An optimization model for a manufacturing-inventory system with rework process based on failure severity under multiple constraints

Ata Allah Taleizadeh
Department of Industrial Engineering, College of Engineering University of Tehran, Tehran, Iran.
E-mail address: taleizadeh@ut.ac.ir

Reza Askari
Department of Industrial Engineering, Faculty of Engineering, Kharazmi University, Tehran, Iran
E-mail address: reza.askari.g@gmail.com

Ioannis Konstantaras*
Department of Business Administration, University of Macedonia
156 Egnatia Street, GR- 54636 Thessaloniki, Greece, Email address: ikonst@uom.gr

Abstract

The present work investigates a manufacturing-inventory system with a single machine and multiple products, featuring returns on sales and backorders. In the proposed model, some imperfect items, including scrapped and defective items, are produced by the manufacturer. Such items can be classified, based on the severity of the failure, into several categories; as a result, the rework process is carried out at different rates. Moreover, the implementation of the quality control policy requires monitoring and checking the items during the production and reworking processes via an inspection process. The present study is aimed to calculate and obtain the optimal values of the cycle length and backorders quantity for every product in order to achieve the minimum total cost of system considering machine capacity, service level, warehouse space, and budget constraints. To solve the presented model, given as a Nonlinear Programming (NLP) problem, the GAMS software as well as four commonly used algorithms, which are categorized among the meta-heuristic algorithms, are used. These algorithms include the GA (Genetic Algorithm), IWO (Invasive Weed Optimization), GWO (Grey Wolf Optimizer) and HHO (Harris Hawks Optimization) algorithms. Along with these algorithms, the Response Surface Methodology (RSM) is applied to calibrate the parameters of the proposed algorithms. Finally, several numeric problems are solved, the results of which are then compared with each other. Moreover, an analytical hierarchy process (AHP) technique for order performance by similarity to ideal solution (TOPSIS), which is a hybrid method of decision making with multiple attributes, is used for ranking the algorithms.

Keywords: Imperfect production process; scrapped items; rework process; quality control

*Corresponding author

1. Introduction

One of the most common models of lot-sizing widely used in the industries is a mathematical architecture concerned with the manufacturing-inventory system, which is known as the *EPQ* (*Economic Production Quantity*) model. The EPQ model is yielded by extending the economic architecture known as *EOQ* (*Economic Order Quantity*) model so that the EOQ model, in the case of being assumed with a constant production rate, would engender the EPQ model; therefore, the EPQ model can be regarded as the extension of the EOQ model (Barzoki et al. [3]). In other words, if instead of purchasing needed products from other suppliers, they are produced inside the company, usually the EPQ model is used to for optimal lot-sizing. This model has so far been generalized for adaptation to the operational sections of companies by considering various development states. Among the most significant developments accomplished in relation to the EPQ patterns, the addition of quality control topics can be mentioned since it prompted the researchers to conduct further studies and investigations on the practical applications of this model. Porteus [40] is the pioneer in the field of integrating issues of quality control and inventory control. Considering product quality in inventory control problems became of interest to researchers by the presentation of this paper. Then, researchers conducted more studies in this field. For instance, an EPQ model was developed by Hayek and Salameh [14] for a case with uniformly distributed percentage of defective products. In this model, the main assumptions included the permissibility of backorders and access of all defective products to optimal quality after rework operations. Later, an extension of the model presented by Hayek and Salameh [14] was proposed by Chiu [8] who incorporated the assumption that only a portion of the defective products, instead of all of them, be subjected to the rework operations, aiming to achieve the intended quality, and the remaining be sold at a predetermined price to sell. A minimized system cost in this model can be achieved by assuming the defective items to have a stochastic rate, scrapping a portion of those items that rate considered defective, as well as the option of reworking the defective parts. In order to calculate the optimum cycle length in an EPQ pattern, Chiu et al. [7] considered scrap, rework, and stochastic machine failure. In their study, defective items were considered either as defective or as repairable. An EPQ pattern was developed by Ouyang and Chang [36] aimed at calculating the optimum size of lot under the permissible delay in payment policy. In their model, shortage was allowed and considered as complete backlogging. Recently, Taleizadeh et al. [49] modeled a multiproduct manufacturing system with a single machine that had defective items and the delayed payment policy. Also, in order for obtaining more realistic results, they assumed a limited capacity of production as well as a partial backlogging for their model. In the same year, by applying the imperfect production process and also executing the screening process both during and after the manufacturing process, Taleizadeh et al. [51] formulated two architectures for the EPQ model with multiple products and a single machine aiming for the identification of defective products. They also considered two different policies for defective products: Selling with discount and implementing the rework process.

One of the most important factors in product quality is inspection. Products inspection in manufacturing systems leads to separation of scrap, defective and non-defective items. Therefore, adopting an appropriate inspection policy can decrease the delivery of products with imperfect quality to customers, and increase customer satisfaction. In this direction, in a study conducted by Sarkar and Saren [41], a product inspection strategy was utilized to model an imperfect manufacturing system with inspection failures and warranty costs. Their objective of adopting a production inspection policy was to reduce inspection costs. In Cheikhrouhou's et al. [5] study, an inventory model was developed applying the inspection policies to the lot-sizing process. In their research, in addition to introducing inspection error, the option of sending back products as defective items was feasible in two different ways: (i) defective items were sent by the retailer, with supplier's investment, while the next lot was received from the supplier and (ii) the retailer immediately removed, with his own payment, the defective items from the system and sent them back to the supplier. Kang et al. [20] conducted a study focusing on the relationship between the efficiency of inspection and the factors concerned with the human labor and the time spent on inspecting different products. In order to obtain the process target values, they investigated the inspection in off-line mode, for the purpose of which they assumed three levels of skill for the inspectors. These levels included the errors of inspection, quantities of inspection, and cost of inspection. In another work, an imperfect manufacturing system was developed by Sarkar et al. [44] aiming to optimize the run-time of production and the policy of inspection. In their model, the product inspection is performed at an arbitrarily chosen time during the cycle of production. Besides, in this work, the inspector committed two types of inspection errors (namely, *type I* error and error *type II*) in order to yield a more realistic model.

Since manufacturing products with defective quality is inevitable in manufacturing systems, reworking defective products can be beneficial to the production system. Therefore, if the reworking process is carried out using appropriate policies, it is possible to reduce production costs. For instance, Jamal et al. [19] studied an EPQ problem with regard to the process of reworking at a single-stage manufacturing system. They could obtain the optimal production quantity. In Yoo's et al. [61] work, an EPQ model was developed with sale return as well as rework possibility. Also, in this model, both the inspection and production processes are imperfect. Yoo's et al. [62] model was later expanded by Taheri-Tolgari et al. [45] who added another stage of inspection that followed the rework process. They also considered a discounted cash-flow approach for the imperfect items under the inflation circumstances. Afterward, Sarkar et al. [42] studied a model of EPQ considering the process of reworking in a one-stage manufacturing system wherein the backorders were planned. They assumed three probability distribution functions (namely uniform, triangular, beta) for the defective production rate. Moreover, an EPQ model was developed by Fadlil et al. [10] for defective items regarding carbon emissions, remanufacturing and rework processes, and inspection error. Several other studies on EPQ inventory model that consider rework process are: Hou [16], Liao et al. [28], Chiu et al. [9], Chiu et al. [6].

The production models are also developed with regard to various constraints in order to accommodate the operational section of companies. In this direction, Nobil et al. [33] introduced a model with multiple products and multiple machines in an imperfect manufacturing system by which two kinds of defective items were produced. The process of reworking, shortages, as well as defective item scrapping are also considered in their model. The constraints of their problem include item allocation, Machine utilization, budget, production floor space, and capacity of the single-machine. In the EPQ model developed by Pasandideh et al. [38], which was a model with multiple products and a single machine, the production process was imperfect and the shortage was considered as backorder. In this model, the authors also imposed some factors, including the capacity of machine, service level, and budget, as the constraint of the problem. Nobil et al. [34] investigated an imperfect production system considering an EPQ model with multiple machines and multiple products. Also, the factors assumed by the authors as the constraints of the problem included: product assignment, machine utilization, budget, production floor space, and capacity of the single machine. In another study conducted by Pasandideh et al. [37], it was attempted to optimize a multi-product EPQ problem assuming some stochastic constraints for it, which included the cost of backorder, cost of space, cost of ordering, cost of procurement, and total accessible budget. Khalilpourazari et al. [23] introduced an EPQ model with multiple products while assuming partial backordering and physical constraints for it. Other relevant works that address the constraints of manufacturing systems are: Nobil et al. [35], Gharaei et al. [12], Pirayesh and Poormoaied [39], Beheshti Fakher et al. [4] and Sarkar et al. [43].Table 1 presents a list of authors who have made major contributions in this regard.

Based on the literature review in the field of production systems and inventory, research gaps in this area have been identified and efforts have been made to address these gaps and provide valuable contributions. The research gaps and important contributions of this research are as follows: First, to date, a limited number of studies have been performed on single-machine multi-product manufacturing-inventory systems that simultaneously consider the imperfect production process, rework process, 100% inspection of items, inspection error, return policy and backorders. We fill this gap by providing a comprehensive model. Second, in the manufacturing-inventory systems literature, no model considers the rework process based on failure severity with inspection process. The proposed model takes these conditions into account. Third, based on our findings in the manufacturing-inventory systems literature, inspection errors in production systems were limited to type I and II type errors. In the present study, by designing different scenarios for inspection error, it is tried to bring the production conditions and inspection error closer to the real world conditions. Fourth, there are limited models that, in addition to considering the impact of returned items with various types of defects in the production cycle, also accurately calculate the holding cost of these items. In the proposed model, this important issue is considered. Fifth, a limited number of optimization models in manufacturing-inventory systems consider the constraints that exist for managers in companies. The proposed model considers the constraints of machine capacity, service level, warehouse space and budget that most production and inventory managers struggle with.

This study is aimed at designing an applicable model with regard to the conditions and constraints of the production systems. In the designed model, manufacturer produces imperfect items including scrapped items and defective items. Since the classification of the defective items into multiple groups is done on the basis of the severity of the failure, the rework process is performed at different rates. In order to implement quality control policy, the items undergo the inspection processes during the production and rework processes. The present study is aimed to calculate and obtain the optimal values for the cycle length and backorder quantity of each of the products in order for achieving the minimum total cost of the manufacturing-inventory system with respect to the different constraints.

Table 1. Review table

Author(s)	Inventory Control System	Defective	Scrap	Rework	Inspection		Inspection Error	Shortage		Products		Constraint			
					100% Inspection	Sampling		Lost Sale	Backorder	Single	Multi	Service level	Capacity	Space	Budget
Haji et al. [13]	EPQ	✓	✓	✓	✓		✓			✓					
Taleizadeh et al. [47]	EPQ	✓	✓					✓	✓		✓	✓			
Taleizadeh et al. [48]	EPQ	✓		✓					✓		✓		✓		
Barzoki et al. [3]	EPQ	✓	✓	✓	✓					✓					
Wee and Widyadana[57]	EPQ	✓		✓						✓					
Yoo et al. [62]	EPQ	✓	✓	✓	✓	✓				✓					
Hsu and Hsu [17]	EPQ	✓			✓		✓		✓	✓					
Taleizadeh et al. [50]	EPQ	✓	✓	✓					✓		✓		✓		
Wee et al. [54]	EPQ	✓		✓					✓	✓					
Sarkar et al. [42]	EPQ	✓		✓					✓	✓					
Taleizadeh et al. [46]	EPQ	✓	✓	✓						✓					
Tayyab and Sarkar [56]	EPQ	✓		✓	✓					✓					
Kang et al. [22]	EPQ	✓	✓	✓	✓					✓					
Kang et al. [21]	EPQ	✓		✓					✓	✓					
Al-Salamah [2]	EPQ	✓		✓	✓				✓	✓					
Taleizadeh et al. [52]	EPQ	✓		✓							✓	✓			
Taleizadeh et al. [53]	EPQ	✓		✓							✓		✓		✓
Taleizadeh et al. [54]	EPQ	✓		✓							✓		✓		
Taleizadeh et al. [55]	EPQ	✓		✓							✓		✓		
This paper	EPQ	✓	✓	✓	✓		✓		✓		✓	✓	✓	✓	✓

The remaining of the paper is structured as follows. Section 2 introduces the single-machine multi-product problem. The mathematical formulation of the model is developed in Section 3. Section 4 presents four meta-heuristic algorithms for solving the model. In Section 5, the four meta-heuristic algorithms are ranked in terms of different criteria. And finally, Section 6 includes the conclusion as well as some suggestions for further research in future.

2. Definition of problems and assumptions

The single-machine multi-product manufacturing-inventory system proposed in the present study produces imperfect items including scrapped items and defective items. Since the basis for classifying the defective items into several groups is the severity of the failure, the rework process is performed at different rates. The behavior and the forward and reverse material flow of the manufacturing-inventory system are depicted in Fig. 1 and 2, respectively.

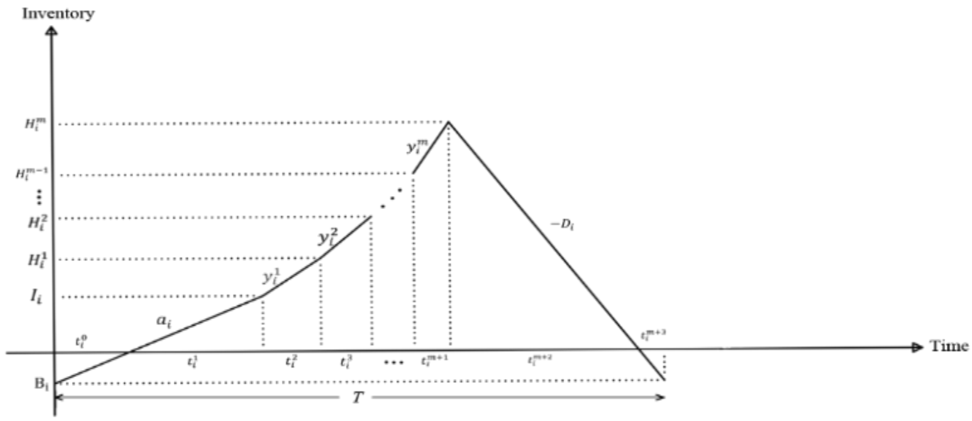


Fig. 1. The inventory system's behavior

Based onFigs.1 and 2, the manufacturer produces the lot size of product i (Q_i) with a rate of P_i in regular production time $(t_i^0 + t_i^1)$ and inspects all of them (100% inspection) simultaneously. Due to the imperfection of the manufacturing process, the lot-sizingof the product i (Q_i) includes three types of items:

(i) Scrapped items $(\theta_i Q_i)$, where θ_i is the proportion of produced scrapped products.
(ii) Defective items $(\alpha_i Q_i)$, where $\alpha_i = \sum_{j=1}^{m} \alpha_i^j$and α_i^j is the ratio of the produced $i - th$ product with the $j - th$defective product.
(iii) Non-defective items $((1 - \sigma_i)Q_i)$, where $\sigma_i = \theta_i + \alpha_i$ and σ_i is the proportion of produced imperfect-quality products.

In addition, since the first stage of the inspection process of the entire lot-size of the product i (Q_i), which is performed simultaneous with the production process over a regular production time, is not perfect, both types of the inspection errors, namely type-I & II, are created by the inspector of the first stage, where the respective ratios are as follows:
$e1_i^j = Pr$ (items screened as the jth defective type | Non − defects),
$e2_i^j = Pr$(items classify as Non − defects | defects with j − th defective type),$(0 < e1_i^j < 1)$,$(0 < e2_i^j < 1)$.
Where $e1_i^j$ and $e2_i^j$ are independent of the proportion of produced imperfect-quality products (σ_i) andfollow these equalities: $e1_i^1 = e1_i^2 = \cdots = e1_i^m$,$e1_i = \sum_{j=1}^{m} e1_i^j = m\, e1_i^j$,$e2_i^1 = e2_i^2 = \cdots = e2_i^m$,$e2_i = \sum_{j=1}^{m} e2_i^j = m\, e2_i^j$.

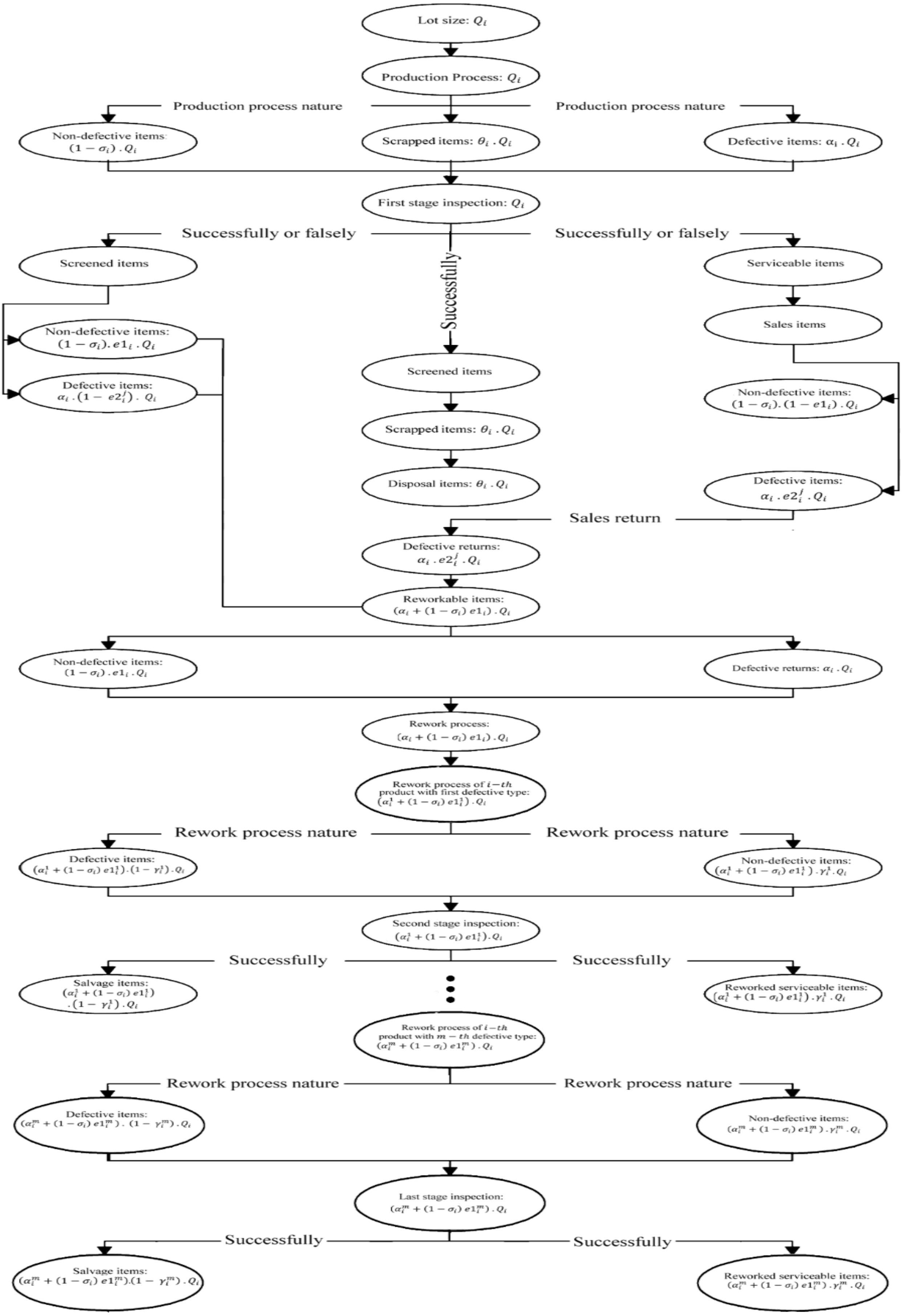


Fig. 2. Inventory flow of the proposed design

Different scenarios of the first stage of the inspection process are shown in Fig. 3.

