048.449). It's important to underline that this result is in concordance with the performance analysis presented in Table 6 since our proposed GA based classification models have obtained the lowest TRC whereas the Traditional-ABC has obtained the highest TRC. Moreover, the average total safety stock inventory cost of all proposed GA based classification models is equal to 927.309 which remains slightly below the cost obtained by the R-model (927.516) (see Table 7). Finally, we measure the overall fill rate - which is the fraction of demands that are satisfied directly from the stock on hand - of each tested classification model (see Table 7). The overall fill rate is an important indicator that reflects the customer satisfaction. It can be seen from Table 7 that the overall fill rate and the total safety stock inventory cost are two conflicting performance measures. The computation details of C_T and FR for the GA-WS-TRC model is reported in Table 8.

An empirical study is also conducted (see Fig. 4) to show the efficiency of safety stock inventory cost vs service level. We have considered three target CSL for categories (A, B, C), namely: (90, 85 and 80%), (95, 90 and 85%)and (99, 95 and 90%)

as proposed in Babai et al.[7]. Table 9 reports the values of total safety stock cost and fill rate under the Normal distribution. This table shows that the proposed GA-WS-ITR has the lowest safety stock cost (510.619), however, the Traditional-ABC has the highest cost (1048.449). The fill rate is high for all tested MCIC models and the proposed GA-TOPSIS-TRC has the highest value (0.993).

Since the Safety Stock Inventory Cost (C_T) and the Fill Rate (FR) service level are inversely proportional, we have carried out an analysis of the combined Service-Cost performance for each MCIC model. This analysis is presented by the efficiency curves for each model (see Fig. 4). These curves report the achieved fill rate as a function of the safety stock cost when the target CLS is varied. For a fixed safety stock cost, the proposed GA-TOPSIS-ITR and GA-TOPSIS-TRC models provide the highest achieved fill rate. It can be seen from Fig. 4, that the GA-TOPSIS-TRC and the GA-TOPSIS-ITR models are very close in terms of the combined service-cost performance and outperform all the remaining tested MCIC models. The efficiency of the proposed models (GA-TOPSIS-TRC and GA-TOPSIS-ITR) is understandable since the inventory items are classified based on the combination of two methodologies: the Artificial Intelligence and the Multicriteria Decision Making. Moreover, the remaining models (R, Ng, H, GA-Classic, AHP and Traditional-ABC) classify the items without considering any inventory control policy.

5. Conclusion

In this paper, we argue the benefits of cross-fertilization of ABC classification models issued from metaheuristics and MCDM methodologies. For this purpose, we have proposed some hybrid classification models based on GA to estimate the criteria weights and two MCDM (WS and TOPSIS) to compute the overall score of each inventory item. It's important to emphasize that the optimization process of the criteria weights estimation in the GA is guided by two crucial inventory performance measures: Minimizing the Total Relevant Cost (TRC) and maximizing the Inventory Turnover Ratio (ITR). Indeed, this feature constitutes the main advantage of our proposed GA based classification models with respect to the classical GA based model proposed by Guvenir and Erel[26] which produces a classification of inventory items without considering any inventory performance measure. Indeed, the classification of inventory items obtained by Guvenir and Erel[26] model seeks to maximize the rate of items that are correctly classified with respect to a reference classification of inventory items provided by the decision maker. As stated in Lopez et al.[31], this subjective reference classification is not necessarily built on the basis of the true performance of inventory items. Sometimes decision-makers, due to their experience and knowledge, built reference classification based on their preference; this situation generates familiarity-biased classifications which lead to unnecessary costs due to the non-homogeneity of categories[31]. To compare the performance of our proposed GA based MCIC models with respect to some existing

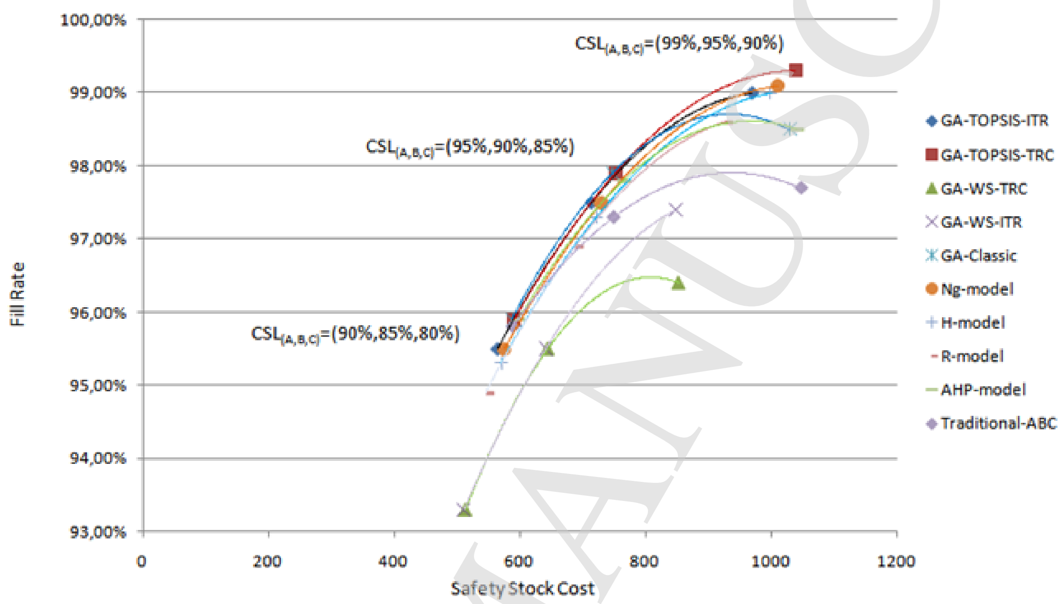

Fig. 4. Efficiency curves for the testes MCIC models under the Normal Distribution.