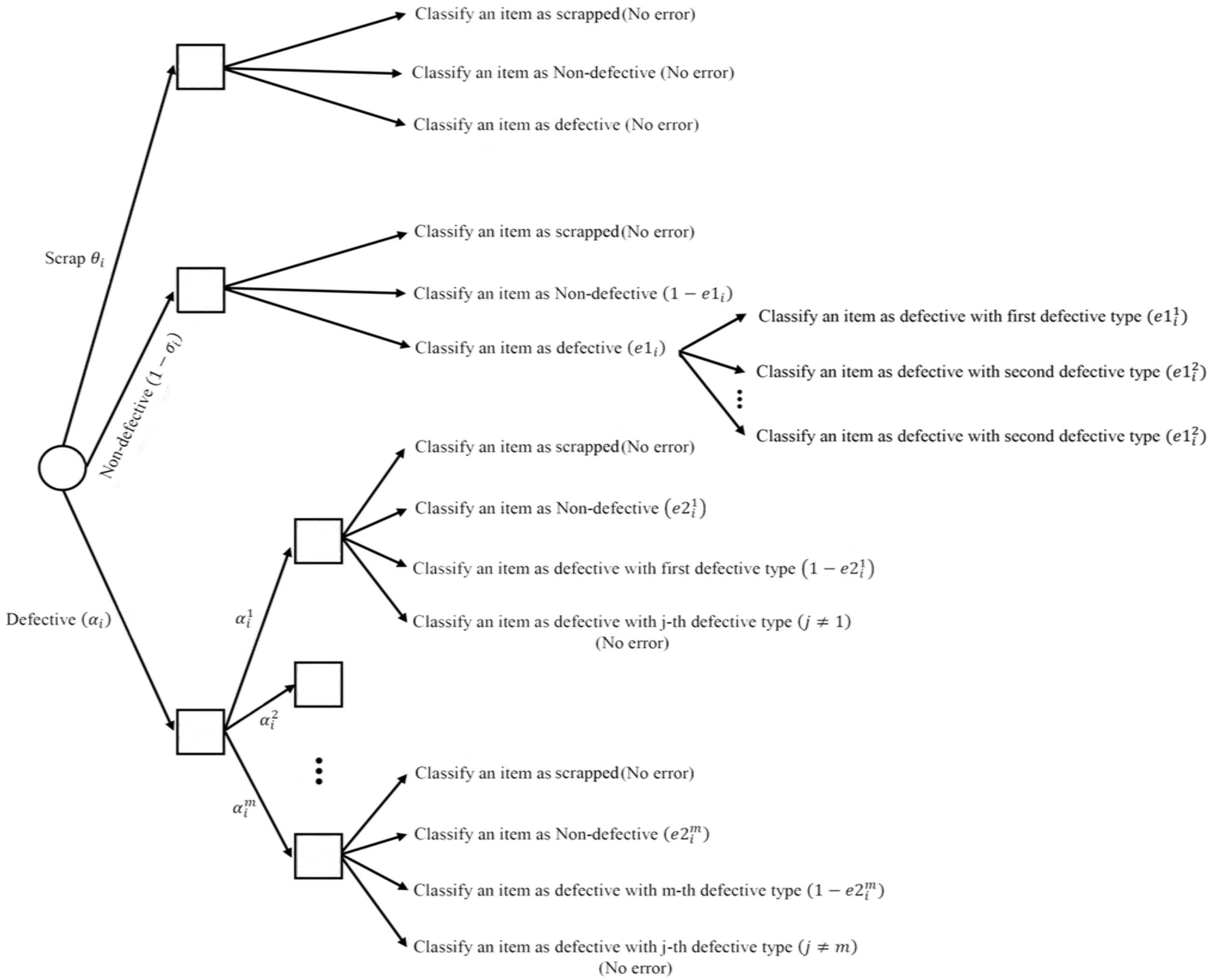


Fig. 3. Different scenarios of the first stage of the inspection process

In the course of regular production time $(t_i^0 + t_i^1)$ when the first stage of the inspection process is performed, since there are no inspection errors for the scrapped items, $\theta_i Q_i$ unit(s) are screened out and successfully recognized as scrapped items and disposed after the termination of regular production time. By contrast, on the one hand, as a result of the inspection error type-I, the $\left((1-\sigma_i)e1_i Q_i = (1-\sigma_i)e1_i^1 Q_i + (1-\sigma_i)e1_i^2 Q_i + \ldots + (1-\sigma_i)e1_i^m Q_i\right)$ unit(s) among the non-defective products of $\left((1-\sigma_i)Q_i\right)$are mistakenly filtered out, recognized as defectives, and sent to the rework process. In addition, the $\left((1-\sigma_i)(1-e1_i)Q_i\right)$ unit(s) that are among the non-defective products of $\left((1-\sigma_i)Q_i\right)$ would be successfully recognized as serviceable items and then sold. On the other hand, because of the inspection error type-II, the $\left(\alpha_i e2_i^j Q_i = \alpha_i^1 e2_i^1 Q_i + \alpha_i^2 e2_i^2 Q_i + \ldots + \alpha_i^m e2_i^m Q_i\right)$ items that are among the defective items of $(\alpha_i Q_i)$ are incorrectly identified as serviceable items and sold while they should be recognized as defectives (those customers who purchase these defective products will recognize such imperfections in the product's quality and, thus, will send back them due to the quality dissatisfaction). Furthermore, the $\left(\alpha_i(1 - e2_i^j)Q_i\right)$ unit(s) among defective items of $(\alpha_i Q_i)$ are successfully screened out and considered as defectives and sent to the rework process.

Therefore, after termination of the regular production time, first stage inspector recognizes the $\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right].Q_i$ units among the lot Q_i as serviceable items and the $\left[(1-\sigma_i)e1_i + \alpha_i(1 - e2_i^j) + \theta_i\right].Q_i$ unit(s) among the lot Q_i as screened items. (See Fig. 4). It must be mentioned that the purpose of using the serviceable items is to meet the customers' continuous demand for the $i-th$ product with a rate of D_i, where $D_i \leq PS_i$ (the production rate and inspection rate of the serviceable products for the i^{th} product) $\leq P_i$ (the production rate and inspection rate of a lot for the $i-th$ product). ($a_i = PS_i - D_i > 0$).

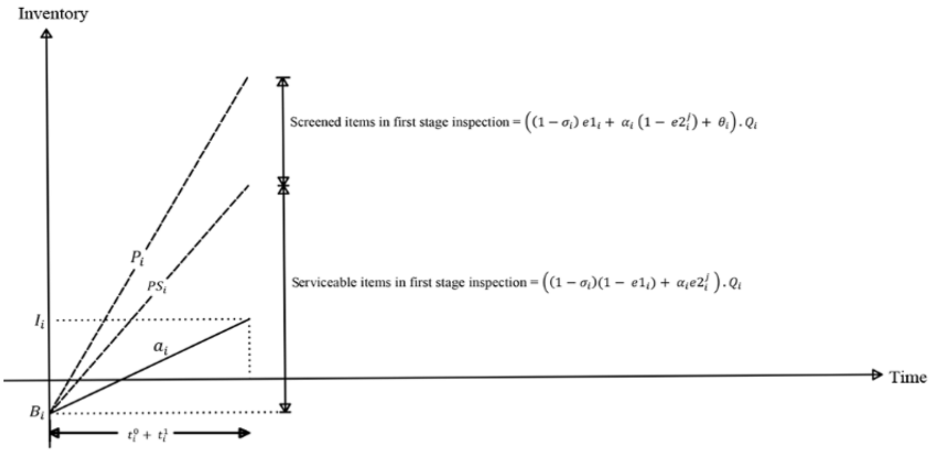


Fig. 4. Production process.

In our proposed model, the screened items in each cycle (excluding the scrapped items that are disposed, i.e. the $[(1-\sigma_i)e1_i + \alpha_i(1- e2_i^j)].Q_i$ units) and the returned items of $\alpha_i e2_i^j Q_i$ from the last cycle (overall the$[\alpha_i + (1-\sigma_i)e1_i]Q_i$ unit(s)) will enter the rework process. During the rework time (t_i^j, $j \geq 2$), the rework and inspection processes are performed simultaneously, and also the assumption is that any imperfect item will be produced during the reworking operation. However, the inspection operation is performed during the rework time for all the reworkable items due to the strict quality control policies. As it was mentioned before, the reworkable items were categorized into multiple groups based on the severity of the failure. Hence, the rework and inspection processes for the $i-th$ product with the $j-th$ defective type are performed at a rate of $PR_i^j = V_i^j P_i$. Since the process of reworking and inspecting a product commonly doesn't take more time in comparison with the process of producing and inspecting the product, the reworking and inspection rate is either equal to or bigger than the rate of production and inspection for all products ($V_i^j P_i > P_i$). As a result: $V_i^j > 1$ (Pasandideh et al. [31]).

In each stage of the reworking phase, there are two alternatives for the manufacturer to choose, including the reworked serviceable items and the salvage items, which are recognized by the corresponding inspectors. Assumingγ_i^j, the $[\alpha_i^j + (1-\sigma_i)e1_i^j]\gamma_i^j Q_i$ unit(s) among the reworkable items of $[\alpha_i + (1-\sigma_i)e1_i]Q_i$ are reworked serviceable items. It should be mentioned that reworked serviceable items don't have any defects and, thus, are capable to meet the customer's demand for the $i-th$ product with a rate of D_i, where, it is assumed that $D_i \leq PRS_i^j$ (the rework and inspection rate of the serviceable items for the $i-th$ product with the $j-th$defective type) $\leq PR_i^j$ (the rework and inspection rate of the reworkable items for the $i-th$ product with the $j-th$ defective type). ($y_i^j = PRS_i^j - D_i > 0$), ($PRS_i^j = \gamma_i^j PR_i^j$). Finally, it is assumed that the $[\alpha_i^j + (1-\sigma_i)e1_i^j](1-\gamma_i^j)Q_i$ unit(s) among the reworkable items of $[\alpha_i + (1-\sigma_i)e1_i]Q_i$ are salvaged as a single batch and then sold at a discounted price following the process of rework and inspection (Taheri-Tolgari et al. [38]) (See Fig.5).

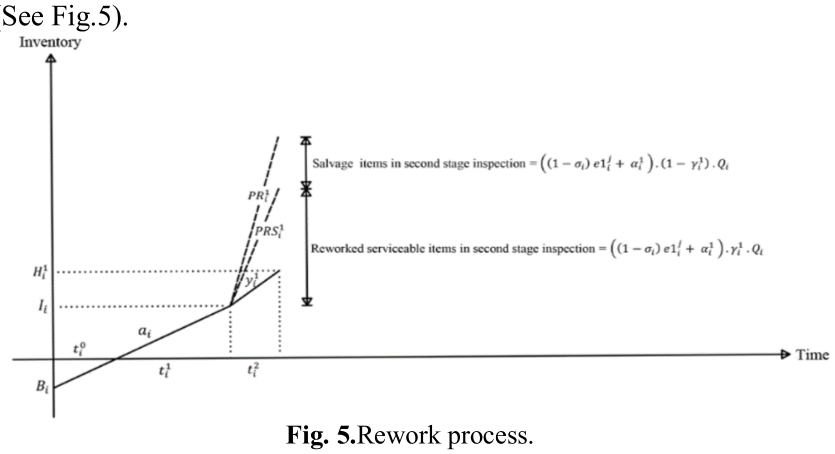


Fig. 5. Rework process.

Additionally, the assumptions taken into account in this study are as follows:

1. All products are produced by the same machine. Therefore, the length of the production cycle is the same for all the products. Mathematically speaking: $T = T_1 = T_2 = \cdots = T_n$.
2. For every item of the products, the shortage is assumed to be allowed and backordered.
3. The proportion of the manufactured $i-th$product with the $j-th$ defective type (α_i^j)obeys this inequality: $\alpha_i^1 \le \alpha_i^2 \le \cdots \le \alpha_i^m$(see Pasandideh et al. [38]).
4. The rework rates (V_i^j) are proportions of the normal rate of production that follows this inequality: $1 \le V_i^1 \le V_i^2 \le \cdots \le V_i^m$(See Pasandideh et al. [38]).
5. The rework proportions among screened and returned items for $i-th$ product with $j-th$ defective type is γ_i^j and follows this inequality: $\gamma_i^1 \le \gamma_i^2 \le \cdots \le \gamma_i^m$.
6. All the parameters are deterministic and known.
7. Production process and inspection process during regular production time are performed simultaneously.
8. Rework process and inspection processes in each stage during rework time are also performed simultaneously.
9. In each cycle, all of the defective items, which have escaped the screening phase, are sold. Then they are returned and finally entered the rework process in the next cycle.
10. The time horizon is infinite.

The presented model, with the aforementioned assumptions, leads to the optimization of production processes and improvement of product quality. Therefore, using this model and calculating the optimal cycle length and allowable shortage, production managers can reduce costs of the manufacturing-inventory system while increasing customer satisfaction.

3. Mathematical Modeling

Pasandideh et al. [38] studied a multiproduct single-machine model which did not adopt any quality policy. Since nowadays product quality is considered a critical competitive advantage in production, this study attempts to expand their model with regard to quality policies and developing expenses.

Table 2, as shown below, includes the notations used in the proposed model as well as their definitions.

Table 2. Notations.

Indices	
i	$The\ index\ of\ products\ (i = 1, 2, \ldots, n).$
j	$The\ index\ of\ defective\ types\ (j = 1, 2, \ldots, m)$
x	$Theindex\ of\ times\ (x = 0, 1, \ldots, m+3)$
Parameters	
N	$Number\ of\ cycles\ per\ year$
Q_i	$The\ production\ lot\ size\ of\ the\ i-th\ product\ in\ a\ cycle$
θ_i	$The\ proportion\ of\ produced\ scrapped\ products$
α_i^j	$The\ proportion\ of\ produced\ i-th\ product\ with\ j-th\ defective\ type$
σ_i	$The\ proportion\ of\ produced\ imperfect\ quality\ products\ (\sigma_i = \theta_i + \alpha_i^1 + \cdots + \alpha_i^m)$
V_i^j	$The\ ratio\ of\ the\ rework\ rate\ of\ i-th\ item\ with\ j-th\ defective\ type\ to\ the\ i-th\ item\ production\ rate\ (V_i^j \ge 1)$
γ_i^j	$The\ rework\ proportion\ among\ screened\ and\ returned\ items\ for\ i-th\ product\ with\ j-th\ defective\ type, independent\ of\ \sigma_i, e1_i\ and\ e2_i. (0 < \gamma_i^j < 1)$
D_i	$The\ demand\ rate\ of\ the\ i-th\ product$
P_i	$The\ production\ and\ inspection\ rate\ of\ a\ lot\ for\ i-th\ product$
PS_i	$The\ production\ and\ inspection\ rate\ of\ serviceable\ items\ for\ i-th\ product, \left(PS_i = \left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] P_i\right). (P_i \ge PS_i \ge D_i)$
PR_i^j	$The\ rework\ and\ inspection\ rate\ of\ reworkable\ items\ for\ i-th\ product\ with\ j-th\ defective\ type, \left(PR_i^j = V_i^j P_i\right)$

PRS_i^j	The rework and inspection rate of serviceable items for $i-th$ product with $j-th$ defective type, $(PRS_i^j = \gamma_i^j PR_i^j = \gamma_i^j V_i^j P_i)$, $(PR_i^j \geq PRS_i^j \geq D_i)$
H_i^m	The maximum on-hand inventory of $i-th$ item
I_i	The maximum on-hand inventory of $i-th$ item, based on the regular production process stops
H_i^j	The maximum on-hand inventory of the $i-th$ item, based on the rework process stops for $j-th$ defective type
ε_i	The safety factor of the total allowable shortage for $i-th$ item
W_i	The maximum available warehouse space for $i-th$ item
M	The total available budget
S_i	The machine setup time to produce $i-th$ product
μ_i	The space required per unit of $i-th$ item
δ_i	The ratio of the aisle space to the maximize level of on-hand inventory of $i-th$ item
F_i	The total required space of $i-th$ item
f_i	The unit warehouse construction cost of $i-th$ item per unit space
c_i	The unit production cost of $i-th$ product
r_i	The unit rework cost of $i-th$ product
d_i	The unit disposal cost of scrapped product i
A_i	The setup production cost of $i-th$ item
h_i	The unit holding cost of $i-th$ product per unit time
π_i	The unit backorder cost of $i-th$ product per time unit
g_i	The unit inspection cost of $i-th$ product
k_i	The unit return cost of $i-th$ product
l_i	The unit penalty cost from goodwill loss of $i-th$ product
$e1_i^j$	Proportion of a Type I inspection error for $i-th$ product with $j-th$ defective type, i.e., $e1_i^j$ = Pr (items screened as $j-th$ defective type $\mid$ Non-defects) independentof σ_i, so $(1-\sigma_i)e1_i^j$ = Pr (non-defective items $\cap$ items screened as $j-th$ defective type), $(0 < e1_i^j < 1)$
$e2_i^j$	Proportion of a Type II inspection error for $i-th$ product with $j-th$ defective type, i.e., $e2_i^j$ = Pr (items classify as Non-defects $\mid$ defects with $j-th$ defective type) independent of α_i, so $\alpha_i^j e2_i^j$ = Pr ($j-th$ defective type $\cap$ items classsify as Non-defective), $(0 < e2_i^j < 1)$
Decision variables	
T	The cycle length (in year), $(T = 1/N)$
B_i	The total shortage quantity of $i-th$ item in a cycle
Other notation	
CA	The annual setup cost
CP	The annual production cost
CR	The annual rework cost
CH	The annual holding cost
CB	The annual backorder cost
CD	The annual disposal cost
CC	The total warehouse construction cost
CI	The annual Inspection cost
CE	The annual Return and penalty cost
TC	The total inventory system cost

3.1. The manufacturing-inventory system

According to the Fig. 1, on the time axis, t_i^0 and $t_i^1$are the production uptimes, t_i^2 , t_i^3, …, $t_i^{m+1}$are reworktimes and $t_i^{m+2}$and t_i^{m+3} are the production downtimes for each product which are calculated as follows.