classification models, a benchmark data set of 47 items consumed in a Hospital Respiratory Therapy Unit (HRTU) is used. Even if the purpose of the comparison might be seen limited since only one dataset is used (which constitutes a limitation of the present work), the computational results showed that our GA based models outperformed all other existing classification models according to the TRC and the ITR functions. Furthermore, the additional performance analysis has also shown the effectiveness of our GA based MCIC models in inventory management since they have obtained the best (on average) Total Safety Stock Inventory Cost (C_T) (927.163) and a competitive overall Fill Rate (FR) (0.965). Some interesting avenues for further research may be exploited: (i) Considering the multi-objective variant of the GA for the ABC inventory classification; (ii) Developing new ABC classification models that consider both quantitative and qualitative criteria since in our proposed GA based models only quantitative criteria are considered (which constitutes an other limitation of the present work); (iii) Considering more than one dataset in the comparative studies and (iiii) Designing a computer-based decision support system which contains the proposed models.

References

1. C.W. Chu, G. S. Liang, and C.T. Liao, Controlling inventory by combining ABC analysis and fuzzy classification, *Computers & Industrial Engineering* **55** (2008) 841–851.
2. T.W., Best practice in Inventory Management, Second Edition , *Elsevier Science Ltd* (2002).
3. B. E. Flores and D. C. Whybark, Implementing multiple criteria ABC analysis, *Journal of Operations Management* **7** (1987) 79–85.
4. F. Lolli, A. Ishizaka, and R. Gamberini, New AHP-based approaches for multi-criteria inventory classification, *Int. J.Production Economics* 156 (2014) 62–74.
5. H. Kaabi, K. Jabeur, and L. Enneifer, Learning criteria weights with TOPSIS method and continuous VNS for multi-criteria inventory classification, *Electronic Notes in Discrete Mathematics* **47** (2015) 197–204.
6. J. Liu, X. Lia, W. Zhao, and N. Yang, A classification approach based on the outranking model for multiple criteria ABC analysis, *OMEGA* 61 (2016) 19–34.
7. M. Z. Babai, T. Ladhari and I. Lajili, On the inventory performance of multi-criteria classification methods: empirical investigation, *International Journal of Production Research* **53** (2014) 279–290.
8. R. Ramanathan, ABC inventory classification with multiple-criteria using weighted linear optimization, *Computers & Operations Research* **33** (2006) 695–700.
9. P. Zhou and L. Fan, A note on multi-criteria ABC inventory classification using weighted linear optimization, *European Journal of Operational Research* **182** (2007) 1488–1491.
10. W.L. Ng, A simple classifier for multiple criteria ABC analysis, *European Journal of Operational Research* **177** (2007) 344-353.
11. A. H. Vencheh, An improvement to multiple criteria ABC inventory classification, *European Journal of Operational Research* **21** (2010) 962–965.
12. Y. Fu, K. K. Laic, Y. Miaoe, and John W.K. Leung, A distance-based decision-making method to improve multiple criteria ABC inventory classification, *International Transactions In Operational Research* (2015) 1–10.
13. T. Ladhari, M.Z. Babai, and I. Laajili, Multi-criteria inventory classification: new consensual procedures, *IMA Journal of Management Mathematics* (2015) 1–17.
14. H. Kaabi and K. Jabeur, A New Hybrid Weighted Optimization Model for Multi Criteria ABC Inventory Classification, *Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA, Advances in Intelligent Systems and Computing* (2016) 261–270.
15. A. Baykasoglu, K. Subulan, and F. S. Karaslan, A new fuzzy linear assignment method for multi-attribute decision making with an application to spare parts inventory classification, *Applied Soft Computing* **42** (2016) 1–17.
16. S. Zheng, Y. Fu, K. K Lai, and L. Liang, An Improvement to Multiple Criteria ABC Inventory Classification Using Shannon Entropy, *Journal of Systems Science and Complexity* **30** (2017) 857-865.
17. T. L. Saaty, The analytic hierarchy process in conflict management, *International Journal of Conflict Management* **1** (1990) 47-68.
18. B. E. Flores, D. L. Olson, and V.K. Dorai, Management of Multicriteria inventory classification, *Mathematical and Computer Modelling* **16** (1992) 71–82.
19. F.Y. Partovi and J. Burton, Using the analytic hierarchy process for ABC analysis, *International Journal of Operations & Production Management* **13** (1993) 29–44.
20. A. Bhattacharya, B. Sarkar, and S.K. Mukherjee, Distance-based consensus method for ABC analysis, *International Journal of Production Research* **45** (2007) 3405–3420.