$$t_i^0 = \frac{B_i}{a_i} = \frac{B_i}{\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] P_i - D_i} \tag{1}$$

$$t_i^1 = \frac{Q_i}{P_i} - \frac{B_i}{a_i} = \frac{Q_i}{P_i} - \frac{B_i}{\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] P_i - D_i} \tag{2}$$

$$t_i^2 = \frac{H_i^1 - I_i}{y_i^1} = \frac{H_i^1 - I_i}{\gamma_i^1 V_i^1 P_i - D_i} = [\alpha_i^1 + (1-\sigma_i)e1_i^1]\frac{Q_i}{V_i^1 P_i} \tag{3}$$

$$t_i^3 = \frac{H_i^2 - H_i^1}{y_i^2} = \frac{H_i^2 - H_i^1}{\gamma_i^2 V_i^2 P_i - D_i} = [\alpha_i^2 + (1-\sigma_i)e1_i^2]\frac{Q_i}{V_i^2 P_i} \tag{4}$$

$$t_i^{m+1} = \frac{H_i^m - H_i^{m-1}}{y_i^m} = \frac{H_i^m - H_i^{m-1}}{\gamma_i^m V_i^m P_i - D_i} = [\alpha_i^m + (1-\sigma_i)e1_i^m]\frac{Q_i}{V_i^m P_i} \tag{5}$$

$$t_i^{m+2} = \frac{H_i^m}{D_I} \tag{6}$$

$$t_i^{m+3} = \frac{B_i}{D_i} \tag{7}$$

Based on Fig. 1, we can claim that the cycle length equals:

$$T = \sum_{x=0}^{m+3} t_i^x \tag{8}$$

Moreover, in Fig. 1, on the inventory axis, I_i is the highest level of the inventory that is available following the normal process of production, and H_i^1, H_i^2, … , H_i^m represent the highest levels of the available inventory after the first, second to$m-th$ rework processes, respectively, which are calculated as follows.

$$I_i = a_i\left(\frac{Q_i}{P_i}\right) - B_i \tag{9}$$

$$H_i^1 = I_i + [\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1 \frac{Q_i}{V_i^1 P_i} \tag{10}$$

$$H_i^2 = H_i^1 + [\alpha_i^2 + (1-\sigma_i)e1_i^2]y_i^2 \frac{Q_i}{V_i^2 P_i} \tag{11}$$

Therefore, we have:

$$H_i^m = H_i^{m-1} + [\alpha_i^m + (1-\sigma_i)e1_i^m]y_i^m \frac{Q_i}{V_i^m P_i} = I_i + \frac{Q_i}{P_i}\left(\sum_{j=1}^{m} \frac{\left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right) \tag{12}$$

Moreover, as shown in Appendix A, the cycle length is proven to be:

$$T = \frac{\left(\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] + \sum_{j=1}^{m} \gamma_i^j \left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]\right)Q_i}{D_i} \tag{13}$$

$$Q_i = \frac{D_i\, T}{\left(\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] + \sum_{j=1}^{m} \gamma_i^j \left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]\right)} \tag{14}$$

3.2. The objective function

The summation of set up cost (CA), production (CP), rework (CR), holding (CH), backorder (CB), disposal (CD), warehouse construction (CC), inspection (CI), penalty and return (CE) are total cost of the manufacturing-inventory system (TC), which is shown in Eq. (15):

$$TC = CA + CP + CR + CH + CB + CD + CC + CI + CE \tag{15}$$

Below, you can see all the components derived from Eq. (15).

3.2.1. The setup cost

Since A_i is the production set up cost for the $i-th$ item and there are N cycles per year, the annual cost of set up for all items can be easily obtained using Eq. (16).

$$CA = \sum_{i=1}^{n} N\, A_i \tag{16}$$

According to the joint production policy$N = 1/T_i = 1/T$, Eq. (16) becomes:

$$CA = \sum_{i=1}^{n} \frac{A_i}{T} \tag{17}$$

3.2.2. The production cost

The total cost of production can be calculated and obtained using Eq.(18) assuming that $c_i$and Q_i, respectively, are the unit cost of production and the lot-size of the $i-th$ product.

$$CP = \sum_{i=1}^{n} N c_i Q_i \tag{18}$$

Again, according to the joint production policies, we have:

$$CP = \sum_{i=1}^{n} \frac{c_i Q_i}{T} \tag{19}$$

Therefore, using Eq. (14) and inserting it in Eq. (19) we have:

$$CP = \sum_{i=1}^{n} \left(\frac{c_i D_i}{\left(\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j \right] + \sum_{j=1}^{m} \gamma_i^j \left[\alpha_i^j + (1-\sigma_i) e1_i^j \right] \right)} \right) \tag{20}$$

3.2.3. The rework cost

In this manufacturing-inventory system, $[\alpha_i + (1-\sigma_i)e1_i]Q_i$ unit(s) enter the rework process and the unit rework cost of $i-th$ each product is r_i. As a result, the annual cost of rework is:

$$CR = \sum_{i=1}^{n} N\, r_i [\alpha_i + (1-\sigma_i)e1_i] Q_i = \frac{1}{T} \sum_{i=1}^{n} r_i [\alpha_i + (1-\sigma_i)e1_i] Q_i \tag{21}$$

Inserting Q_i from Eq. (14) results in:

$$CR = \sum_{i=1}^{n} \left(\frac{r_i [\alpha_i + (1-\sigma_i)e1_i] D_i}{\left(\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j \right] + \sum_{j=1}^{m} \gamma_i^j \left[\alpha_i^j + (1-\sigma_i) e1_i^j \right] \right)} \right) \tag{22}$$

3.2.4. The holding cost

As can be seen in Fig. 6, in each cycle, in the case of a positive inventory level, the holding costs will be imposed. For the under-investigation manufacturing-inventory system, the annual holding cost has been determined in Appendix B, as shown below:

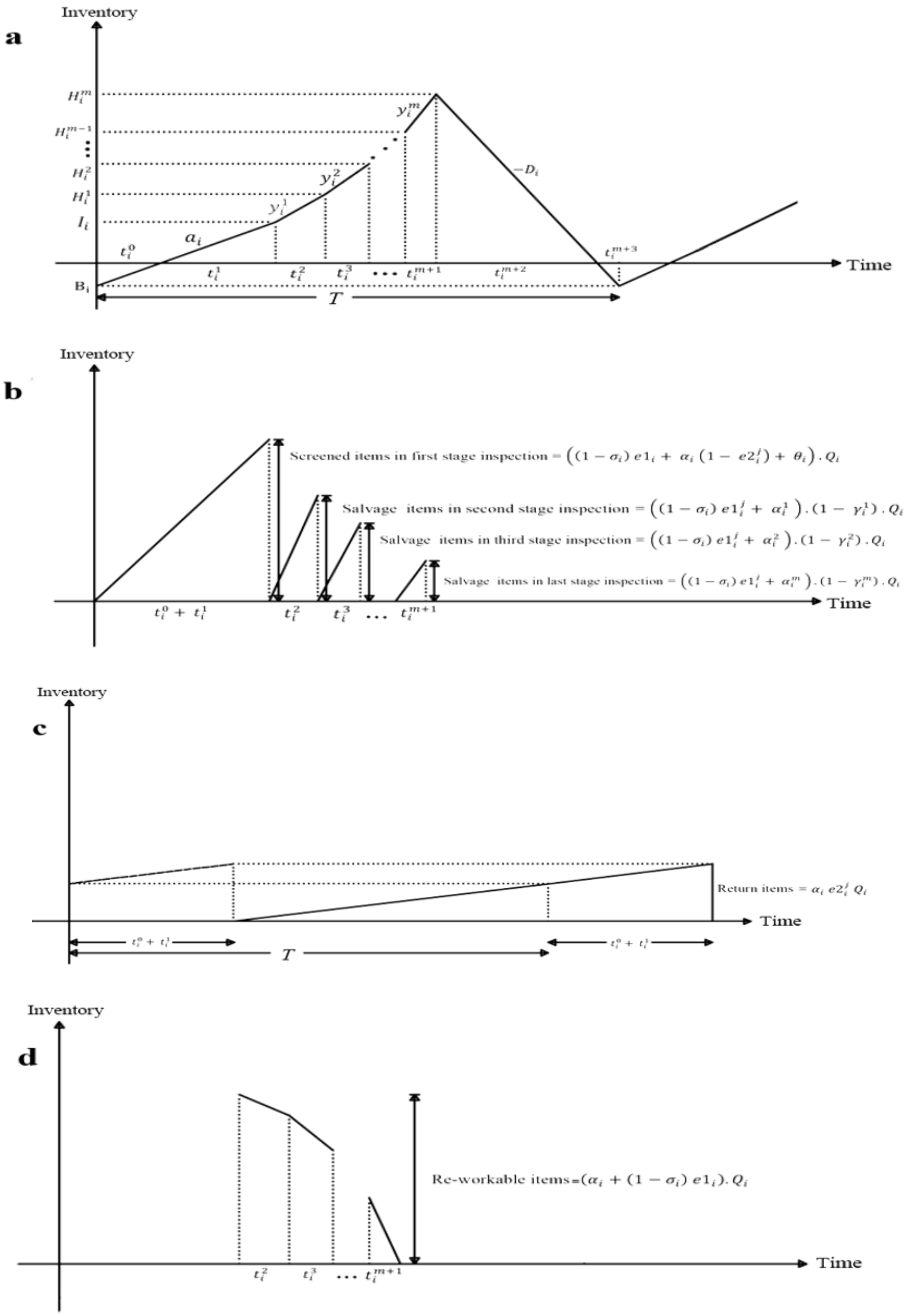


Fig. 6. The available inventory for the items with perfect quality and items with imperfect quality: (a) Serviceable items and reworked serviceable items; (b) screened items and salvage items; (c) sent back items; and (d) reworked items.

$$CH = CH_a + CA_b + CH_c + CH_d \tag{23}$$

$$\begin{aligned} CH_a = \frac{1}{2T}\sum_{i=1}^{n} h_i \Bigg[& I_i\left(\frac{Q_i}{P_i} - \frac{B_i}{a_i}\right) + \left(2I_i + [\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1 \frac{Q_i}{V_i^1\, P_i}\right)\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]Q_i}{V_i^1\, P_i}\right) \\ & + \left(2H_i^1 + [\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1 \frac{Q_i}{V_i^1\, P_i}\right)\left(\frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]Q_i}{V_i^2\, P_i}\right) + \ldots \\ & + \left(2H_i^{m-1} + [\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1 \frac{Q_i}{V_i^1\, P_i}\right)\left(\frac{[\alpha_i^m + (1-\sigma_i)e1_i^m]Q_i}{V_i^m\, P_i}\right) + \frac{(H_i^m)^2}{D_I}\Bigg] \end{aligned} \tag{24}$$

$$\begin{aligned} CH_b = \frac{1}{2T}\sum_{i=1}^{n} h_i \Big[& \big((1-\sigma_i)e1_i + \alpha_i(1 - e2_i^j) + \theta_i\big) Q_i\,(t_i^0 + t_i^1) \\ & + \big((1-\sigma_i)e1_i^j + \alpha_i^1\big)(1-\gamma_i^1)Q_i t_i^2 + \big((1-\sigma_i)e1_i^j + \alpha_i^2\big)(1-\gamma_i^2)Q_i t_i^3 + \ldots \\ & + \big((1-\sigma_i)e1_i^j + \alpha_i^m\big)(1-\gamma_i^m)Q_i t_i^{m+1}\Big] \end{aligned} \tag{25}$$

$$CH_c = \frac{1}{2}\sum_{i=1}^{n} h_i\left[\alpha_i e2_i^j Q_i\right] \tag{26}$$

$$\begin{aligned} CH_d = \frac{1}{2T}\sum_{i=1}^{n} h_i \Bigg[& \left[\big((1-\sigma_i)e1_i + \alpha_i\big) + \left(\sum_{j=2}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=2}^{m} e1_i^j\right)\right] Q_i t_i^2 \\ & + \left[\left(\sum_{j=2}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=2}^{m} e1_i^j\right) + \left(\sum_{j=3}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=3}^{m} e1_i^j\right)\right] Q_i t_i^3 + \cdots \\ & + \left[\sum_{j=m}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=m}^{m} e1_i^j\right] Q_i t_i^{m+1}\Bigg] \end{aligned} \tag{27}$$

3.2.5. The backorder cost

In each cycle, the backorder cost is imposed in the case of a negative inventory level. Hence, according to Fig. 6, the manufacturing-inventory system's annual backorder cost can be calculated and obtained as follows:

$$CB = \frac{1}{2T}\sum_{i=1}^{n} \pi_i\left[B_i t_i^0 + B_i t_i^{m+3}\right] \tag{28}$$

Inserting Eq. (1) and (7) into Eq. (28) results in:

$$CB = \frac{1}{2T}\sum_{i=1}^{n} \pi_i B_i\left[t_i^0 + t_i^{m+3}\right] = \frac{1}{2T}\sum_{i=1}^{n}\left(\frac{\pi_i P_i\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right]}{a_i D_i}\right)(B_i)^2 \tag{29}$$

3.2.6. The disposal cost

After the termination of the regular production time, $\theta_i Q_i$ unit(s) is considered as scrapped so that for each scrapped item, the disposal cost will be d_i. Consequently, the annual cost of disposal will be:

$$CD = \sum_{i=1}^{n} N\, d_i\theta_i Q_i = \frac{1}{T}\sum_{i=1}^{n} d_i\theta_i Q_i \tag{30}$$

Inserting Q_i from Eq. (14) results in:

$$CD = \sum_{i=1}^{n} d_i\theta_i\left[\frac{D_i}{\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]}\right] \tag{31}$$

3.2.7. The inspection cost

In the first stage of inspection process which is done in regular production period, $Q_i$unit(s) are inspected, and from the second stage to the $m+1-th$ stage of inspection process that are done in the rework period, $(\alpha_i^1 + (1-\sigma_i)e1_i^1)Q_i$, $(\alpha_i^2 + (1-\sigma_i)e1_i^2)Q_i$, ... , $(\alpha_i^m + (1-\sigma_i)e1_i^m)Q_i$ unit(s) are inspected. Therefore, annual cost of inspection of all of the items will be:

$$CI = \sum_{i=1}^{n} N\, g_i Q_i + \sum_{i=1}^{n} N\, g_i(\alpha_i^1 + (1-\sigma_i)e1_i^1)Q_i + \sum_{i=1}^{n} N\, g_i(\alpha_i^2 + (1-\sigma_i)e1_i^2)Q_i + \cdots + \sum_{i=1}^{n} N\, g_i(\alpha_i^m + (1-\sigma_i)e1_i^m)Q_i \tag{32}$$

$$CI = \frac{1}{T}\sum_{i=1}^{n} g_i Q_i + \frac{1}{T}\sum_{i=1}^{n} g_i(\alpha_i + (1-\sigma_i)e1_i)Q_i = \frac{1}{T}\sum_{i=1}^{n} g_i Q_i(1+\alpha_i + (1-\sigma_i)e1_i) \tag{33}$$

InsertingQ_i from Eq. (14) results in:

$$CI = \sum_{i=1}^{n} g_i D_i(1+\alpha_i + (1-\sigma_i)e1_i)\frac{1}{\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]} \tag{34}$$

3.2.8. The return and penalty cost

The unit return cost of the $i-th$product, including the communications and reverse logistics costs,is k_i, and the unit penalty cost of the $i-th$ product isl_i because of the loss of creditability due to the customer's quality dissatisfaction. Therefore, the annual costs of penalty and return are obtained by:

$$CE = \sum_{i=1}^{n} N\,(l_i + k_i)\alpha_i e2_i^j\, Q_i = \frac{1}{T}\sum_{i=1}^{n}(l_i + k_i)\alpha_i e2_i^j Q_i \tag{35}$$

InsertingQ_i from Eq. (14) results in:

$$CE = \sum_{i=1}^{n} \frac{(l_i + k_i)\alpha_i e2_i^j D_i}{\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]} \tag{36}$$

3.2.9. Warehouse construction cost

In the presented manufacturing-inventory system, the total space of warehouse for the $i-th$ itemincludes the storage space and the aisles. The space required for every unit and the maximum available inventory of the $i-th$ item are indicated by$\mu_i$and H_i^m, respectively. Thus, the space required for the storage will be $\mu_i H_i^m$. Furthermore, the aisle space for the $i-th$ product item is assumed to be a percentage (δ_i) of the space required for its storage. Therefore, the total warehouse space can be calculated and obtained as shown below:

$$F_i = \mu_i H_i^m + \delta_i \mu_i H_i^m = \mu_i H_i^m(1+\delta_i) \tag{37}$$

Since, the unit cost of warehouse construction for the $i-th$ item isf_i, we can claim that the construction cost of warehouse for all items is

$$CC = \sum_{i=1}^{n} f_i F_i = \sum_{i=1}^{n} f_i\, \mu_i H_i^m(1+\delta_i) \tag{38}$$

By using Eqs. (9), (12) and (14), the warehouse construction cost is:

$$CC = \sum_{i=1}^{n} f_i\, \mu_i(1+\delta_i)\left(\frac{D_i\, T}{\left([(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j] + \sum_{j=1}^{m}\gamma_i^j[\alpha_i^j+(1-\sigma_i)e1_i^j]\right)P_i}\right)\left[a_i + \left(\sum_{j=1}^{m}\frac{[\alpha_i^j+(1-\sigma_i)e1_i^j]y_i^j}{V_i^j}\right) - \frac{\left([(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j] + \sum_{j=1}^{m}\gamma_i^j[\alpha_i^j+(1-\sigma_i)e1_i^j]\right)P_iB_i}{D_i\, T}\right] \quad (39)$$

3.3. The constraints

The following sub-sections provide a description of the constraints considered for the given problem.

3.3.1. The constraint of machine capacity

It should be borne in mind that when only a single machine is present in the manufacture system, it limits the production potential. In other words, the maximal potential of only one machine is the single restriction of the model as explained below. As $t_i^0+t_i^1$, $t_i^2+t_i^3+\ldots+t_i^{m+1}$ and S_i are the regular production time, rework time, and setup time of the $i-th$ product,in respective order,then the total regular production time, rework time and setup time (for all products) is summed as $\sum_{i=1}^{n}(t_i^0+t_i^1+t_i^2+\cdots+t_i^{m+1})+\sum_{i=1}^{n}S_i$. Clearly, this needs to be lower or equivalent to the period length T. Hence,

$$\sum_{i=1}^{n}(t_i^0+t_i^1+t_i^2+\cdots+t_i^{m+1})+\sum_{i=1}^{n}S_i \le T \quad (40)$$

It is proven in Appendix C that the machine capacity constraint is:

$$\sum_{i=1}^{n}\frac{D_i\, T}{\left([(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j] + \sum_{j=1}^{m}\gamma_i^j[\alpha_i^j+(1-\sigma_i)e1_i^j]\right)P_i}\left(1+\frac{[\alpha_i^1+(1-\sigma_i)e1_i^1]}{V_i^1} + \frac{[\alpha_i^2+(1-\sigma_i)e1_i^2]}{V_i^2}+\cdots+\frac{[\alpha_i^m+(1-\sigma_i)e1_i^m]}{V_i^m}\right)+\sum_{i=1}^{n}S_i \le T \quad (41)$$

3.3.2. The constraint of service level

The service level is a quantity-centered performance scale that describes the percentage of total demand in a reference duration, the delivery of which is not delayed from the available stock. Hence, as ε_i is the factor of safeness of the total allowable shortage for the $i-th$ item, the annual service level constraint for the $i-th$ item can be determined as below:

$$\frac{N\, B_i}{D_i} \le \varepsilon_i \quad ; \quad \forall i \quad (42)$$

Based on the joint production policies, we have:

$$\frac{B_i}{\varepsilon_i D_i} \le T \quad ; \quad \forall i \quad (43)$$

3.3.3. The constraint of warehouse space

Warehouse space constraint is one of the common constraints in production and inventory management that production and inventory managers often face. This constraint states that the total space required to store items should not exceed the total available space. The space required to store items is often designed based on storage conditions and the type of items. However, in most inventory systems, warehouse space includes storage space and aisles. In the proposed manufacturing-inventory system, Eq. (37) shows the total space required for each item. Therefore, the constraint of warehouse space is as follows:

$$\mu_i H_i^m (1+\delta_i) \le W_i \quad ; \quad \forall\, i \tag{44}$$

WhereW_i is the maximum available warehouse for $i-th$ item. By using Eqs (9), (12), and (14), the warehouse space constraint becomes:

$$\begin{aligned}\mu_i(1+\delta_i)\left(\frac{D_i\,T}{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]+\sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)P_i}\right)\Bigg[a_i \\ +\left(\sum_{j=1}^{m}\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right) \\ -\frac{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]+\sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)P_iB_i}{D_i\,T}\Bigg] \\ \le W \quad ; \quad \forall\, i\end{aligned} \tag{45}$$

3.3.4. The constraint of budget

Since the budget available in most manufacturing-inventory systems is limited, it is necessary to impose this constraint on the model in order to consider the actual conditions of the proposed model. Therefore, the total cost of set up, production, rework, holding, backorder, disposal, warehouse construction, inspection, penalty and return should be less than the total budget available.

$$\begin{aligned}\sum_{i=1}^{n}\Big(c_iQ_i+r_i[\alpha_i+(1-\sigma_i)e1_i]Q_i+g_i(1+\alpha_i+(1-\sigma_i)e1_i)\,Q_i+d_i\theta_iQ_i+f_i\mu_iH_i^m(1+\delta_i) \\ +(l_i+k_i)\alpha_i e2_i^jQ_i\Big)\le M\end{aligned} \tag{46}$$

Where M is the total available budget. Based on, Eqs. (9), (12) and (14), the budget constraint is:

$$\begin{aligned}\sum_{i=1}^{n}\left(\frac{D_i\,T}{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]+\sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)\Bigg(c_i+r_i[\alpha_i+(1-\sigma_i)e1_i] \\ +g_i(1+\alpha_i+(1-\sigma_i)e1_i)+d_i\theta_i \\ +\left(\frac{f_i\,\mu_i(1+\delta_i)}{P_i}\right)\Bigg(a_i+\left(\sum_{j=1}^{m}\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right) \\ -\frac{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]+\sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)P_iB_i}{D_i\,T}\Bigg) \\ +(l_i+k_i)\alpha_i e2_i^j\Bigg)\le M\end{aligned} \tag{47}$$

3.4. The final model

Based on the objective function of Eq. (15) and also the constraints given in the Inequalities (41), (43), (45), and (47), the model that is yielded as the final model will be as follows:

$$TC = CA + CP + CR + CH + CB + CD + CC + CI + CE$$

s.t.:

$$\sum_{i=1}^{n} \frac{D_i\, T}{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right) P_i}\left(1+\frac{\left[\alpha_i^1+(1-\sigma_i)e1_i^1\right]}{V_i^1} + \frac{\left[\alpha_i^2+(1-\sigma_i)e1_i^2\right]}{V_i^2} + \cdots + \frac{\left[\alpha_i^m+(1-\sigma_i)e1_i^m\right]}{V_i^m}\right) + \sum_{i=1}^{n} S_i \le T$$

$$\frac{B_i}{\varepsilon_i D_i} \le T \quad ; \quad \forall\, i$$

$$\mu_i(1+\delta_i)\left(\frac{D_i\, T}{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)P_i}\right)\left[a_i + \left(\sum_{j=1}^{m}\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right) - \frac{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)P_i B_i}{D_i\, T}\right] \le W \quad ; \quad \forall\, i \tag{48}$$

$$\sum_{i=1}^{n}\left(\frac{D_i\, T}{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)\left(c_i + r_i\left[\alpha_i+(1-\sigma_i)e1_i\right] + g_i(1+\alpha_i+(1-\sigma_i)e1_i) + d_i\theta_i + \left(\frac{f_i\,\mu_i(1+\delta_i)}{P_i}\right)\left(a_i + \left(\sum_{j=1}^{m}\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right) - \frac{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)P_i B_i}{D_i\, T}\right) + (l_i+k_i)\alpha_i e2_i^j\right) \le M$$

$$T \ge 0$$
$$B_i \ge 0 \quad ; \quad i = 1, 2, \ldots, n$$

In the next section, we will present four algorithms to efficiently solve the problem given in Eq. (48).

4. Solving methods

In Eq. (48), the formulation is a problem of the nonlinear programming (NLP) type, for the solution of which the exact methods cannot be much effective in the case of large dimensions and reasonable time of computation. As a result, in order to obtain near-optimal solutions for it, four meta-heuristic algorithms were designed. These algorithms included the Genetic Algorithm (GA), Invasive Weed Optimization (IWO), Grey Wolf Optimizer (GWO), and Harris Hawks Optimization (HHO). A description of the framework of these algorithms is presented in the following sections.

4.1. Genetic algorithm

The GA is a method for finding a solution to optimize complex problems with the help of computer science. This algorithm is a special type of evolutionary algorithm based on Darwinian evolutionary theory and uses biology techniques. Generally, GA contains the following steps:

Step one - Creating a randomized initial population of the chromosomes.

Step two - Calculating the value of fitness for every chromosome.

Step three - Selecting the parent chromosomes with regard to the best fitness value to form the offspring's population.

Step Four - Creating a population of the offspring by applying a crossover operator and a mutation operator. The former operator should be applied to two parent chromosomes and the latter one should be applied to one parent chromosome.

Step Five - Creating the population of the new generation out of the chromosomes of the current population and the offspring population and then replacing it for the current population.

Step Six - If the stop condition is set, the algorithm is stopped and, in such a case, the best chromosome existing in the current population will be the solution to the given problem; otherwise, go to step two.

Below, the implementation of the genetic algorithm is described in detail.

4.1.1. The chromosomes

The genetic algorithm requires random populations of chromosomes to start the process. Therefore, at first step, a number of chromosomes are created randomly (N_{pop}). Each chromosome in a population is usually equivalent to a solution to the problem that is properly encoded and is a string of genes. In this study, chromosomes are a string of continuous variables they were made in two parts:

Section 1: A 1 × n matrix of the total quantity of shortage of the $i-th$ item in cycle (B_i),
Section 2: A 1 × 1 matrix of the cycle length (T).

For a problem in which there are$n$ products, the structure of the chromosome will be as shown in Fig. 7.

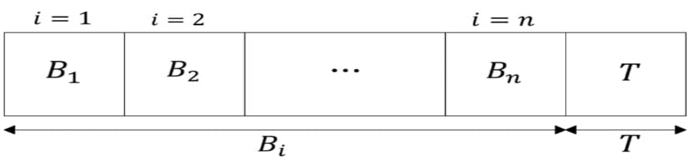


Fig. 7. Presentation of the chromosome.

4.1.2. Evaluation and initial population

In GA, the fitness of each chromosome is determined as soon as it is formed. Certainly the more optimal chromosomes will have a greater chance of combining with other chromosomes. In this stage, a function called the fitness function is used for evaluation of the chromosomes. Since there are four constraints in the presented model, some of the chromosomes generated in this problem may be infeasible. As a result, a control is exerted on the chromosomes in order to achieve the feasible chromosomes for satisfying all of the constraints at any time. In other words, once an infeasible chromosome is generated, it will be removed from the population.

4.1.3. Chromosomes selection

The selection mechanism for creating each new generation is repeated in the genetic algorithm. Among the chromosomes in a generation, only a few of the best chromosomes should be selected. In the present study, the roulette wheel method has been applied to select the best chromosome. So, the probability that is considered for each chromosome is inversely proportional to its cost and a random number is tasked with selecting a chromosome from a set of weighted chromosomes.

4.1.4. Crossover

Crossover operator combines the characteristics of the two chromosomes to generate new responses. This operator considers a solution and exchanges its places with other solutions and creates new solutions. The important thing about this operator is the crossover points where swaps occur. The less responses they receive in this operation, the closer the solutions will be to the previous population. In this study, the crossover is performed in a continuous space by random selection of a pair of chromosomes out of the generation with the probability of P_c. The proposed arithmetic crossover operator is implemented as described below:

(**i**) Choosing a random crossover point,
(**ii**) Creating offspring by exchanging tails using linear combination of two selective parents based on the following equation.

For example, if $x_1 = (x_{11}, x_{12}, ..., x_{1n})$ and $x_2 = (x_{21}, x_{22}, ..., x_{2n})$ are selected as two parents and $\alpha_i = (\alpha_1, \alpha_2, \dots, \alpha_n)$, where $-\tau \le \alpha_i \le \tau + 1$ and $\tau \ge 0$, the offspring are generated as follows:

$$\begin{cases} y_{1i} = \alpha_i x_{1i} + (1-\alpha_i) x_{2i} \quad ; \quad i = 1,2,3 \dots, n \\ y_{2i} = \alpha_i x_{2i} + (1-\alpha_i) x_{1i} \quad ; \quad i = 1,2,3 \dots, n \end{cases} \tag{49}$$

4.1.5. Mutation

The operator of mutation raises the population diversity and such variation will be the basis of development and progress towards the final solution. In principle, the mutation is a process in which a child's chromosome is produced by changing one or more genes associated with a parent chromosome. The mutation operator makes changes in the population, promotes the emergence of new genetic conditions and allows more areas to be evaluated from the search space which increases the chance of finding the optimal solution and prevents early algorithmic convergence to optimal topical points. The mutation operator with the probability of p_m and ϑ which is called mutation rate and shows the number of genes that are mutated, applies to the genes of a selected parent chromosome. The process of implementing the proposed operator of mutation is described below:

1. Selecting a random point (x_i) for mutation from the interval of x_{min} to x_{max},
2. Selecting a new point (x_i^{new}) that obeys a normal pattern of distribution with a mean of x_i and a standard deviation of σ. Mathematically speaking, $x_i^{new} \sim N(x_i, \sigma^2)$.

4.1.6. Stopping criterion

One of the criteria for controlling the genetic algorithm is the condition for the termination of generation. Choosing the right method for implementing the algorithm plays an effective role in its efficiency. Given that evolutionary algorithms are of a duplicate type, a condition must be considered for the completion of the algorithm. The type of the stop condition is dependent on the optimization problem's nature. A few methods are proposed to end the implementation of the algorithm:

(1) The algorithm is stopped as soon as reaching a certain number of generations,
(2) The algorithm is stopped as soon as reaching a certain number of function evaluations,
(3) Once the maximum number of evaluations is reached,
(4) Once the objective function shows no more improvements,
(5) As soon as reaching a certain value for the objective function.

In the present study, from among the above-mentioned criteria, the first one is applied for stopping the algorithm.

4.2. Invasive weed optimization algorithm

The algorithm known as IWO (invasive weed optimization) is an optimization algorithm with a smart and evolutionary nature, which has been inspired by the propagation, survival, and adaptability behaviors of the weeds. IWO was primarily presented by Mehrabian and Lucas [30]. The weeds seek for the most suitable environment for living and quickly get adapted to the conditions of their surrounding environment while showing resistance to the changes. Initially, the weed tends to be reproduced in large numbers leading to an increase both in its quantity and in the covered area of the surrounding environment (exploration behavior). Subsequently, due to the constraints of capacity, it continues its growth in a competitive mode with a growing quality (exploitation behavior). Generally, the target of the weeds is to find an environment that best suits for living. The general steps involved in an IWO algorithm are as follows:

Step one - Initialization.

1.1. Setting the parametersP_{max},S_{max},$\sigma_{initial}$,σ_{final}, Number of initial population, Minimum number of seeds, Maximum number of iterations).
1.2. Creating and dispersing an initial population of seeds randomly.

Step two–The second stage includes the seeds' dispersion. So, once they are grown, they are dispersed and transformed into a plant in terms of fitness and competence. Afterwards, they produce the seeds themselves.
Step three–The third stage involves the dispersion of the child seeds and their growth around their parent.
Step four - Finally, the stages 2 and 3 are repeated until the population of the seeds reaches a specific level (reachable range).
Step five - Otherwise, those plants that are more competent will survive and the rest will be eliminated.

The procedure of implementing the GA in the present work is described in detail in the following section.

4.2.1. Population initialization

The first stage involves the generation and then distribution of a certain number of seeds in a random – dimensional space. Then, the seed's fitness is calculated using the fitness function. The initially dispersed weeds are controlled in order that the feasible weeds can be obtained. In fact, in the case that a seed isn't feasible for the satisfaction of all constraints at all times, that seed will be eliminated and removed from the population in order that the generation of the infeasible solutions is avoided.

4.2.2. Reproduction

Seeds can be produced by each member of the population regarding the highest and lowest fitness of that member and also of the entire colony. For each member, the number of seeds begins with a value indicated by S_{min}, which represents the worst member. This value increases linearly up to S_{max}representing the best member. These values can be calculated using Eq. (50). This procedure is shown in Fig. 8.

$$Number\ of\ seeds\ around\ weed\ i = S_{min} + \frac{F_i - F_{worst}}{F_{best} - F_{worst}}(S_{max} - S_{min}) \quad (50)$$

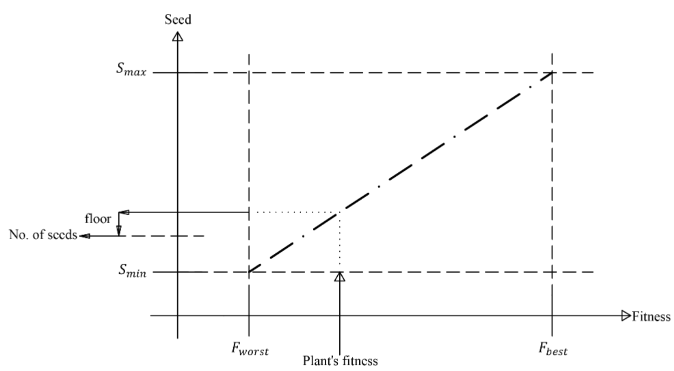


Fig. 8. Procedure of seed production in a weed colony

4.2.3. Spatial dispersal

Being produced in the search space, the seeds are dispersed based on a normal pattern of distribution with a zero mean and different variances. This is indicative of the fact that the seeds are distributed randomly and very close to their parents. The value of standard deviation, from the initial value set to the final value, will decrease in each step. Assuming σ_{init} and σ_{final} as the maximum and minimum values of the standard deviation, the value of standard deviation in a specific iteration can be calculated by Eq. (51).

$$\sigma_{iter} = \sigma_{final} + \left(1 - \frac{iter}{iter_{max}}\right)^n (\sigma_{init} - \sigma_{final}) \tag{51}$$

In Eq. (51), $iter_{max}$shows the maximum number of algorithm iterations and σ_{iter} is the standard deviation at the present iteration. Moreover, n indicates the nonlinear modulation index that has been set to 2 in this paper. The Fig. 9 shows the Eq. (51), when $n = 2$.

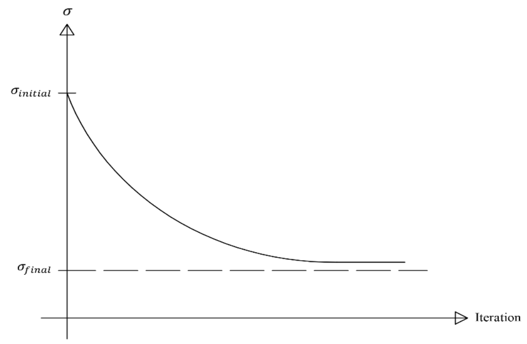


Fig. 9. The standard deviation of distance between parent weed and seed in each iteration (generation).

This step guarantees the nonlinear reduction of the probability of a seed's dropping in a distant area after each iteration. This will ultimately lead to the grouping of the plants with higher fitness so that the unsuitable plants are eliminated and removed from the population. On this basis, by applying such a mechanism of selection in the IWO algorithm, as the generations increase, the algorithm gradually moves from the exploration behavior toward the exploitation behavior.

4.2.4. Competitive exclusion

As a plant is gradually destroyed in the case of having no reproduction, it attempts to have reproduction and dispersion. Thus, limiting the maximum number of the plants in a colony necessitates the presence of competitiveness. After some repetitions are made, the population of the plants reaches its maximum number (P_{max}) by fast reproduction. As could be anticipated, plants with better fitness are more productive. As soon as the population of the plants in a colony reaches its maximum number, the elimination mechanism is activated to remove the plants that exhibit lower fitness compared to others.

Once the population of the plants reaches its maximum level (P_{max}), each seed can reproduce according to the mechanism mentioned in reproduction subsection. When all seeds find their place in the search space, they will adapt to their parents (seeds' colony). Subsequently, the plants with the lowest fitness for reaching the colony's highest acceptable population will be eliminated. The mechanism allows the lower-fit plants to have reproductions and, in the case of having children with appropriate fitness, survive.

4.3. Grey Wolf Optimizer algorithm

The ***gray wolf optimization*** (GWO) algorithm, which has been inspired by the hierarchal structure of the grey wolves during predation, was introduced for the first time by Mirjalili et al. [32].This method has a relatively simple and population-based procedure and can be easily generalized to the large-dimension problems. Generally, the GWO algorithm includes the following steps:

4.3.1. Social hierarchy

In GWO, the wolves in group $\boldsymbol{\alpha}$ are called the fittest solution while those in groups $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ are considered as the second and third solutions, respectively. Moreover, the remaining wolves are placed in group $\boldsymbol{\omega}$. It is worth noting that the group $\boldsymbol{\omega}$ wolves always follow the other three groups and the process of optimization in the GWO algorithm is invariably executed by groups $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\delta}$.The structure of the grey wolves' social hierarchy is shown in Fig. 10.

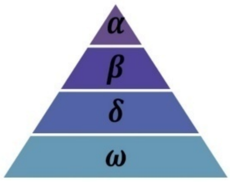


Fig. 10. Structure of the grey wolves' social hierarchy

4.3.2. Encircling prey

The process of prey encirclement by the grey wolves in the GWO algorithm is modeled as shown below:

$$\vec{X}(t+1) = \overrightarrow{X_p}(t) - \vec{A}.\left|\vec{C}.\overrightarrow{X_p(t)} - \overrightarrow{X(t)}\right| \tag{52}$$

Where $\overrightarrow{X_p}$ is the prey's position vector, $\vec{X}$ representsa grey wolf's position vector, t indicates the current iteration, and $\vec{C} = 2.\vec{r}_2$, and $\vec{A} = 2\vec{a}.\vec{r}_1 - \vec{a}$ are the coefficient vectors so that r_1 and r_2 randomly vary within the interval of $[0.1]$ and $\vec{a}$ declines steadily from 2 to 0 during the iterations of GWO algorithm as follows:

$$\vec{a}(t) = 2 - \frac{2t}{T} \tag{53}$$

Where T and t indicate the maximum number of iterations and the current iteration, respectively.