21. A.H Vencheh and A. Mohamadghasemi, A fuzzy AHP-DEA approach for multiple criteria ABC inventory classification, *Expert Systems with Applications* **38** (2011) 3346–3352.
22. M. K. Ghorabaee, E. K. Zavadskas, L. Olfat, and Z. Turskis, Multi-Criteria Inventory Classification Using a New Method of Evaluation Based on Distance from Average Solution (EDAS), *INFORMATICA* **26** (2015) 435–451.
23. F. Arikan and S. Citak, Multiple Criteria Inventory Classification in an Electronics Firm, *International Journal of Information Technology & Decision Making* **16** (2017) 1–17.
24. Z. Li, X. Wu, F. Liu, Y. Fu and K. Chen, Multicriteria ABC inventory classification using acceptability analysis, *International Transactions in Operational Research* **00** (2017) 1–14.
25. G. Kou, D. Ergu, C. Lin and Y. Chen, Pairwise comparison matrix in multiple criteria decision making, *Technological and Economic Development of Economy***5** (2016) 738–765.
26. H. A. Guvenir and E. Erel, Multicriteria inventory classification using a genetic algorithm, *European Journal of Operational Research* **105** (1998) 29–37.
27. F. Y. Partovi and M. Anandarajan, Classifying inventory using an artificial neural network approach, *Computers & Industrial Engineering* **41** (2002) 389–404.
28. C. H. Tsai and S.W. Yeh, A multiple objective particle swarm optimization approach for inventory classification, *Int. J. Production Economics* **114** (2011) 656–666.
29. D. Mohammaditabar, S.H. Ghodsypour, and C. OB́rien, Inventory control system design by integrating inventory classification and policy selection, *Int. J.Production Economics* **140** (2012) 655–659.
30. M. C. Yu, Multi-criteria ABC analysis using artificial-intelligence-based classification techniques, *Expert Systems with Applications* **38** (2011) 3416–3421.
31. D. Lopez-Soto, S. Yacout, and F. Angel-Bello, Root cause analysis of familiarity biases in classification of inventory items based on logical patterns recognition, *Computers & Industrial Engineering* 93 (2016) 121-130.
32. B. Roy, ELECTRE III: Un algorithme de classements fondé sur une représentation floue des préférences en présence de critéres multiples, *Cahiers du CERO* 20 (1978) 3–24.
33. B. E. Flores and D. C. Whybark, Implementing multiple criteria ABC analysis, *Journal of Operations Management* 7 (1987) 79-85.
34. Y. Chen, K. W. Li, D. M. Kilgour and K. W. Hipel, A case-based distance model for multiple criteria ABC analysis, *Computers & Operations Research* 35 (2008) 776-796.
35. B. Soylu and B. Akyol, Multi-criteria inventory classification with reference items, *Computers & Industrial Engineering* 69 (2014) 12-20.
36. H. Kartal, A. Oztekin, A. Gunasekaran and F. Cebi, An integrated decision analytic framework of machine learning with multi-criteria decision making for multi-attribute inventory classification, *Computers & Industrial Engineering* 101 (2016) 599-613.
37. J. H. Holland, Adaptation in Natural and Artificial Systems, *Cambridge, MA: MIT Press, Second edition (First edition, 1975)* (1975/1992).
38. J. R. Koza, Genetic Programming, *The MIT press, Cambridge* (1991).
39. P. J. Fleming and R.C. Purshouse, Genetic algorithms in control systems engineering, *Research Report N0 789, University Sheffield* (2001).
40. C. L. Hwang and K. Yoon, Multiple attribute decision making-methods and applications: a state-of-the-art survey, *Multiple Attribute Decision Making* **186** (1981).
41. G. Kou, Y. Lu, Y. Peng and Y. Shi, Evaluation of Classification Algorithms using MCDM and Rank Correlation, *International Journal of Information Technology &*

Decision Making **11** (2012) 197–225.
42. D. E. Goldberg and J. H. Holland, Genetic Algorithms and Machine Learning, *Machine Learning* **3** (1988) 95–99.
43. J. E. Baker, Reducing bias and inefficiency in the selection algorithm, *of the Second International Conference on Genetic Algorithms (L. Erlbaum Associates, Hillsdale, MA)* (1987) 14–21.
44. J. E. Baker, Adaptive selection methods for genetic algorithms, *Proceedings of an International Conference on Genetic Algorithms and Their Applications (L. Erlbaum Associates, Hillsdale, MA)* (1985) 100–111.
45. A. Brindle, Genetic algorithms for function optimization, *Doctoral dissertation and Technical Report TR81-2. Edmonton: University of Alberta, Department of Computer Science* (1981).
46. T. Bck, F. Hoffmeister, and H.P Schwefel, Extended Selection Mechanisms in Genetic Algorithms, *Proceeding of the Fourth International Conference on Genetic Algorithms, R. Belew and L.B. Booker (Eds.) (Morgan Kaufmmann, San Mateo)* (1991) 92–99.
47. A. Wright, Genetic Algorithms for Real Parameter Optimization *Foundations of Genetic Algorithms* (1991) 205–218.
48. Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs *Springer Verlag, New York* (1992).
49. F. Herrera, M. Lozano, and J. L. Verdegay, Tackling real-coded genetic algorithms: Operators and tools for behavioral analysis, *Artificial Intelligence Review* **12** (1998) 265–319.
50. H. Muhlenbein and D. Schlierkamp-Voosen, Predictive Models for the Breeder Genetic Algorithm I: Continuous Parameter Optimization, *Evolutionary Computation* **1** (1993) 25–49.
51. R. H. Teunter, M.Z. Babai, and A.A. Syntetos, ABC classification: Service levels and inventory costs, *Production and operations management* **19** (2009) 343–352.
52. M. Mitchell, An Introduction to Genetic Algoritms, *MIT Press Cambridge*, MA, USA (1998).
53. E. A. Hansen and S. Zilberstein, A heuristic search algorithm that finds solutions with loops, *Artificial Intelligence* **129** (2001) 35–62.
54. D. Yong, Plant location selection based on fuzzy TOPSIS, *International Journal of Advanced Manufacturing Technology* **28** (2006) 839–844.
55. J.R. Figueira, V. Mousseau, and B. Roy, Multiple Criteria Decision Analysis, *International Series in Operations Research Management Science* **233** (2016).
56. R. A. Reid, The ABC method in hospital inventory management: a practical approach, *Production and Inventory Management* **28** (1987), 67–70.