4.3.3. Hunting

Normally, the three groups α ، β, and δ altogether participate in the predation process. Therefore, the optimal solutions of these three groups are saved and the group $\boldsymbol{\omega}$ updates its location in accordance with the best location. This process can be modeled as the following:

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}\right| \quad , \quad \overrightarrow{D_\beta} = \left|\overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}\right| \quad , \quad \overrightarrow{D_\delta} = \left|\overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}\right| \tag{54}$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.(\overrightarrow{D_\alpha}) \quad , \quad \overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}.(\overrightarrow{D_\beta}) \quad , \quad \overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}.(\overrightarrow{D_\delta}) \tag{55}$$

$$\vec{X}(t+1) = w_1.\overrightarrow{X_1} + w_2.\overrightarrow{X_2} + w_3.\overrightarrow{X_3} \tag{56}$$

Where w_i is the weight of the grey wolf's position vector and $\sum_{j=1}^{n} w_j = 1$.

4.3.4. Search for prey (exploration) and attacking prey (exploitation)

Parameter C plays a major role in promoting the exploration process in the GWO algorithm. This parameter is independent of the number of the iterations of the algorithm and always generates a random number within the range of (0-2). This would alter the prey's contribution to the determination of the next location. If $C > 1$, the contribution will be stronger. Furthermore, parameter a, which is reduced steadily from 2 to 0 over the course of the iterations of the algorithm, causes the parameter A to vary within the interval of $[-2.2]$. If $|A| > 1$, the exploration process is improved and if $|A| < 1$, the exploitation process will be improved. The exploration and exploitation processes in different iterations are illustrated in Fig. 11 (Faris et al. [11]).

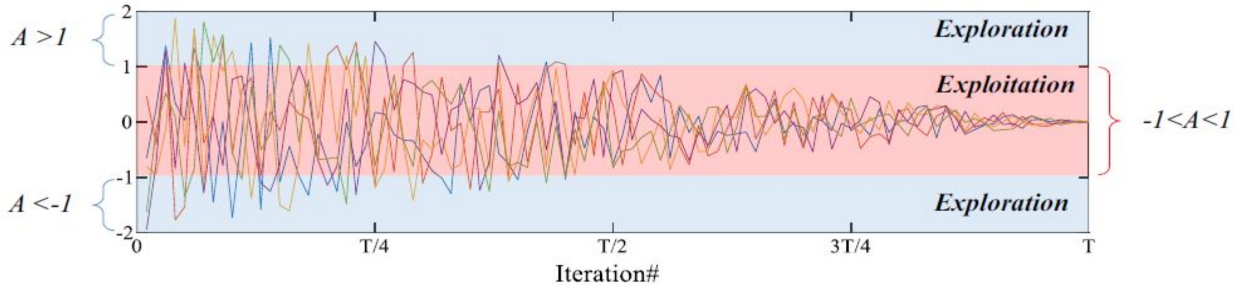


Fig. 11. The behavior parameter *A* during T iteration.

In addition to meta-heuristic algorithms, the exploration and exploitation processes are also employed in clustering of Bandits for use in recommendation systems. Such environments use a double challenge to recommendation approaches. **1)** Presentation of the novel items to the operators (or conversely the type of items to offer to novel operators) for optimal collection of preference information on the novel content (exploration). **2**) Using the whole accessible collected user-item preference information (exploitation) (Li and Kar [25]).

In this regard, Li et al. [26] announced a novel bandit algorithm that can add an additional exploration element over the cluster of operators. Besides the standard exploration-exploitation approach over items, the algorithm discovers various clustering tasks of novel operators and less-active operators. The four actual datasets evaluated experimentally against baselines and high-tech techniques affirm that the tendency of extra dynamic paradigm is the translation into solid performance advantages. Furthermore, an adaptable clustering procedure was explored for content recommendation according to exploration-exploitation approaches in contextual multi-armed bandit settings (Li et al. [27]). The collaborative impacts arising because of the interplay of the operators with the items are taken into consideration in the present algorithm, in which operators are grouped in a dynamic manner according to the items being considered and, simultaneously, items are grouped according to the resemblance of the clustering exerted over the operators. Recently, to overcome the limitations of the Distributed Clustering of Confidence Ball (DCCB) algorithm (Korda et al. [24]) leading to such problems as slowing down the cluster discovery, lowering accuracy communication and making a bottleneck, Mahadic et al. [29] have proposed a novel distributed bandit-based algorithm called DistCLUB. In the design of this algorithm, one of their strategies is that when clustering information are not used in recommendations and are dependent upon only the operator's previous interplays, this allows the explicit utilization of user-level interplay parallelism.

4.4. Harris Hawks optimization algorithm

The HHO algorithm, which is a meta-heuristic and population-based algorithm, was first introduced by Heidari et al. [15].As a nature-inspired algorithm, it has been inspired by the Harris hawks' chasing style – commonly known as the surprise pounce - and cooperative behavior. One of the major advantages of these hawks' cooperative tactic is the wearing down of the detected prey (usually a rabbit), which results in the vulnerability of the prey. The conceptual stages of the HHO algorithm are depicted in Fig. 12. As shown in Fig. 12, different stages of the HHO algorithm are as follows:

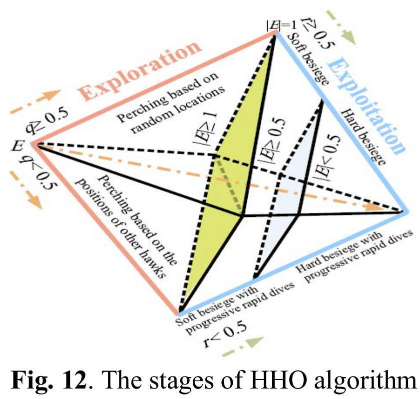


Fig. 12. The stages of HHO algorithm

4.4.1. Exploration stage

In this stage, the Harris hawks perch randomly in different locations and attempt the predation based on two strategies.

$$X(t+1) = \begin{cases} X_{rand}(t) - r_1|X_{rand}(t) - 2r_2X(t)| & ; \quad q \geq 0.5 \\ (X_{rabbit}(t) - X_m(t)) - r_3(LB + r_4(UB - LB)) & ; \quad q \geq 0.5 \end{cases} \tag{57}$$

Where q, r_1, $r_2 . r_3$, and r_4 are random numbers between 0 and 1 that are updated in each of the iterations of the HHO algorithm, LB and UB are, respectively, the lower and upper bounds of the variables, $X_{rabbit}(t)$indicates the rabbit's position, $X(t+1)$ indicates the hawks' position vector in the next iteration, $X_{rand}(t)$is a hawk that has been randomly chosen from among the existing population, and$X_m(t)$indicates the hawk's average position and can be obtained using Eq.(58).

$$X_m(t) = \frac{1}{N}\sum_{i=1}^{N} X_i(t) \tag{58}$$

Where N indicates the total number of the hawks. Moreover, $X_i(t)$ denotes the current position of each hawk in iteration t.

4.4.2. The exploration-to-exploitation transition

Since the prey's energy drops during the escaping process, the transition from the exploration stage to the exploitation stage can occur in accordance with the prey's escaping energy, which can be calculated and obtained by Eq. (59).

$$E = 2E_0\left(1 - \frac{t}{T}\right) \tag{59}$$

Where Tindicates the maximum number of the iterations of the HHO algorithm and E_0represents the prey's initial energy, which can be selected randomly between $(-1, 1)$ at each iteration. Since E_0varies randomly within the interval of $(-1, 1)$, the prey's energy will fall within the interval of $(-2, 2)$. It should be also mentioned that if $|E| < 1$, the HHO algorithm is in the exploitation process and if $|E| \geq 1$, it is in the exploration process. Fig. 13 illustrates the prey's energy over 500 iterations of the HHO algorithm.

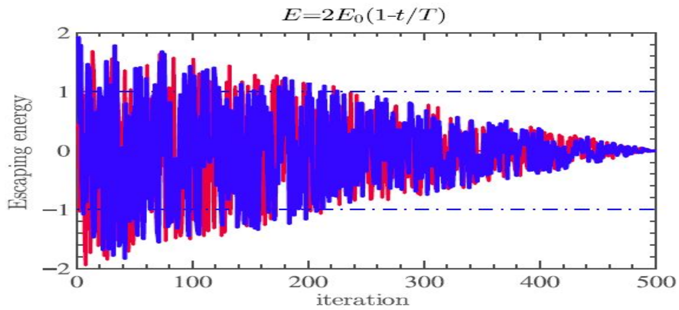


Fig. 13. The behavior of variation of the escaping energy during 500 iterations.

4.4.3. Exploitation stage

At this stage, the hawks' surprise pounces occur to confuse and attack the prey. By contrast, the prey tries to escape the dangerous circumstances. While the prey is escaping, the hawks perform the soft or hard besiegement of the prey in order to catch it (Heidari et al. [15]). Therefore, assuming r as the likelihood of the prey's escape, if$r < 0.5$, the prey's escaping process is successful and otherwise, it is unsuccessful. Furthermore, if$|E| < 0.5$, the hard besiegement is performed and otherwise, the soft besiegement occurs. Therefore, according to the values of $E$and r, four strategies are used to model the attack stage.

4.4.3.1. Strategy 1: soft besiege

If $r \geq 0.5$ and $|E| \geq 0.5$, then the prey's escape is unsuccessful and Harris hawks forms a soft besiegement on the prey and perform a sudden pounce. This fact is modeled as follows:

$$X(t+1) = \Delta X(t) - E\left|JX_{rabbit}(t) - X(t)\right| \quad (60)$$

$$\Delta X(t) = X_{rabbit(t)} - X(t) \quad (61)$$

Where $J = 2(1 - r_5)$shows the random physical power of the rabbit's jump while escaping the situation, r_5 is a random number within the interval of (-1.1), and $\Delta X(t)$represents the amount of difference in the present location at iteration *t* and the rabbit's position vector.

4.4.3.2. Strategy 2: hard besiege

If $r \geq 0.5$ and $|E| < 0.5$, the prey plans a successful escape and the Harris hawks form a hard besiege on the prey and perform a sudden pounce. This process can be modeled as the following:

$$X(t+1) = X_{rabbit}(t) - E|\Delta X(t)| \quad (62)$$

The hard besiegement is illustrated conceptually in Fig. 14.

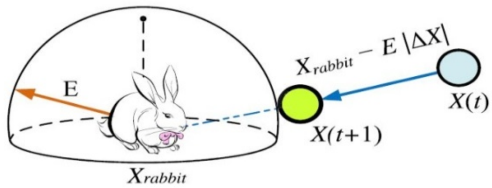


Fig. 14. The rabbit's position vector and current location in the hard besiege strategy.

4.4.3.3. Strategy 3: soft besiege with progressive rapid dives

If $r < 0.5$ and $|E| \geq 0.5$,the prey has a successful escape and the Harris hawks still carries out the soft besiege before doing the sudden pounce. Since this process exhibits more intelligence compared to the previous ones, the patterns of escaping and the leapfrog movements are modeled using a concept known as the ***levy flight*** (LF) concept. Thus, modeling of the hawk's next movement will be as shown below:

$$Y = X_{rabbit}(t) - E|JX_{rabbit}(t) - X(t)| \quad (63)$$

$$Z = Y + S \times LF(D) \quad (64)$$

Where D indicates the problem's dimension, S represents a random vector with a size of $(1 \times D)$.Also, the LF function can be obtained through the Eq. (65):

$$LF(X) = 0.01 \times \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}} \quad , \quad \sigma = \left(\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}}\right)^{\frac{1}{\beta}} \quad (65)$$

Where β represents a constant number that is set to 1.5 and u and v indicate random numbers within the interval of $(0, 1)$. Therefore, the hawks' new position is obtained using Eq. (66).

$$X(t+1) = \begin{cases} Y & if\ F(Y) < F(X(t)) \\ Z & if\ F(Z) < F(X(t)) \end{cases} \quad (66)$$

Fig. 15 shows a general illustration of the concept of this stage.

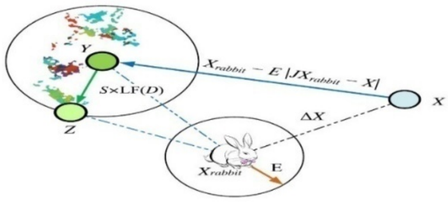


Fig. 15. The rabbit's position vector and current location in the soft besiege associated with progressive rapid dives.

4.4.3.4. Strategy 4: hard besiege with progressive rapid dives

If $r < 0.5$ and $|E| < 0.5$, there isn't enough energy for the prey to escape and the Harris hawks carries out the hard besiege before the sudden pounce. This behavior can be modeled by the Eq.(67)given below:

$$X(t+1) = \begin{cases} Y & if\ F(Y) < F(X(t)) \\ Z & if\ F(Z) < F(X(t)) \end{cases} \quad (67)$$

Where Z and Y can be calculated using new rule in Eq. (63) and Eq. (64), respectively. Fig.16 demonstrates a self-evident idea of this step.

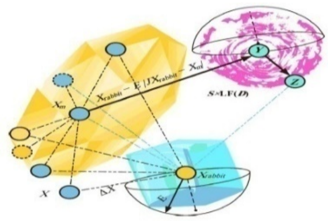


Fig. 16. The rabbit's position vector and current locations in hard besiege associated with progressive rapid dives.

4.5. Penalty functions to handle constraints

In evolution-based algorithms, three widespread methods exist for handling constraints, viz. direct, Lagrange multiplier, and penalty function (Mellouk et al. [31]). The direct method typically has inefficiency and slowness in convergence. The Lagrange multiplier procedure is based on meticulously mathematical method and has difficulty in implementation. Penalty function procedures have more simplicity in implementation practically and are highly rapid (Yang [59]).In this regard, the equality/inequality constraint functions are used for determining the Penalty function (Yeniay[60]). In this study, the penalty function method is employed to deal with the constraints and in order to minimize $F(x)$, the penalty function is presented in Eq. (68):

$$F(x) = f(x) \pm [p(x)],\ x = (x_r, x_2, ..., x_d) \in \mathbb{R}^d \quad (68)$$

In Eq. (68), $f(x)$ is the objective function, and $p(x)$ is the penalty function. The formulation of the exterior penalty function can be as below:

$$p(x) = \sum_{i=1}^{m} u.K.Eq_i^2(x) + \sum_{i=1}^{m} v.Z.Inq_i^2(x) \quad (69)$$

In which equality and inequality constraints are Eq_i and Inq_i, and follow these equalities: $\sum_{i=1}^{m} Eq_i(x) = 0$, and $\sum_{i=1}^{m} Inq_i(x) \le 0$. It is worth mentioning that u and v are penalty coefficients that are taken as constant values in most of cases. Yang [59] assumes that coefficients u and v equal 10^{10} and 10^{15} in respective order.

4.6. Parameter tuning

Since the meta-heuristic algorithms' efficiency is considerably reliant on their parameter values, and different combinations of parameters may end in different results, their parameters must be calibrated to ensure their maximum performance. In order to calibrate significant parameters and properly discover how the response and the significant parameters are related with each other, a regression analysis using *Response Surface Methodology* (RSM) is used. RSM is a method that estimates the relationships between one or more response variables with several independent variables, using a series of designed experiments and the regression analysis method. Generally, RSM includes the following steps:

1. Identifying independent variables that influence the response variable
2. Determining levels of the independent variables
3. Regression analysis for estimating the fitness equation of response variables in regard to the independent variables
4. Optimizing to specify the independent variables' optimum level

Our goal for using the RSM method is to find the optimal level of parameters of GA, IWO, GWO and HHO algorithms so that the performance of the mentioned algorithms to achieve the optimal solution is maximized.

According to the RSM method, it is first necessary to identify the parameters affecting the meta-heuristic algorithms (step 1) and then to determine the level of their parameters (step 2). For this purpose, the significant parameters of the algorithms have been identified and the level of their parameters has been determined based on extensive experiments and trial and error methods. It is important to note that after identifying the parameters affecting the algorithms, three levels of low, mean and high are considered for each parameter. Table 3 shows the significant parameters and the different levels of these parameters, which affect the response obtained in the GA, IWO, GWO and HHO algorithms.

Table 3. The level of parameters.

Algorithm	Parameter	description	Range	Low level	Mean level	High level
GA	N_{pop}	the population size	50-100	50	75	100
	P_c	the crossover probability	0.55-0.85	0.55	0.7	0.85
	P_m	the mutation probability	0.15-0.45	0.15	0.3	0.45
	ϑ	mutation rate	0.001-0.003	0.001	0.002	0.003
IWO	P_{max}	the maximum number of plant population	80-120	80	100	120
	S_{max}	the maximum number of seeds	4-10	4	7	10
	$\sigma_{initial}$	the initial value of standard deviation	0.2-0.3	0.2	0.25	0.3
	σ_{final}	the final value of standard deviation	0.001-0.003	0.001	0.002	0.003
GWO	It_{max}	the maximum iteration	1000-1500	1000	1250	1500
	N_{pop}	the population size of grey wolf	50-100	50	75	100
	α	the weight of group α in updating positions of grey wolf around prey	0.33-0.50	0.33	0.415	0.50
	β	the weight of group β in updating positions of grey wolf around prey	0.33-0.50	0.33	0.415	0.50
HHO	It_{max}	the maximum iteration	1000-1500	1000	1250	1500
	N_{pop}	the population size of hawks	50-100	50	75	100

Then, in order to implement regression analysis (step 3), Minitab 17.3.1 software is used. For this purpose, the number of effective parameters and different levels of each algorithm as RSM method inputs are given to Minitab 17.3.1 software. Then, the RSM method designs a number of experiments according to the number of significant parameters of each algorithm and their levels. In these experiments, there are different modes of the parameter levels. Next, to perform the designed experiments, a problem whose data is selected based on Table 4 is designed and coded in MATLAB (R2016a). Then, the designed problem is solved by considering different combinations of the parameter level of GA, IWO, GWO and HHO algorithms. (See Table 5 and 6 for IWO and GWO algorithms).

Table 4. Data generation.

Parameters	Value	Parameters	Value
P_i	$\sim U\ (5000\ ,6000)$	h_i	$\sim U\ (8\ ,16)$
θ_i	$\sim U\ (0.001\ ,0.005)$	π_i	$\sim U\ (16\ ,30)$
D_i	$\sim U\ (1000\ ,1400)$	g_i	$\sim U\ (0.5\ ,0.9)$
ε_i	$\sim U\ (0.07\ ,0.09)$	k_i	$\sim U\ (3\ ,5)$
S_i	$\sim U\ (0.00004\ ,0.0007)$	l_i	$\sim U\ (10\ ,15)$
μ_i	$\sim U\ (2\ ,5)$	W_i	$\sim U\ (250\ ,500)$
δ_i	$\sim U\ (2\ ,4)$	α_i^j	$\sim U\ (0\ ,0.08)$
f_i	$\sim U\ (50\ ,70)$	V_i^j	$\sim U\ (2\ ,5)$
c_i	$\sim U\ (35\ ,50)$	γ_i^j	$\sim U\ (0.7\ ,0.85)$
r_i	$\sim U\ (20\ ,25)$	$e1_i^j$	$\sim U\ (0.01\ ,0.03)$
d_i	$\sim U\ (12,20)$	$e2_i^j$	$\sim U\ (0.03\ ,0.07)$
A_i	$\sim U\ (400,800)$	M	$\sim U\ (25000\ ,85000)$

Note: U = uniform distribution.