Appendix A. Computation details of TOPSIS steps for the GA-TOPSIS-TRC model

The computation details of TOPSIS method for the GA-TOPSIS-TRC model are reported in Table 10. $A\tilde{D}U$, $A\tilde{U}C$ and $\tilde{LT}$ represent respectively the normalized values of criteria ADU, AUC and LT. All above criteria are considered to be benefit criteria. For this illustration, the criteria weight vector obtained by the GA-TOPSIS-TRC will be used $(0.41, 0.17, 0.42)$.

Table 1. Comparison of key features among MCIC fields.

References	Model formulation	Evaluation function	Nature of criteria weights	Type of criteria	Inventory classification based on	Service-Cost Analysis	Criteria used to perform the ABC classification
Classification Models based on Mathematical Programming (MP) Methodology							
Ramanathan, (2006)[8]	Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC, LT and CF
Zhou and Fan, (2007)[9]	Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC, LT and CF
Ng[10], (2007)	Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC and LT
Vencheh[11], (2009)	Non Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC and LT
Fu et al., (2015)[12]	Non Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC and LT
Ladhari et al., (2015)[13]	Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC and LT
Kaabi and Jabeur, (2016)[14]	Linear	Single	Objective	Quantitative	Scores	Yes	ADU, AUC and LT
Baykasoglu et al., (2016)[15]	Linear	Single	Subjective	Quantitative&Qualitative	Scores	No	AD[c], UC[d], AC[e]
Zheng et al., (2017)[16]	Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC and LT
Babai et al., (2014)[7]	Linear	Single	Objective	Quantitative	Inventory costs	Yes	ADU, AUC and LT
Soylu et al., (2014)[35]	Linear	Single	Objective	Quantitative	Scores	Yes	ADU, AUC and LT
Teunter et al., (2009)[51]	Non Linear	Single	Objective	Quantitative	Inventory costs	Yes	ADU, AUC and LT
Classification Models based on Multicriteria Decision Making (MCDM) Methodology							
Flores et al., (1992)[18]	Linear	Single	Subjective	Quantitative	Scores	No	ADU, AUC, LT and CF
Partovi and Burton, (1993)[19]	Linear	Single	Subjective	Quantitative&Qualitative	Scores	No	ADU, AUC, LT and CF
Bhattacharya et al., (2007)[20]	Non Linear	Single	Subjective	Quantitative	Scores	No	AUC, LT, CR[f], PI[g] and SC[h]
Vencheh and Mohamadghasemi, (2011)[21]	Non Linear	Single	Subjective	Qualitative	Scores	No	ADU, AUC, LT and LWS[i]
Ghorabaee et al., (2015)[22]	Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC and LT
Arikan and Citak, (2017)[23]	Linear	Single	Subjective	Quantitative	Scores	No	AUC, LT, CF and D
Li et al., (2017)[24]	Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC and LT
Flores and Whybark, (1987)[33]	Linear	Single	Subjective	Quantitative&Qualitative	Scores	No	ADU and CF
Chen et al.[34], (2008)	Non Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC, LT and CF
Classification Models based on both Metaheuristics and Artificial Intelligence (AI) Methodologies							
Guvenir and Erel, (1998)[26]	Non Linear	Single	Objective	Quantitative	Scores	No	UP[j], NR[k], LT, D, S[l], Su[m], R[n], OSR[o]
Partovi and Anandarajan, (2002)[27]	Non Linear	Single	Objective	Quantitative	Scores	No	UP, OC[p], LT and D
Tsai and Yeh, (2011)[28]	Non Linear	Multiple	Objective	Quantitative	Scores	No	SUC[q], HC[r], SOC[s] and D
Mohammaditabar et al., (2012)[29]	Non Linear	Single	Objective	Quantitative	Inventory costs	No	ADU, AUC and LT
Yu, (2011)[30]	Non Linear	Single	Objective	Quantitative	Scores	No	ADU, AUC, LT and CF
Lopez et al., (2016)[31]	Non Linear	Single	Subjective	Quantitative&Qualitative	Scores	No	UP, OC, LT and D
Hybrid Classification Models							
Lolli et al., (2014)[4]	Non Linear	Single	Subjective	Quantitative	Scores	No	ADU, AUC, LT and CF
Kaabi et al., (2015)[5]	Non Linear	Single	Objective	Quantitative	Inventory costs	No	ADU, AUC and LT
Liu et al., (2015)[6]	Non Linear	Single	Subjective	Quantitative	Inventory costs	No	ADU, AUC, LT and T
Kartal et al., (2016)[36]	Non Linear	Single	Subjective	Mixed	Scores	No	AUC, CF, D and US[t]

Table 2. Different notations of the proposed models.

Notation	Explanation
N	The number of inventory items
M	The number of criteria
K	Population size
i	The inventory item index $i = 1..N$
$A = \{a_1, a_2, ..., a_N\}$	The items set where a_i is the i^{th} item
j	The criteria index $j = 1..M$
$G = \{g_1, g_2, ..., g_M\}$	The criteria set where g_j is the j^{th} criterion
k	The chromosome index $k = 1..K$
$w_k = (w_{k1}, w_{k2}, ..., w_{kM})$	The k^{th} chromosome. It corresponds to the vector of criteria weight
w_{kj}	The weight of the criterion j proposed by the k^{th} chromosome
x_{ij}	The evaluation of the inventory item a_i according to the criterion g_j
$\tilde{x}_{ij}$	The normalized evaluation of the inventory item a_i according to the criterion g_j
D_i	Demand of item a_i ($D_i = \frac{AnnualDollarUsage(ADU)}{AverageUnitCost(AUC)}$)
σ_i	Standard deviation of the demand of item a_i ($\sigma_i = 0.5 \times D_i$)
h_i	Inventory holding cost of item a_i (20% of the average unit cost)
W_i	Unit ordering cost of item a_i (in our case W_i is equal to 1)
L_i	Lead Time of item a_i
Q_i	Order quantity of item a_i, ($Q_i = \sqrt{\frac{2W_iD_i}{h_i}}$)
FR_i	Fill rate of item a_i
FR_T	Overall fill rate of the inventory system
C_T	Total safety stock inventory cost
CSL_i	Cycle service level of item a_i, (99% if $a_i \in$ category A, 95% if $a_i \in$ category B, 90% if $a_i \in$ category C)
sk_i	Safety factor of item a_i
$\phi(.)$	Standard normal probability distribution function
$G(.)$	Loss function of the standard normal distribution

Table 3. Best obtained criteria weights vectors by our proposed GA based models.

GA model	Criteria weight	Evaluation function
GA-TOPSIS-TRC	(0.41 0.17 0.42)	53769.29
GA-WS-TRC	(0.5 0.03 0.47)	52562.23
GA-TOPSIS-ITR	(0.32 0.06 0.62)	2.48
GA-WS-ITR	(0.3 0.17 0.53)	2.42

Table 4. Comparison of ABC classification obtained by various benchmark models and our proposed GA based classification models.

SKUs	ADU	AUC	LT	Traditional-ABC	AHP-model	R-model	Ng-model	H-model	GA-Classic	GA-TOPSIS-TRC	GA-TOPSIS-ITR	GA-WS-TRC	GA-WS-ITR
1	5840.64	49.92	2	A	C	A	A	A	A	A	A	C	C
2	5670	210	5	A	A	A	A	A	A	A	A	A	A
3	5037.12	23.76	4	A	B	A	A	A	A	A	A	C	C
4	4769.56	27.73	1	A	C	B	A	A	A	A	A	C	C
5	3478.8	57.98	3	A	C	B	A	A	A	A	A	B	C
6	2936.67	31.24	3	A	C	C	A	B	A	A	B	C	C
7	2820	28.2	3	A	C	C	B	B	A	A	B	C	C
8	2640	55	4	A	B	B	B	B	A	A	A	B	B
9	2423.52	73.44	6	A	A	A	A	A	A	A	C	A	A
10	2407.5	160.5	4	A	B	B	A	A	B	A	B	A	A
11	1075.2	5.12	2	B	C	C	C	C	C	C	C	A	A
12	1043.5	20.87	5	B	B	B	B	B	B	B	B	C	C
13	1038	86.5	7	B	A	A	A	A	A	B	A	A	A
14	883.2	110.4	5	B	A	B	B	A	B	B	B	A	A
15	854.4	71.2	3	B	C	C	C	C	C	C	C	B	B
16	810	45	3	B	C	C	C	C	C	C	C	C	C
17	703.68	14.66	6	B	C	C	C	C	B	B	B	C	B
18	594	49.5	6	B	A	A	B	B	B	B	B	B	A
19	570	47.5	5	B	B	B	B	B	B	B	B	B	B
20	467.6	58.45	4	B	B	C	C	C	C	C	C	B	B
21	463.6	24.4	4	B	C	C	C	C	C	C	C	C	C
22	455	65	4	B	B	C	C	C	C	C	C	B	B
23	432.5	86.5	4	B	B	C	B	B	C	C	C	A	B
24	398.4	33.2	3	B	C	C	C	C	C	C	C	C	C
25	370.5	37.01	1	C	C	C	C	C	C	C	C	C	C
26	338.4	33.84	3	C	C	C	C	C	C	C	C	C	C
27	336.12	84.03	1	C	C	C	C	C	C	C	A	A	C
28	313.6	78.4	6	C	A	A	B	B	B	B	B	A	A
29	268.68	134.34	7	C	A	A	A	A	C	B	A	A	A
30	224	56	1	C	C	C	C	C	C	C	C	B	C
31	216	72	5	C	B	B	B	B	B	B	B	B	A
32	212.08	53.02	2	C	C	C	C	C	C	C	C	C	C
33	197.92	49.48	5	C	B	B	B	B	B	B	B	B	B
34	190.89	7.07	7	C	A	A	B	B	B	B	A	C	B
35	181.8	60.6	3	C	C	C	C	C	C	C	C	B	C
36	163.28	40.82	3	C	C	C	C	C	C	C	C	C	C
37	150	30	5	C	B	B	C	C	B	C	C	C	B
38	134.8	67.4	3	C	C	C	C	C	C	C	C	B	C
39	119.2	59.6	5	C	B	B	B	B	B	B	B	B	B
40	103.36	51.68	6	C	A	B	B	B	B	B	B	B	B
41	79.2	19.8	2	C	C	C	C	C	C	C	C	C	C
42	75.4	37.7	2	C	C	C	C	C	C	C	C	C	C
43	59.78	29.89	5	C	B	B	C	C	C	C	C	C	B
44	48.3	48.3	3	C	C	C	C	C	C	C	C	C	C
45	34.4	34.4	7	C	A	A	B	B	B	B	B	C	B
46	28.8	28.8	3	C	C	C	C	C	C	C	C	C	C
47	25.38	8.46	5	C	B	B	C	C	C	C	C	C	C