Ultimately, at the confidence level 95%, a regression model is fitted for each algorithm based on obtained response values (as a response variable) and different level of parameters (as independent variables).It should be noted that the RSM typically uses a multiple regression model to fit a model. In this regard, if the response variable is well modeled by a linear function of independent variables, the multiple regression model will be of the first order. However, data may not be distributed linearly, in which case the multiple regression model should be fitted using higher-order polynomials, such as the second-order model. Generally, the regression equation in the RSM literature is as follows:

$$E(Y) = \beta_0 + \sum_{i=1}^{k} \beta_i X_i + \sum_{i=1}^{k} \beta_{ii} X_{ii}^2 + \sum_{i<j}^{k} \sum_{i<j}^{k} \beta_{ij} X_i X_j \tag{70}$$

In this Eq. (70), E(Y) is the expectable value of the response variable,$\beta_0, \beta_i, \beta_{ii}$ and $\beta_{ij}$are the model components, $X_i$and $X_j$are the input variables affecting the response Y, and k is the number of parameters. In this research, k factors affecting the response in the GA, IWO, GWO and HHO were presented in Table 3.

Based on the above- mentioned points, Tables 7-14 represent the estimated regression coefficients and the analysis of variance (ANOVA) for GA, IWO, GWO, and HHO algorithms after fitting the regression equation.

Table 5. The results of designed experiments based on RSM for IWO.

Number	P_{max}	P_c	$\sigma_{initial}$	σ_{final}	OBJ values
1	110	8.5	0.225	0.0025	290949.03
2	100	7	0.25	0.002	288079.90
3	90	8.5	0.225	0.0015	289138.07
4	90	8.5	0.225	0.0025	289322.29
5	90	5.5	0.225	0.0025	285911.13
6	90	5.5	0.275	0.0015	286452.37
7	100	7	0.25	0.002	287788.07
8	90	5.5	0.225	0.0015	285353.57
9	110	5.5	0.275	0.0015	288405.86
10	90	5.5	0.275	0.0025	286230.98
11	110	8.5	0.275	0.0015	291561.85
12	100	7	0.25	0.002	288080.78
13	110	8.5	0.225	0.0015	291230.07
14	90	8.5	0.275	0.0025	288647.63
15	100	7	0.25	0.002	288249.02
16	110	5.5	0.225	0.0025	288022.75
17	110	5.5	0.275	0.0025	289603.57
18	110	8.5	0.275	0.0025	290848.53
19	110	5.5	0.225	0.0015	288276.27
20	90	8.5	0.275	0.0015	288635.85
21	100	7	0.2	0.002	289924.63
22	100	7	0.25	0.002	287720.82
23	120	7	0.25	0.002	291169.32
24	100	7	0.3	0.002	290708.42
25	100	7	0.25	0.001	287837.96
26	100	10	0.25	0.002	290337.53
27	100	7	0.25	0.002	287188.98
28	80	7	0.25	0.002	285881.28
29	100	4	0.25	0.002	285602.77
30	100	7	0.25	0.003	288672.44

Table 6. The results of designed experiments based on RSM for GWO.

Number	It_{max}	N_{pop}	α	β	OBJ values
1	1250	75	0.35	0.35	285425.85
2	1250	75	0.5	0.35	286526.26
3	1000	75	0.35	0.35	286504.54
4	1250	100	0.35	0.35	287232.87
5	1500	75	0.35	0.35	285736.64
6	1250	75	0.35	0.2	288367.61
7	1250	50	0.35	0.35	286552.74
8	1250	75	0.35	0.35	285655.71
9	1250	75	0.2	0.35	288732.15
10	1250	75	0.35	0.5	287608.72
11	1375	62.5	0.275	0.425	287790.93
12	1125	62.5	0.425	0.275	287953.28
13	1125	87.5	0.275	0.275	289591.17
14	1250	75	0.35	0.35	285940.08
15	1375	62.5	0.275	0.275	288964.35
16	1125	87.5	0.425	0.425	287245.35
17	1375	87.5	0.275	0.275	288756.76
18	1250	75	0.35	0.35	286368.16
19	1250	75	0.35	0.35	286318.71
20	1125	62.5	0.425	0.425	287015.01
21	1375	62.5	0.425	0.275	287553.36
22	1375	87.5	0.425	0.425	287101.11
23	1375	87.5	0.275	0.425	288681.15

24	1375	62.5	0.425	0.425	287142.17
25	1250	75	0.35	0.35	286117.86
26	1125	87.5	0.425	0.275	287804.04
27	1125	62.5	0.275	0.275	288741.62
28	1375	87.5	0.425	0.275	288451.26
29	1125	87.5	0.275	0.425	287989.32
30	1125	62.5	0.275	0.425	287969.89

Table 7. Coefficients of GA fitness**.**

Term	Coef	SE – Coef	T – Value	P – Value
Constant	285293	102	2792.89	0
N_{pop}	913.7	72.2	12.65	0
P_c	851.8	72.2	11.79	0
P_m	196.5	72.2	2.72	0.012
ϑ	41.2	72.2	0.57	0.574
$N_{pop}*N_{pop}$	212.1	66.3	3.2	0.004
P_m*P_m	331.1	66.3	4.99	0
$N_{pop}*P_c$	492.7	88.5	5.57	0
$R-sq$ = 94.39% , $R-sq(adj)$ = 92.60%				

Table 8. Results of ANOVA for GA fitness**.**

Source	DF	Adj – SS	Adj – MS	F – Value	P – Value
Model	7	4.63E+07	6615183	52.83	0.000
Linear	4	38419717	9604929	76.71	0.000
Square	2	4003120	2001560	15.98	0.000
2-Way Interaction	1	3883444	3883444	31.01	0.000
Error	22	2754740	125215		
Lack-of-Fit	18	2440057	135559	1.72	0.319
Pure Error	4	314684	78671		
Total	29	4.91E+07			

Table 9. Coefficients of IWO fitness**.**

Term	Coef	SE – Coef	T – Value	P – Value
Constant	287947	108	2668.81	0
P_{max}	1240.9	76.3	16.27	0
S_{max}	1314.4	76.3	17.23	0
$\sigma_{initial}$	156.3	76.3	2.05	0.053
σ_{final}	89.6	76.3	1.17	0.253
$P_{max}*P_{max}$	139.1	70.1	1.98	0.06
$\sigma_{initial}*\sigma_{initial}$1	586.9	70.1	8.37	0
$S_{max}*\sigma_{initial}$	-254.7	93.4	-2.73	0.012
$R-sq$ = 96.70% , $R-sq(adj)$ = 95.66%				

Table 10. Results of ANOVA for IWO fitness.

Source	DF	Adj – SS	Adj – MS	F – Value	P – Value
Model	7	9.02E+07	12884702	92.24	0.000
Linear	4	79201728	19800432	141.74	0.000
Square	2	9953510	4976755	35.63	0.000
2-Way Interaction	1	1037673	1037673	7.43	0.012
Error	22	3073227	139692		
Lack-of-Fit	18	2821745	156764	2.49	0.195
Pure Error	4	251482	62871		
Total	29	9.33E+07			

Table 11. Coefficients of GWO fitness.

Term	Coef	SE – Coef	T – Value	P – Value
Constant	285831	122	2337.05	0
Blocks1	420.2	57.3	7.33	0
It_{max}	-58.5	60.4	-0.97	0.344
N_{pop}	160.4	60.4	2.66	0.015
α	-526.3	60.4	-8.71	0
β	-349.9	60.4	-5.79	0
$It_{max}*It_{max}$	185.6	56.5	3.28	0.004
$N_{pop}*N_{pop}$	378.6	56.5	6.7	0
$\alpha*\alpha$	562.7	56.5	9.96	0
$\beta*\beta$	652.5	56.5	11.55	0
$R-sq$ = 95.14% , $R-sq(adj)$ = 92.96%				

Table 12. Results of ANOVA for GWO fitness.

Source	DF	Adj – SS	Adj – MS	F – Value	P – Value
Model	9	34310568	3812285	43.54	0
Blocks	1	4709276	4709276	53.78	0
Linear	4	10286818	2571704	29.37	0
Square	4	19314474	4828619	55.15	0
Error	20	1751219	87561		
Lack-of-Fit	16	1608886	100555	2.83	0.162
Pure Error	4	142333	35583		
Total	29	36061787			

Table 13. Coefficients of HHO fitness.

Term	Coef	SE – Coef	T – Value	P – Value
Constant	286298	30	9417.99	0
It_{max}	83.3	24	3.46	0.002
N_{pop}	-33	24	-1.37	0.185
$N_{pop}*N_{pop}$	777.6	25.8	30.17	0
$It_{max}*It_{max}$	447.4	25.8	17.36	0
$R-sq$ = 98.14% , $R-sq(adj)$ = 97.79%				

Table 14. Results of ANOVA for HHO fitness.

Source	DF	Adj – SS	Adj – MS	F – Value	P – Value
Model	4	10236231	2559058	276.92	0
Linear	2	128314	64157	6.94	0.005
Square	2	10107917	5053959	546.9	0
Error	21	194061	9241		
Lack-of-Fit	4	144135	36034	12.27	0.11
Pure Error	17	49927	2937		
Total	25	10430293			

Based on the results of Tables 7-14 and with respect to the significance level of 95%, the following results are obtained.

1. Since the p-value of the model for the four algorithms GA, IWO, GWO and HHO is almost zero ($p < 0.05$), it can be concluded that at least one variable has a non-zero effect and regression at 95% confidence level is significant for the four algorithms.
2. The values of $R - sq$ and $R - sq\ (adj)$ for the four algorithms GA, IWO, GWO and HHO are close to 1. Therefore, it can be concluded that a significant part of the performance of the mentioned algorithms is influenced by the selected parameters.
3. The lack-of-fit values for the four algorithms GA, IWO, GWO and HHO are 0.195, 0.319, 0.162 and 0.11, respectively ($P > 0.05$). Thus, the lack-of-fit value is not significant, which confirms the appropriate prediction of the regression model.

Since the results confirm the proper performance of the regression analysis of the RSM method, the estimated regression functions for the GA, IWO, GWO and HHO algorithms according to Tables 7, 9, 11 and 13 are given in Eqs. (71)- (74), respectively.

$$Fitness_{GA} = 312243 - 524.7(N_{pop}) + 30059(P_c) - 28581(P_m) + 82447(\vartheta) + 1.358(N_{pop})^2 + 51730\,(P_m)^2 + 563\,(N_{pop})(P_c) \tag{71}$$

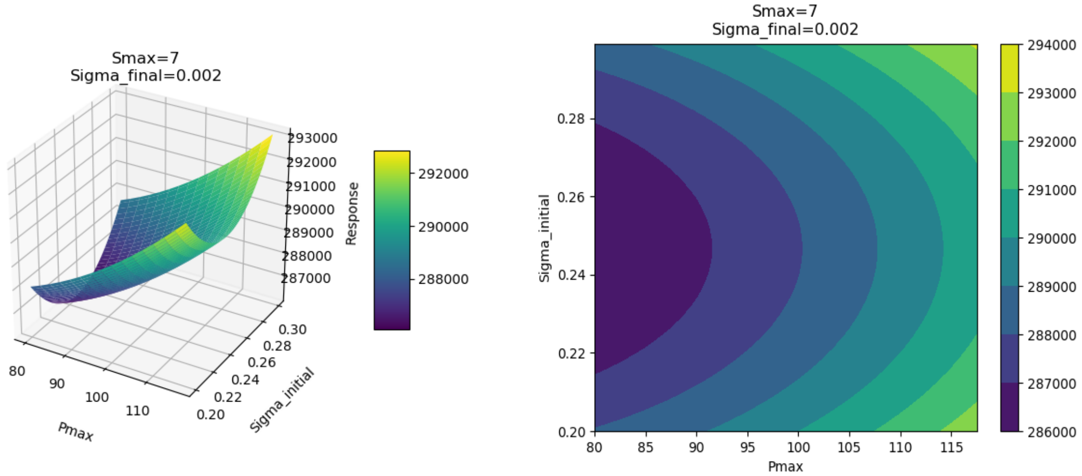
$$Fitness_{IWO} = 328190 - 154(P_{max}) + 2574\,(S_{max}) - 415700(\sigma_{initial}) + 179247\,(\sigma_{final}) + 1.391(P_{max})^2 + 938979(\sigma_{initial})^2 - 6791(S_{max})(\sigma_{initial}) \tag{72}$$

$$Fitness_{GWO} = 348198 - 30.16(It_{max}) - 350.7(N_{pop}) - 77047(\alpha) - 85864(\beta) + 0.01188(It_{max})^2 + 2.423\,(N_{pop})^2 + 100043(\alpha)^2 + 115997(\beta)^2 \tag{73}$$

$$Fitness_{GWO} = 322693 - 373.6(N_{pop}) - 36.08(It_{max}) + 2.5393(N_{pop})^2 + 0.014362(It_{max})^2 \tag{74}$$

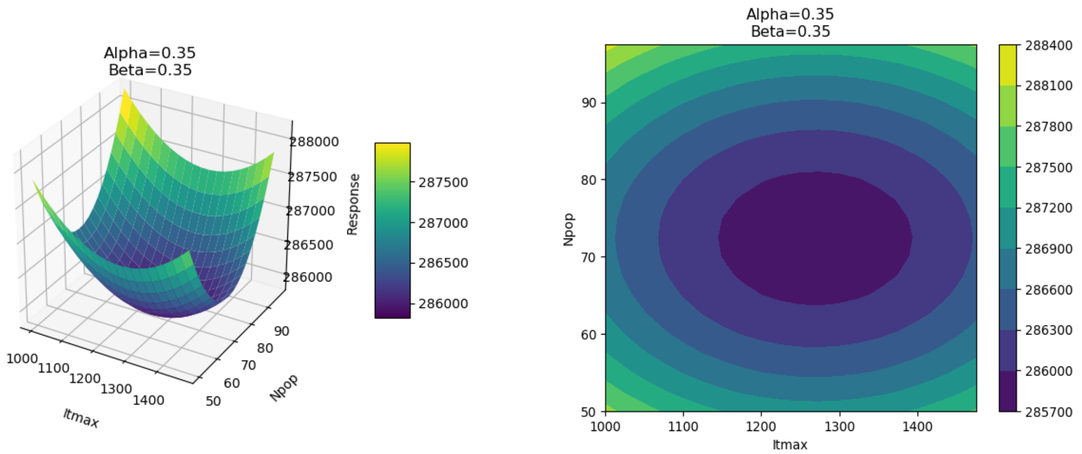
Fig. 17 shows the Surface Plot and Contour Plot of Eq. (72) based on the two variables P_{max} and $\sigma_{initial}$ when the two variables S_{max} and σ_{final} have values of 7 and 0.002, respectively. According to Fig. 17, where P_{max} is low and the $\sigma_{initial}$ values are in the middle of the corresponding range, the lowest response values are obtained for the IWO algorithm ($Fitness_{IWO}$). Fig. 18 also shows the Surface Plot and Contour Plot of Eq. (73) based on the two variables It_{max} and N_{pop} when the two variables α and β both have values of 0.35. According to Fig. 18, where the values It_{max} and N_{pop} are approximately in the middle range of their respective intervals, the lowest response values are obtained for the GWO algorithm ($Fitness_{GWO}$).

Eventually, in order to determine the optimal values of the parameters of the GA, IWO, GWO and HHO algorithms (Step 4), the objective functions obtained from the Eqs. (71)- (74), in the domains of their corresponding variables, are minimized using GAMS software. In this way, the optimal values of the four meta-heuristic algorithms will be obtained to have the best performance for solving the problem given in Eq. (48). Table 15 illustrates the optimal parameter values, which are actually the tuned values of the GA, IWO, GWO and HHO parameters.



a b

Fig. 17. Plot of response versus It_{max} and N_{pop} for IWO algorithm: (a) surface plot, (b) contour plot



a b

Fig. 18. Plot of response versusIt_{max} and N_{pop} for GWO algorithm: (a) surface plot, (b) contour plot

Table 15. Tuned values of GA and IWO parameters.

Algorithm	Parameter	Value tuned
GA	N_{pop}	79
	P_c	0.55
	P_m	0.276
	ϑ	0.001
IWO	P_{max}	80
	P_c	4
	$\sigma_{initial}$	0.236
	σ_{final}	0.001
GWO	It_{max}	1269
	N_{pop}	72
	α	0.385
	β	0.370
HHO	It_{max}	1256
	N_{pop}	74

5. *Ranking the algorithms*

Here, for ranking and showing the efficiency of the proposed GA, IWO, GWO, and HHO algorithms in solving single-machine multi-product model, 30 problems have been designed randomly with in different dimensions regarding the parameters provided in Table 4.At first, the GAMS/BARON 24.8 software is used to solve these problems. For this purpose, an Intel(R), core (TM) i7, 2.20 GHz laptop with 6.00 GB RAM was used. In addition to BARON, GA, IWO, GWO and HHO algorithms are implemented with their optimal significant parameters that presented in Table 15 in five separate executions for solving each 30 problems while the input parameters were the same. Afterwards, the best total costs of manufacturing-inventory system and its respective CPU time(s) were compared. Moreover, another quality measure that is called the objective function's percentage deviation is introduced to compare the results of BARON and four algorithms. This quality measure according to the Eq. (75) sets:

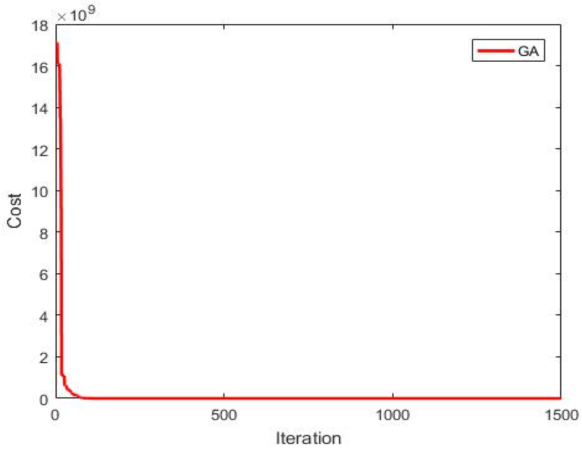
$$\%Deviation_{OBJ} = \left(\frac{z_{algorithm}}{z_{BARON}} - 1\right) \times 100\% \tag{75}$$

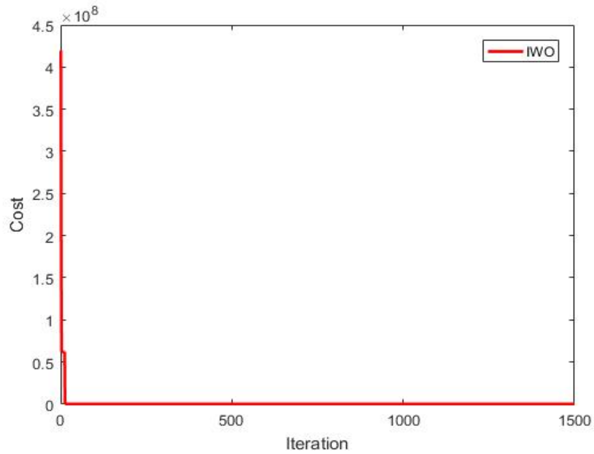
Tables16and 17 show the obtained results using the BARON, GA, IWO, GWO and HHO. It should be noted that the lower values of the percentage deviation and CPU time(s) measure indicate the higher performance of the presented algorithms.