Table 5. Classification accuracy of our proposed GA models vs. benchmark models.

GA based models	Benchmark models						
	Traditional-ABC (%)	AHP-model (%)	R-model (%)	Ng-model(%)	H-model(%)	GA-Classic (%)	Average (%)
GA-TOPSIS-TRC	65.95	53.19	65.95	87.23	82.97	91.48	74.46
GA-TOPSIS-ITR	53.19	57.49	70.21	82.97	85.10	85.10	72.34
GA-WS-TRC	44.68	65.95	57.44	57.44	59.57	46.8	55.31
GA-WS-ITR	46.80	74.46	63.82	63.82	63.82	57.44	61.69
Average	52.65	62.76	64.35	72.86	72.86	70.20	

Table 6. Comparison of TRC and ITR obtained by all tested MCIC models.

ABC MCIC models	Total Relevant Cost	Inventory Turnover Ratio
Traditional-ABC	56245.28	2.29
AHP-model	55121.55	2.33
R-model	53866.63	2.36
Ng-model	54351.46	2.37
H-model	54211.38	2.35
GA-Classic	55195.29	2.31
GA-TOPSIS-TRC	53769.29	2.42
GA-TOPSIS-ITR	53119.50	2.48
GA-WS-TRC	52562.23	2.38
GA-WS-ITR	52475.29	2.40

Table 7. Total safety stock inventory cost and fill rate results obtained by all tested MCIC models.

ABC MCIC models	Safety Stock Inventory Cost	Fill Rate
Traditional-ABC	1048.449	0.977
AHP-model	1040.965	0.985
R-model	927.516	0.986
H-model	999.892	0.990
Ng-model	1011.007	0.991
GA-Classic	1030.450	0.985
GA-TOPSIS-TRC	1038.537	0.974
GA-TOPSIS-ITR	969.261	0.968
GA-WS-TRC	853.256	0.974
GA-WS-ITR	847.600	0.944

Table 8. Computation details of C_T and FR for the GA-WS-TRC model.