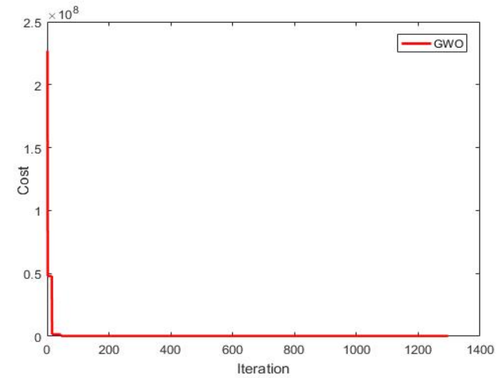
According to Tables 16 and 17, among the algorithms used to solve the model, the HHO algorithm has the best performance in terms of % Deviation with a value of 0.02 in problem 18. However, the best performance of GWO, GA and IWO algorithms in terms of % Deviation occurs in problems 3, 19 and 1, respectively. In Problem 3, the % Deviation for the GWO algorithm is 0.06, while the% Deviation for GA and IWO is 0.11 and 0.061 for Problems 19 and 1, respectively. Moreover, in terms of CPU (s), the best performance belongs to the GA algorithm with a value of 36.33 which is presented in problem 6. In the following, the best performance of IWO, GWO and HHO algorithms in terms of CPU (s) occurs in problems 26, 1 and 7, respectively. In problem 26, the % Deviation for the IWO algorithm is 39.31, while the CPU (s) for GWO and HHO are 54.74 and 57.43 for problems 1 and 7, respectively.

Overall, it can be seen from Tables 16 and 17 that GA has on average less CPU (s) than other algorithms. In fact, the GA algorithm on average requires less time to solve problems. By contrast, the HHO algorithm has on average less % Deviation compared to other algorithms. This means that the algorithm provides more accurate solutions on average.

With all that was mentioned, results demonstrate appropriate performance of all four suggested algorithms regarding convergence to objective function values and there is no significant difference between the precise objective function values and outputs of the four presented algorithms. In addition, Fig. 19 depicts the convergence curve of four algorithms for problem 15.







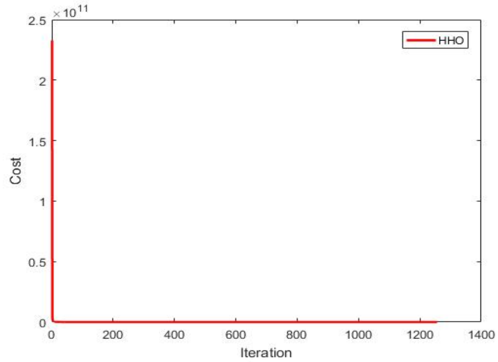


Fig. 19. Convergence curves of the GA, IWO, GWO and HHO.

Table 16. The outcome yielded by BARON, GA, IWO, GWO and HHO algorithms

Problem	Dimension	Baron	GA			IWO		
		OBJ values	OBJ value	CPU (s)	% Deviation	OBJ values	CPU (s)	% Deviation
1	2x1	237841.89	238972.66	39.79	0.48	239284.40	66.37	0.61
2	2x2	171206.70	171759.91	39.67	0.32	172826.93	77.46	0.95
3	2x3	169640.95	170710.86	41.83	0.63	171564.45	69.47	1.13
4	2x3	207644.19	208095.44	41.82	0.22	211082.94	68.85	1.66
5	2x4	221564.32	221869.90	44.78	0.14	225088.50	79.41	1.59
6	3x1	265088.23	265753.09	36.33	0.25	268882.12	63.25	1.43
7	3x1	280765.95	282845.62	37.07	0.74	283518.16	61.99	0.98
8	3x2	303198.80	305777.36	39.96	0.85	307828.26	57.57	1.53
9	3x2	324731.83	328149.08	40.39	1.05	329352.23	59.07	1.42
10	3x2	301323.15	302813.37	40.46	0.49	304409.57	58.11	1.02
11	3x3	336207.72	337498.91	42.05	0.38	340336.85	73.46	1.23
12	3x3	276771.22	279089.70	42.57	0.84	280230.40	74.05	1.25
13	3x4	328499.61	331499.83	44.03	0.91	333820.86	70.22	1.62
14	3x4	297201.39	300457.86	45.29	1.10	301593.12	71.72	1.48
15	3x4	284224.63	286728.80	44.86	0.88	287955.98	70.06	1.31
16	4x1	370709.37	375466.21	38.93	1.28	375803.11	56.18	1.37
17	4x1	413058.11	413751.36	40.38	0.17	418809.39	57.45	1.39
18	4x2	399163.56	402188.96	42.04	0.76	404634.81	51.37	1.37
19	4x2	419758.09	420216.71	39.94	0.11	426768.58	52.26	1.67
20	4x2	360574.58	364318.34	40.33	1.04	365408.85	50.05	1.34
21	4x3	484666.19	489168.05	42.19	0.93	492914.52	54.79	1.70
22	4x3	413873.91	417084.99	42.64	0.78	420247.29	56.91	1.54
23	4x3	390066.85	396350.25	44.70	1.61	396474.10	56.82	1.64
24	4x4	537675.86	542884.44	45.19	0.97	543202.54	59.01	1.03
25	4x4	448469.16	453795.85	44.79	1.19	456887.55	60.35	1.88
26	5x1	574664.35	577988.61	37.94	0.58	583349.17	39.31	1.51
27	5x2	510338.51	514718.23	41.39	0.86	519034.51	41.46	1.70
28	5x2	464007.48	468630.49	41.75	1.00	473691.69	42.41	2.09
29	5x3	627778.09	632512.31	44.04	0.75	637549.68	48.07	1.56
30	5x4	540780.55	547633.02	45.46	1.27	546792.72	53.82	1.11
$\overline{X}$		365383.17	368291.01	41.75	0.75	370644.78	60.04	1.40

Table 17. The outcome yielded by BARON, GA, IWO, GWO and HHO algorithms (continued).

Problem	Dimension	Baron	GWO			HHO		
		OBJ values	OBJ values	CPU (s)	% Deviation	OBJ values	CPU (s)	% Deviation
1	2x1	237841.89	240143.34	54.74	0.97	238797.95	58.82	0.40
2	2x2	171206.70	171839.47	59.48	0.37	171696.70	63.43	0.29
3	2x3	169640.95	169743.98	62.66	0.06	170492.95	65.47	0.50
4	2x3	207644.19	208338.91	62.29	0.33	208003.03	65.45	0.17
5	2x4	221564.32	222992.26	66.68	0.64	222996.50	69.09	0.65
6	3x1	265088.23	267714.86	56.71	0.99	265296.60	57.74	0.08
7	3x1	280765.95	282721.71	55.46	0.70	281478.72	57.43	0.25
8	3x2	303198.80	308968.21	59.97	1.90	303692.05	61.82	0.16
9	3x2	324731.83	327262.96	60.37	0.78	324809.93	62.81	0.02
10	3x2	301323.15	302557.01	60.82	0.41	302343.16	63.42	0.34
11	3x3	336207.72	337929.01	63.96	0.51	337459.66	66.98	0.37
12	3x3	276771.22	279058.22	64.45	0.83	277516.23	66.81	0.27
13	3x4	328499.61	330335.82	66.91	0.56	328605.62	71.05	0.03
14	3x4	297201.39	300915.71	67.82	1.25	298307.68	72.40	0.37
15	3x4	284224.63	285937.80	67.83	0.60	284974.74	69.55	0.26
16	4x1	370709.37	377331.74	56.89	1.79	371062.70	58.70	0.10
17	4x1	413058.11	419101.25	58.18	1.46	414101.69	58.93	0.25
18	4x2	399163.56	405048.49	62.24	1.47	399241.11	65.29	0.02
19	4x2	419758.09	423636.76	62.55	0.92	420354.70	63.90	0.14
20	4x2	360574.58	363205.83	62.17	0.73	361728.90	64.44	0.32
21	4x3	484666.19	487798.21	65.85	0.65	485245.42	67.26	0.12
22	4x3	413873.91	417889.46	66.36	0.97	414507.76	67.90	0.15
23	4x3	390066.85	393453.56	65.96	0.87	390458.51	68.39	0.10
24	4x4	537675.86	539898.58	70.44	0.41	537989.31	72.19	0.06
25	4x4	448469.16	456560.43	68.44	1.80	449531.69	71.78	0.24
26	5x1	574664.35	580031.88	58.13	0.93	575335.67	61.09	0.12
27	5x2	510338.51	517686.97	63.25	1.44	510734.85	65.47	0.08
28	5x2	464007.48	469662.79	64.45	1.22	464785.49	66.55	0.17
29	5x3	627778.09	631883.25	68.38	0.65	627979.13	68.43	0.03
30	5x4	540780.55	545646.35	69.78	0.90	541118.28	73.82	0.06
	$\overline{X}$	365383.17	368843.16	63.11	0.90	366021.56	65.55	0.20

According to Figure 19, since the algorithms were used to solve the problem after setting the parameter and selecting the optimal parameters, they show the best performance for solving the model and have a rapid convergence to the optimal solution. Furthermore, by comparing the convergence diagrams of the algorithms, it can be seen that the convergence diagrams of all four algorithms are almost smooth and fall rapidly. Therefore, it can be concluded that all four algorithms perform better in exploitation than in exploration (Ahmadianfar et al. [1]).

In spite of appropriate performance of four presented algorithms, the algorithms are ranked using the AHP-TOPSIS approach, which is a hybrid multi-criterion method of decision making. In this approach, the AHP method is used for determining the criteria's weights, and the TOPSIS method is employed in prioritizing the algorithms.

According to AHP method, the criteria's weights are determined as shown below:

Step1: Creating a matrix for pair-wise comparison

At this stage, the criteria are compared in pairs in order to create the pair – wise comparison matrix. The arrays of the pair – wise comparison matrix (a_{jk}) follow the Eq. (76).

$$a_{jk} \,.\, a_{kj} = 1 \quad ; \qquad j, k = 1,2, \dots, n \tag{76}$$

Obviously, $a_{jj} = 1$ for all j. The matrix created for the pair-wise comparison is showed in Table 18.

Table 18. The matrix of pair-wise comparison for the given criteria.

	OBJ values	CPU (s)	% Deviation
OBJ values	1	5	0.33
CPU (s)	0.20	1	0.14
% Deviation	3	7	1

Step2: Calculating the normalized pair-wise comparison matrix

To calculate the normalized pair – wise comparison matrix, each array of the matrix is divided by the sum of the corresponding column.

$$\bar{a}_{jk} = \frac{a_{jk}}{\sum_{l=1}^{n} a_{lk}} \quad ; \qquad j, k = 1,2, \dots, n \tag{77}$$

Where $\bar{a}_{jk}$ is array of normalized pair – wise comparison matrix.

Step3: Calculating the criteria weight vector

The weight vector of the criteria is obtained by averaging the normalized pair comparison matrix arrays of each row.

$$w_j = \frac{\sum_{l=1}^{n} \overline{a_{jl}}}{n} \quad ; \quad j = 1,2, \dots, n \tag{78}$$

Wherew_j is the weight of the $j - th$ criterion and$\sum_{j=1}^{n} w_j = 1$. The attributes weight vector is depicted in Table 19.

Table 19. The criteria weights.

Metric (Attribute)	Weight
OBJ values	0.28
CPU (s)	0.07
% Deviation	0.64

After obtaining the criteria's weight vector, the weight vector derived from the AHP method is considered as input to the TOPSIS method.

In the TOPSIS technique, which has been proposed by Hwang and Yoon [18], $m$alternatives are assessed by n criteria. In this model, the positive and negative ideal solutions are defined by the principle logic. The positive ideal solution yields an increase in the profit criteria and a reduction in the cost criteria. The optimum alternative is the one that has the shortest distance from the positive ideal solution and, meanwhile, the longest distance from the negative ideal solution. In other words, the alternatives that are ranked as the highest ones, are those that have the most closeness to the ideal solution. Below, the implementation of the TOPSIS method is described in detail.

Stage 1. Creating decision matrix

In the matrix of decision, the criteria are located in the columns and the alternatives are located in the row and each array of the matrix is the evaluation of each alternative relative to each criterion. It is necessary to note that the mean of the values of the objective function, the mean of CPU time(s), and the mean percentage deviation of the 30 problems solved are considered as decision-matrix arrays. Table 20 shows the decision matrix.

Table 20. The decision matrix.

	OBJ	CPU (s)	% Deviation
GA	368291.01	41.76	0.75
IWO	370644.78	60.04	1.40
GWO	368843.16	63.11	0.90
HHO	366021.56	65.55	0.20

Stage 2. Calculating the normalized decision matrix.

To obtain the normalized decision matrix, the arrays of the decision matrix are divided by the square root of the sum of square of the arrays of the corresponding column.

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}} \quad ; \quad i = 1,2,\dots,m\, ; \; j = 1,2,\dots,n \tag{79}$$

Where m, n, x_{ij} and r_{ij} are the number of alternatives, the number of the criteria, the array of the decision matrix and the array of the normalized decision matrix, respectively. Table 21depicts the normalized decision matrix for the number of alternatives and the criteria.

Table 21. The normalized decision matrix

	OBJ	CPU	% Deviation
GA	0.500	0.36	0.41
IWO	0.503	0.51	0.76
GWO	0.501	0.54	0.49
HHO	0.497	0.56	0.11

Stage 3. Calculating the weighted normalized decision matrix.

By multiplying the criteria's weight, which has been obtained via the AHP method, by the normalized decision matrix, the weighted normalized decision matrix is calculated.

$$v_{ij} = w_j \times r_{ij} \quad ; \quad i = 1,2,\dots,m; \; j = 1,2,\dots,n \tag{80}$$

Where $v_{ij}$is the array of the weighted normalized decision matrix. Table 22 represents the weighted normalized decision matrix.

Table 22. The weighted normalized decision matrix

	OBJ	**CPU**	**% Deviation**
GA	0.141	0.03	0.26
IWO	0.142	0.04	0.49
GWO	0.142	0.04	0.32
HHO	0.140	0.04	0.07

Stage 4. Specifying the positive ideal solution and negative ideal solution.

The type of criteria should be specified in this stage. The criteria have the positive or negative aspects. Positive criteria are criteria whose increase improves the system, and the ideal positive solution equals the largest array in the column of the positive criteria. Negative criteria are criteria that their reduction improves the system and the ideal negative solution equals the largest array in the column of the negative criteria.

$$A^{+} = (v_1^{+}, v_2^{+}, \dots, v_n^{+}) = \left(\left(\underset{i}{Max}\, v_{ij} \,\middle|\, j \in I \right), \left(\underset{i}{Min}\, v_{ij} \,\middle|\, j \in J \right) \right) \quad (81)$$

$$A^{-} = (v_1^{-}, v_2^{-}, \dots, v_n^{-}) = \left(\left(\underset{i}{Min}\, v_{ij} \,\middle|\, j \in I \right), \left(\underset{i}{Max}\, v_{ij} \,\middle|\, j \in J \right) \right) \quad (82)$$

Where$I$ is related to the benefit criteria and $J$to the cost criteria, $i = 1,2, \dots, m, j = 1,2, \dots, n$.

Stage 5. Calculating the measures of separation from the positive and negative ideal solutions.

Distance between each positive and negative ideal and each of the alternatives can be calculated using the Eqs.(83) and (84):

$$d_i^{+} = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^{+})^2} \quad ; \quad i = 1,2,\dots,m \quad (83)$$

$$d_i^{-} = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^{-})^2} \quad ; \quad i = 1,2,\dots,m \quad (84)$$

Where d_i^{+} and d_i^{-}indicate the distances between the$i - th$ alternative and the positive and negative ideal solutions, $v_j{}^{+}$and $v_j{}^{-}$represent the positive ideal and negative ideal solutions, and v_{ij}indicates the weighted normalized matrix of decision.

Stage 6. Calculating the relative closeness to the positive ideal solution.

The relative closeness represents the score of each alternative and is calculated based on the Eq. (85). The closer the index is to one, the better it is considered.

$$cl_i = \frac{d_i^{-}}{d_i^{+} + d_i^{-}} \quad ; \quad i = 1,2, \dots, m \quad (85)$$

Table 23 shows the results of the TOPSIS method. As indicated by the obtained results, the HHO algorithm exhibits the highest efficiency in solving problems.

Table 23. The result of the TOPSIS method.

	d_i^{+}	d_i^{-}	cl_i	**Rank**
GA	0.1913	0.2280	0.544	2
IWO	0.4189	0.0035	0.008	4
GWO	0.2448	0.1744	0.416	3
HHO	0.015	0.4187	0.965	1

6. Conclusion and future research

In classical production and inventory models, it is often assumed that the items produced are of perfect quality and there are no restrictions on production and inventory processes. However, in the real world, it is possible for the production process to be incomplete and for items of poor quality to be produced. Therefore, the implementation of quality control policies in production systems is essential to identify defective items. The study proposed a multi-product single-machine production-inventory system, considering the possibility of producing defective items. In this study, all items were inspected during the production process and were divided into three categories of non-defective items, scrapped items and defective items according to the inspector. In this regard, non-defective items were sold, scrapped items was deposed, and defective items were reworked based on the severity of the failure. Since the proposed model considered the possibility of inspection error during the production process as well as the return policy, defective items that had been sold due to the inspector's error in identifying them could be returned and entered the rework process in the next cycle. Also, four constraints including machine capacity, service level, warehouse space and budget that most production-inventory systems face, were considered to provide a more practical model for the model.

The machine capacity constraint was imposed on the model due to the fact that the manufacturing-inventory system is single-machine and showed that the existence of only one machine leads to the limitation of production capacity. Service level constraints ensured that the ratio of demand shortages for each product did not exceed a certain amount per year. The storage space constraint provided conditions that the space required for storing items did not exceed the total available space. Budget constraint was also added to the model due to the limited budget available and the possibility of covering the costs of the manufacturing-inventory system.

The presented model as a nonlinear programming problem was solved using the GAMS software and GA, IWO, GWO and HHO meta-heuristic algorithms. The parameters of the proposed algorithms were calibrated by RSM method for minimizing the total cost of the manufacturing-inventory system, which included the setup, production, rework, holding, backorder, disposal, warehouse construction, inspection, penalty and return costs. Ultimately, several problems were examined aiming to show the proposed model's efficiency as well as the efficiency of the solution methodologies. The obtained results indicated the desirable efficiency of the solution methodologies regarding the total cost, objective function's percentage deviation, and CPU time. Besides, based on results of AHP-TOPSIS method, HHO algorithm had the best performance in comparison with the other three algorithms.