SKUs	Category	D_i	Q_i	$\sigma_i\sqrt{l_i}$	h_i	$CSL(i)$	sk_i	$G(sk_i)$	FR_i	Satisfied demand	Inventory cost
2	A	27		1.582	42.000	0.990	2.326	0.003	0.995	26.872	154.594
9	A	33	2.120	2.118	14.688	0.990	2.326	0.003	0.997	32.888	72.385
10	A	15	0.967	0.786	32.100	0.990	2.326	0.003	0.997	14.959	58.711
11	A	210	20.252	7.783	1.024	0.990	2.326	0.003	0.999	209.727	18.541
13	A	12	1.178	0.832	17.300	0.990	2.326	0.003	0.998	11.971	33.487
14	A	8	0.851	0.469	22.080	0.990	2.326	0.003	0.998	7.985	24.081
23	A	5	0.760	0.262	17.300	0.990	2.326	0.003	0.999	4.994	10.547
27	A	4	0.690	0.105	16.806	0.990	2.326	0.003	0.999	3.998	4.098
28	A	4	0.714	0.257	15.680	0.990	2.326	0.003	0.999	3.995	9.366
29	A	2	0.386	0.139	26.868	0.990	2.326	0.003	0.999	1.998	8.668
5	B	60	3.217	2.724	11.596	0.950	1.645	0.021	0.982	58.939	51.948
8	B	48	2.954	2.516	11.000	0.950	1.645	0.021	0.982	47.146	45.521
15	B	12	1.298	0.545	14.240	0.950	1.645	0.021	0.991	11.895	12.758
18	B	12	1.557	0.770	9.900	0.950	1.645	0.021	0.990	11.876	12.544
19	B	12	1.589	0.703	9.500	0.950	1.645	0.021	0.991	11.889	10.988
20	B	8	1.170	0.419	11.690	0.950	1.645	0.021	0.993	7.940	8.063
22	B	7	1.038	0.367	13.000	0.950	1.645	0.021	0.993	6.948	7.845
30	B	4	0.845	0.105	11.200	0.950	1.645	0.021	0.997	3.990	1.931
31	B	3	0.645	0.176	14.400	0.950	1.645	0.021	0.994	2.983	4.164
33	B	4	0.899	0.234	9.896	0.950	1.645	0.021	0.995	3.978	3.815
35	B	3	0.704	0.136	12.120	0.950	1.645	0.021	0.996	2.988	2.715
38	B	2	0.545	0.091	13.480	0.950	1.645	0.021	0.997	1.993	2.013
39	B	2	0.579	0.117	11.920	0.950	1.645	0.021	0.996	1.992	2.298
40	B	2	0.622	0.128	10.336	0.950	1.645	0.021	0.996	1.991	2.183
1	C	117	4.841	4.336	9.984	0.900	1.282	0.047	0.958	112.039	55.483
3	C	212	9.458	11.126	4.746	0.900	1.282	0.047	0.944	200.446	67.670
4	C	172	7.876	4.508	5.546	0.900	1.282	0.047	0.973	167.339	32.038
6	C	94	5.486	4.267	6.248	0.900	1.282	0.047	0.963	90.542	34.167
7	C	100	5.955	4.539	5.640	0.900	1.282	0.047	0.964	96.391	32.809
12	C	50	4.895	2.930	4.174	0.900	1.282	0.047	0.972	48.583	15.673
16	C	18	2.000	0.817	9.000	0.900	1.282	0.047	0.981	17.652	9.424
17	C	48	5.722	3.081	2.932	0.900	1.282	0.047	0.975	46.776	11.578
21	C	19	2.790	0.996	4.880	0.900	1.282	0.047	0.983	18.679	6.228
24	C	12	1.901	0.545	6.640	0.900	1.282	0.047	0.986	11.837	4.635
25	C	10	1.645	0.262	7.402	0.900	1.282	0.047	0.992	9.935	2.489
26	C	10	1.719	0.454	6.768	0.900	1.282	0.047	0.987	9.875	3.937
32	C	4	0.869	0.148	10.604	0.900	1.282	0.047	0.992	3.968	2.015
34	C	27	6.180	1.872	1.414	0.900	1.282	0.047	0.986	26.613	3.392
36	C	4	0.990	0.182	8.164	0.900	1.282	0.047	0.991	3.965	1.900
37	C	5	1.291	0.293	6.000	0.900	1.282	0.047	0.989	4.946	2.253
41	C	4	1.421	0.148	3.960	0.900	1.282	0.047	0.995	3.980	0.752
42	C	2	0.728	0.074	7.540	0.900	1.282	0.047	0.995	1.990	0.716
43	C	2	0.818	0.117	5.978	0.900	1.282	0.047	0.993	1.986	0.898
44	C	1	0.455	0.045	9.660	0.900	1.282	0.047	0.995	0.995	0.562
45	C	1	0.539	0.069	6.880	0.900	1.282	0.047	0.994	0.994	0.611
46	C	1	0.589	0.045	5.760	0.900	1.282	0.047	0.996	0.996	0.335
47	C	3	1.991	0.197	1.692	0.900	1.282	0.047	0.995	3.339	0.426
		1415								1379.802	853.256

Table 9. Safety Stock Cost vs Fill Rate of the MCIC methods: an empirical results.

Target CLS for categories (A, B, C)	(90%,85%,80%)		(95%, 90%, 85%)		(99%, 95%, 90%)	
	Safety Stock Cost	Fill Rate	Safety Stock Cost	Fill Rate	Safety Stock Cost	Fill Rate
GA-TOPSIS-ITR	564.406	0.955	713.71	0.975	969.439	0.99
GA-TOPSIS-TRC	591.225	0.959	752.18	0.979	1038.769	0.993
GA-WS-TRC	514.109	0.933	644.949	0.955	853.256	0.964
GA-WS-ITR	510.619	0.933	640.77	0.955	847.614	0.974
GA-Classic	588.184	0.959	747.758	0.979	1030.45	0.985
R-model	546.811	0.949	689.262	0.969	927.516	0.986
H-model	570.21	0.953	721.586	0.973	999.892	0.99
Ng-model	575.328	0.955	729.391	0.975	1011.007	0.991
AHP-model	598.325	0.959	765.741	0.978	1040.965	0.985
Traditional-ABC	589.551	0.958	750.074	0.973	1048.449	0.977

Table 10. Computation details of TOPSIS steps for the GA-TOPSIS-TRC model.

Item	ADU	AUC	LT	$\tilde{ADU}$	$\tilde{AUC}$	$\tilde{LT}$	V_{ADU}	V_{AUC}	V_{LT}	S^+	S^-	SM_i	Category
1	5840.64	49.92	2	0.448	0.110	0.068	0.184	0.019	0.028	0.093	0.184	0.665	A
2	5670	210	5	0.435	0.461	0.170	0.178	0.078	0.071	0.029	0.201	0.874	A
3	5037.12	23.76	4	0.386	0.052	0.136	0.158	0.009	0.057	0.085	0.163	0.657	A
4	4769.56	27.73	1	0.366	0.061	0.034	0.150	0.010	0.014	0.114	0.149	0.567	A
5	3478.8	57.98	3	0.267	0.127	0.102	0.109	0.022	0.043	0.109	0.114	0.510	A
6	2936.67	31.24	3	0.225	0.069	0.102	0.092	0.012	0.043	0.127	0.096	0.432	A
7	2820	28.2	3	0.216	0.062	0.102	0.089	0.011	0.043	0.130	0.093	0.417	A
8	2640	55	4	0.202	0.121	0.136	0.083	0.021	0.057	0.124	0.094	0.433	A
9	2423.52	73.44	6	0.186	0.161	0.203	0.076	0.027	0.085	0.120	0.107	0.471	A
10	2407.5	160.5	4	0.185	0.352	0.136	0.076	0.060	0.057	0.117	0.104	0.469	A
11	1075.2	5.12	2	0.082	0.011	0.068	0.034	0.002	0.028	0.183	0.036	0.164	C
12	1043.5	20.87	5	0.080	0.046	0.170	0.033	0.008	0.071	0.169	0.066	0.280	B
13	1038	86.5	7	0.080	0.190	0.237	0.033	0.032	0.100	0.158	0.096	0.378	B
14	883.2	110.4	5	0.068	0.242	0.170	0.028	0.041	0.071	0.163	0.074	0.313	B
15	854.4	71.2	3	0.065	0.156	0.102	0.027	0.027	0.043	0.175	0.046	0.208	C
16	810	45	3	0.062	0.099	0.102	0.025	0.017	0.043	0.179	0.040	0.185	C
17	703.68	14.66	6	0.054	0.032	0.203	0.022	0.005	0.085	0.178	0.074	0.295	B
18	594	49.5	6	0.046	0.109	0.203	0.019	0.018	0.085	0.176	0.075	0.300	B
19	570	47.5	5	0.044	0.104	0.170	0.018	0.018	0.071	0.179	0.062	0.256	B
20	467.6	58.45	4	0.036	0.128	0.136	0.015	0.022	0.057	0.183	0.049	0.212	C
21	463.6	24.4	4	0.036	0.054	0.136	0.015	0.009	0.057	0.188	0.045	0.195	C
22	455	65	4	0.035	0.143	0.136	0.014	0.024	0.057	0.183	0.050	0.215	C
23	432.5	86.5	4	0.033	0.190	0.136	0.014	0.032	0.057	0.181	0.054	0.229	C
24	398.4	33.2	3	0.031	0.073	0.102	0.013	0.012	0.043	0.192	0.033	0.145	C
25	370.5	37.01	1	0.028	0.081	0.034	0.012	0.014	0.014	0.203	0.016	0.074	C
26	338.4	33.84	3	0.026	0.074	0.102	0.011	0.013	0.043	0.194	0.032	0.142	C
27	336.12	84.03	1	0.026	0.184	0.034	0.011	0.031	0.014	0.199	0.031	0.135	C
28	313.6	78.4	6	0.024	0.172	0.203	0.010	0.029	0.085	0.181	0.077	0.298	B
29	268.68	134.34	7	0.021	0.295	0.237	0.008	0.050	0.100	0.177	0.098	0.357	B
30	224	56	1	0.017	0.123	0.034	0.007	0.021	0.014	0.204	0.020	0.089	C
31	216	72	5	0.017	0.158	0.170	0.007	0.027	0.071	0.186	0.062	0.251	B
32	212.08	53.02	2	0.016	0.116	0.068	0.007	0.020	0.028	0.199	0.024	0.106	C
33	197.92	49.48	5	0.015	0.109	0.170	0.006	0.018	0.071	0.189	0.060	0.239	B
34	190.89	7.07	7	0.015	0.016	0.237	0.006	0.003	0.100	0.193	0.086	0.307	B
35	181.8	60.6	3	0.014	0.133	0.102	0.006	0.023	0.043	0.195	0.036	0.154	C
36	163.28	40.82	3	0.013	0.090	0.102	0.005	0.015	0.043	0.198	0.032	0.138	C
37	150	30	5	0.011	0.066	0.170	0.005	0.011	0.071	0.193	0.058	0.230	C
38	134.8	67.4	3	0.010	0.148	0.102	0.004	0.025	0.043	0.195	0.037	0.159	C
39	119.2	59.6	5	0.009	0.131	0.170	0.004	0.022	0.071	0.190	0.061	0.241	B
40	103.36	51.68	6	0.008	0.113	0.203	0.003	0.019	0.085	0.190	0.073	0.278	B
41	79.2	19.8	2	0.006	0.043	0.068	0.002	0.007	0.028	0.207	0.015	0.069	C
42	75.4	37.7	2	0.006	0.083	0.068	0.002	0.014	0.028	0.205	0.019	0.084	C
43	59.78	29.89	5	0.005	0.066	0.170	0.002	0.011	0.071	0.196	0.058	0.228	C
44	48.3	48.3	3	0.004	0.106	0.102	0.002	0.018	0.043	0.200	0.033	0.141	C
45	34.4	34.4	7	0.003	0.075	0.237	0.001	0.013	0.100	0.194	0.086	0.308	B
46	28.8	28.8	3	0.002	0.063	0.102	0.001	0.011	0.043	0.203	0.030	0.128	C
47	25.38	8.46	5	0.002	0.019	0.170	0.001	0.003	0.071	0.200	0.057	0.222	C
			${V_j}^+$				0.184	0.078	0.100				
			${V_j}^-$				0.001	0.002	0.014				