Overall, the study can help managers make decisions in different parts of the company. In this regard, production managers can produce several products with one machine in order to increase the variety of products and attract more customers. Furthermore, since high quality product is an important feature for customers, quality managers can implement quality control policies to improve the quality of manufactured products. Eventually, inventory managers will be able to ensure that the space required to store items do not exceed the total available space and that the amount of shortage is optimal.

The following suggestions can be considered for future research:

- Interruption and breakdown in production process can be added to the model.
- Some of the parameters can be assumed as being characterized by features of fuzziness and stochasticity or even a combination of both of these two features.
- Various maintenance policies in imperfect production process can be considered.
- A discount strategy can be employed to the problem.
- Rework process can be performed at synchronous and asynchronous flexible rates.
- Investigating the effect of learning during the manufacturing process on the proposed model.

Acknowledgements

The authors wish to thank the Associate Editor and the two anonymous reviewers for their critical and constructive comments.

Appendix

Appendix A: Determining the length of the cycle.

In accordance with Eq.(8), we will have:

$$T = t_i^0 + t_i^1 + t_i^2 + \cdots + t_i^{m+1} + t_i^{m+2} + t_i^{m+3} \quad (A.1)$$

$$T = \frac{B_i}{\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]P_i - D_i} + \frac{Q_i}{P_i} - \frac{B_i}{\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]P_i - D_i} + [\alpha_i^1 + (1-\sigma_i)e1_i^1]\frac{Q_i}{V_i^1 P_i} + [\alpha_i^2 + (1-\sigma_i)e1_i^2]\frac{Q_i}{V_i^2 P_i} + \dots + [\alpha_i^m + (1-\sigma_i)e1_i^m]\frac{Q_i}{V_i^m P_i} + \frac{H_i^m}{D_i} + \frac{B_i}{D_i} \tag{A.2}$$

$$T = \frac{Q_i}{P_i} + [\alpha_i^1 + (1-\sigma_i)e1_i^1]\frac{Q_i}{V_i^1 P_i} + [\alpha_i^2 + (1-\sigma_i)e1_i^2]\frac{Q_i}{V_i^2 P_i} + \dots + [\alpha_i^m + (1-\sigma_i)e1_i^m]\frac{Q_i}{V_i^m P_i} + \frac{\left[\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]P_i - D_i\right]\left(\frac{Q_i}{P_i}\right) - B_i}{D_i} + \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1](\gamma_i^1 V_i^1 P_i - D_i)\frac{Q_i}{V_i^1 P_i}}{D_i} + \frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2](\gamma_i^2 V_i^2 P_i - D_i)\frac{Q_i}{V_i^2 P_i}}{D_i} + \dots + \frac{[\alpha_i^m + (1-\sigma_i)e1_i^m](\gamma_i^m V_i^m P_i - D_i)\frac{Q_i}{V_i^m P_i}}{D_i} + \frac{B_i}{D_i} \tag{A.3}$$

$$T = \frac{Q_i}{P_i} + \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]Q_i}{V_i^1 P_i} + \frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]Q_i}{V_i^2 P_i} + \dots + \frac{[\alpha_i^m + (1-\sigma_i)e1_i^m]Q_i}{V_i^m P_i} + \frac{\left[\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]P_i - D_i\right]Q_i}{P_i D_i} - \frac{B_i}{D_i} + \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1](\gamma_i^1 V_i^1 P_i - D_i)Q_i}{V_i^1 P_i D_i} + \frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2](\gamma_i^2 V_i^2 P_i - D_i)Q_i}{V_i^2 P_i D_i} + \dots + \frac{[\alpha_i^m + (1-\sigma_i)e1_i^m](\gamma_i^m V_i^m P_i - D_i)Q_i}{V_i^m P_i D_i} + \frac{B_i}{D_i} \tag{A.4}$$

$$T = \frac{Q_i}{P_i} + \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]Q_i}{V_i^1 P_i} + \frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]Q_i}{V_i^2 P_i} + \dots + \frac{[\alpha_i^m + (1-\sigma_i)e1_i^m]Q_i}{V_i^m P_i} + \frac{\left[\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] - \left(\frac{D_i}{P_i}\right)\right]P_i Q_i}{D_i P_i} - \frac{B_i}{D_i} + \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]\gamma_i^1 V_i^1 P_i Q_i}{V_i^1 P_i D_i} - \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]D_i Q_i}{V_i^1 P_i D_i} + \frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]\gamma_i^2 V_i^2 P_i Q_i}{V_i^2 P_i D_i} - \frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]D_i Q_i}{V_i^2 P_i D_i} + \dots + \frac{[\alpha_i^m + (1-\sigma_i)e1_i^m]\gamma_i^m V_i^m P_i Q_i}{V_i^m P_i D_i} - \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^m]D_i Q_i}{V_i^m P_i D_i} + \frac{B_i}{D_i} \tag{A.5}$$

$$T = \frac{Q_i}{P_i} + \frac{\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]Q_i}{D_i} - \frac{Q_i}{P_i} + \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]\gamma_i^1 Q_i}{D_i} + \frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]\gamma_i^2 Q_i}{D_i} + \dots + \frac{[\alpha_i^m + (1-\sigma_i)e1_i^m]\gamma_i^m Q_i}{D_i} \tag{A.6}$$

$$T = \frac{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^m \gamma_i^j\left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]\right)Q_i}{D_i} \tag{A.7}$$

Appendix B. Calculating the cost of holding.

Considering Eq. (23), we have:

$$CH = CH_a + CA_b + CH_c + CH_d \tag{B.1}$$

According to Eq. (24), CH_a is:

$$CH_a = \frac{1}{2T}\sum_{i=1}^{n} h_i \Bigg[I_i\left(\frac{Q_i}{P_i} - \frac{B_i}{a_i}\right) + \left(2I_i + [\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1 \frac{Q_i}{V_i^1\, P_i}\right)\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]Q_i}{V_i^1\, P_i}\right)$$
$$+ \left(2H_i^1 + [\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1 \frac{Q_i}{V_i^1\, P_i}\right)\left(\frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]Q_i}{V_i^2\, P_i}\right) + \dots \tag{B.2}$$
$$+ \left(2H_i^{m-1} + [\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1 \frac{Q_i}{V_i^1\, P_i}\right)\left(\frac{[\alpha_i^m + (1-\sigma_i)e1_i^m]Q_i}{V_i^m\, P_i}\right) + \frac{(H_i^m)^2}{D_I}\Bigg]$$

Based on Eqs. (9) – (12) we have:

$$I_i = a_i\left(\frac{Q_i}{P_i}\right) - B_i \tag{B.1}$$

$$(I_i)^2 = \frac{(a_i)^2 (Q_i)^2}{(P_i)^2} + (B_i)^2 - 2\left(\frac{a_i}{P_i}\right)Q_i B_i \tag{B.2}$$

$$(I_i)^2 = \left(\frac{a_i}{P_i}\right)^2 Q_i^{\,2} + (B_i)^2 - 2\left(\frac{a_i}{P_i}\right)Q_i B_i \tag{B.3}$$

and

$$H_i^1 = I_i + \frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\frac{Q_i}{P_i} \tag{B.4}$$

$$(H_i^1)^2 = (I_i)^2 + \left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)^2 \left(\frac{Q_i}{P_i}\right)^2 + 2\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\frac{Q_i}{P_i}\right)(I_i) \tag{B.5}$$

$$(H_i^1)^2 = \left(\frac{a_i}{P_i}\right)^2 Q_i^{\,2} + (B_i)^2 - 2\left(\frac{a_i}{P_i}\right)Q_i B_i + \left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)^2 \left(\frac{Q_i}{P_i}\right)^2$$
$$+ 2a_i\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)\left(\frac{Q_i}{P_i}\right)^2 - 2\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)\left(\frac{Q_i}{P_i}\right)B_i \tag{B.6}$$

Also

$$H_i^2 = H_i^1 + \frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]y_i^2}{V_i^2}\frac{Q_i}{P_i} \tag{B.7}$$

$$(H_i^2)^2 = (H_i^1)^2 + \left(\frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]y_i^2}{V_i^2}\right)^2 \left(\frac{Q_i}{P_i}\right)^2 + 2\left(\frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]y_i^2}{V_i^2}\right)\left(\frac{Q_i}{P_i}\right)(H_i^1) \tag{B.8}$$

$$(H_i^2)^2 = \left(\frac{a_i}{P_i}\right)^2 Q_i^{\,2} + (B_i)^2 - 2\left(\frac{a_i}{P_i}\right)Q_i B_i + \left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)^2 \left(\frac{Q_i}{P_i}\right)^2$$
$$+ 2a_i\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)\left(\frac{Q_i}{P_i}\right)^2 - 2\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)\left(\frac{Q_i}{P_i}\right)B_i$$
$$+ \left(\frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]y_i^2}{V_i^2}\right)^2 \left(\frac{Q_i}{P_i}\right)^2 + 2a_i\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)\left(\frac{Q_i}{P_i}\right)^2 \tag{B.9}$$
$$- 2\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1}{V_i^1}\right)\left(\frac{Q_i}{P_i}\right)B_i$$
$$+ 2\left(\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]y_i^1[\alpha_i^2 + (1-\sigma_i)e1_i^2]y_i^2}{V_i^1 V_i^2}\right)\left(\frac{Q_i}{P_i}\right)^2$$

Therefore

$$(H_i^m)^2 = \left(\left(\frac{a_i}{P_i}\right)^2 Q_i^{\ 2} + (B_i)^2 - 2\left(\frac{a_i}{P_i}\right) Q_i B_i + \left(\frac{Q_i}{P_i}\right)^2 + \sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right)^2 + 2a_i\left(\frac{Q_i}{P_i}\right)^2 \sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right)^2\right) \tag{B.10}$$

Then, based on Eqs. (2) – (6), we have:

$$CH_a = \sum_{i=1}^{n}\Bigg(\left(\frac{h_i a_i}{2}\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2\right)(T) +$$
$$\left(h_i a_i\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2 \sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j}\right)\right)(T) +$$
$$\left(\frac{h_i}{2}\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2 \sum_{i=1}^{m}\left(\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j}\right)^2 y_i^j\right)\right)(T) +$$
$$\left(h_i\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2 \sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j} + \sum_{k=1}^{j-1}\left(\frac{\left[\alpha_i^k+(1-\sigma_i)e1_i^k\right]}{V_i^k}y_i^k\right)\right)\right)(T) +$$
$$\left(\frac{h_i D_i}{2}\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2\right)(T) +$$
$$\left(\frac{a_i h_i D_i}{\left(P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)\right)^2}\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right)\right)(T) +$$
$$\left(\frac{h_i D_i}{2}\left(\frac{1}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2 \sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j}\right)^2\right)(T) +$$
$$\left(\frac{h_i D_i}{\left(P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)\right)^2}\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j} + \sum_{k=1}^{j-1}\left(\frac{\left[\alpha_i^k+(1-\sigma_i)e1_i^k\right]}{V_i^k}y_i^k\right)\right)^2\right)(T) +$$
$$\frac{h_i}{2D_i}\left(\frac{(B_i)^2}{T}\right) + \frac{h_i}{2a_i}\left(\frac{(B_i)^2}{T}\right) - \left(\frac{h_i a_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)(B_i) -$$
$$\left(\frac{h_i}{P_i(1-\theta_i)}\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right)\right)(B_i) - \left(\frac{h_i D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)(B_i) -$$
$$\left(\left(\frac{h_i D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right)^2\right)(B_i)\Bigg) \tag{B.11}$$

According to Eq. (25), CH_b is:

$$CH_b = \frac{1}{2T}\sum_{i=1}^{n} h_i\Big[\left((1-\sigma_i)e1_i + \alpha_i\left(1 - e2_i^j\right) + \theta_i\right)Q_i\,(t_i^0 + t_i^1) + \left((1-\sigma_i)e1_i^j + \alpha_i^1\right)(1-\gamma_i^1)Q_i t_i^2 + \left((1-\sigma_i)e1_i^j + \alpha_i^2\right)(1-\gamma_i^2)Q_i t_i^3 + \ldots + \left((1-\sigma_i)e1_i^j + \alpha_i^m\right)(1-\gamma_i^m)Q_i t_i^{m+1}\Big] \tag{B.12}$$

Base on Eqs. (1) – (5) we have:

$$CH_b = \frac{1}{2T}\sum_{i=1}^{n} h_i \Bigg[\Big(\big(1-\sigma_i\big)e1_i + \alpha_i\big(1 - e2_i^j\big) + \theta_i\Big)\frac{{Q_i}^2}{P_i} + \Big(\big(1-\sigma_i\big)e1_i^1 + \alpha_i^1\Big)\big(1-\gamma_i^1\big)\frac{\big[\alpha_i^1 + (1-\sigma_i)e1_i^1\big]{Q_i}^2}{V_i^1\, P_i} + \Big(\big(1-\sigma_i\big)e1_i^2 + \alpha_i^2\Big)\big(1-\gamma_i^2\big)\frac{\big[\alpha_i^2 + (1-\sigma_i)e1_i^2\big]{Q_i}^2}{V_i^2\, P_i} + \ldots + \Big(\big(1-\sigma_i\big)e1_i^m + \alpha_i^m\Big)\big(1-\gamma_i^m\big)\frac{\big[\alpha_i^m + (1-\sigma_i)e1_i^m\big]{Q_i}^2}{V_i^m\, P_i}\Bigg] \tag{B.13}$$

$$CH_b = \frac{1}{2T}\sum_{i=1}^{n} h_i \Bigg[\Big(\big(1-\sigma_i\big)e1_i + \alpha_i\big(1 - e2_i^j\big) + \theta_i\Big)\frac{{Q_i}^2}{P_i} + \Big(\big(1-\sigma_i\big)e1_i^1 + \alpha_i^1\Big)^2\big(1-\gamma_i^1\big)\frac{{Q_i}^2}{V_i^1\, P_i} + \Big(\big(1-\sigma_i\big)e1_i^2 + \alpha_i^2\Big)^2\big(1-\gamma_i^2\big)\frac{{Q_i}^2}{V_i^2\, P_i} + \ldots + \Big(\big(1-\sigma_i\big)e1_i^m + \alpha_i^m\Big)^2\big(1-\gamma_i^m\big)\frac{{Q_i}^2}{V_i^m\, P_i}\Bigg] \tag{B.14}$$

$$CH_b = \frac{1}{2T}\sum_{i=1}^{n} \frac{h_i\, {Q_i}^2}{P_i}\Bigg[\Big(\big(1-\sigma_i\big)e1_i + \alpha_i\big(1 - e2_i^j\big) + \theta_i\Big) + \frac{\Big(\big(1-\sigma_i\big)e1_i^1 + \alpha_i^1\Big)^2\big(1-\gamma_i^1\big)}{V_i^1} + \frac{\Big(\big(1-\sigma_i\big)e1_i^2 + \alpha_i^2\Big)^2\big(1-\gamma_i^2\big)}{V_i^2} + \ldots + \frac{\Big(\big(1-\sigma_i\big)e1_i^m + \alpha_i^m\Big)^2\big(1-\gamma_i^m\big)}{V_i^m}\Bigg] \tag{B.15}$$

Using Eq. (14), we have:

$$CH_b = \frac{T}{2}\sum_{i=1}^{n} \frac{h_i\, {D_i}^2}{\Big\{\big[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\big] + \sum_{j=1}^{m}\gamma_i^j\big[\alpha_i^j + (1-\sigma_i)e1_i^j\big]\Big\}^2}\Bigg[\Big(\big(1-\sigma_i\big)e1_i + \alpha_i\big(1 - e2_i^j\big) + \theta_i\Big) + \frac{\Big(\big(1-\sigma_i\big)e1_i^1 + \alpha_i^1\Big)^2\big(1-\gamma_i^1\big)}{V_i^1} + \frac{\Big(\big(1-\sigma_i\big)e1_i^2 + \alpha_i^2\Big)^2\big(1-\gamma_i^2\big)}{V_i^2} + \ldots + \frac{\Big(\big(1-\sigma_i\big)e1_i^m + \alpha_i^m\Big)^2\big(1-\gamma_i^m\big)}{V_i^m}\Bigg] \tag{B.16}$$

According to Eq. (26), CH_c is:

$$CH_c = \frac{1}{2}\sum_{i=1}^{n} h_i\big[\alpha_i e2_i^j Q_i\big] \tag{B.17}$$

Using Eq. (14), we have:

$$CH_c = \frac{T}{2}\sum_{i=1}^{n} h_i\left[\frac{\alpha_i e2_i^j D_i}{\Big(\big[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\big] + \sum_{j=1}^{m}\gamma_i^j\big[\alpha_i^j + (1-\sigma_i)e1_i^j\big]\Big)}\right] \tag{B.18}$$

According to Eq. (27), CH_d is:

$$CH_d = \frac{1}{2T}\sum_{i=1}^{n} h_i \left[\left[\left((1-\sigma_i)e1_i + \alpha_i\right) + \left(\sum_{j=2}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=2}^{m} e1_i^j\right)\right] Q_i t_i^2 + \left[\left(\sum_{j=2}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=2}^{m} e1_i^j\right) + \left(\sum_{j=3}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=3}^{m} e1_i^j\right)\right] Q_i t_i^3 + \cdots + \left[\sum_{j=m}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=m}^{m} e1_i^j\right] Q_i t_i^{m+1}\right] \tag{B.19}$$

According to Eqs. (3) – (5), we have:

$$CH_d = \frac{1}{2T}\sum_{i=1}^{n} h_i \left[\left[\left((1-\sigma_i)e1_i + \alpha_i\right) + \left(\sum_{j=2}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=2}^{m} e1_i^j\right)\right] [\alpha_i^1 + (1-\sigma_i)e1_i^1]\frac{{Q_i}^2}{V_i^1\, P_i} + \left[\left(\sum_{j=2}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=2}^{m} e1_i^j\right) + \left(\sum_{j=3}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=3}^{m} e1_i^j\right)\right] [\alpha_i^2 + (1-\sigma_i)e1_i^2]\frac{{Q_i}^2}{V_i^2\, P_i} + \cdots + \left[\sum_{j=m}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=m}^{m} e1_i^j\right] [\alpha_i^m + (1-\sigma_i)e1_i^m]\frac{{Q_i}^2}{V_i^m\, P_i}\right] \tag{B.20}$$

Then, using Eq. (14), we have:

$$CH_d = \frac{T}{2}\sum_{i=1}^{n} \frac{h_i {D_i}^2}{\left(\left[(1-\sigma_i)(1-e1_i) + \alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j + (1-\sigma_i)e1_i^j\right]\right)^2} \left[\left[\left((1-\sigma_i)e1_i + \alpha_i\right) + \left(\sum_{j=2}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=2}^{m} e1_i^j\right)\right]\frac{[\alpha_i^1 + (1-\sigma_i)e1_i^1]}{V_i^1} + \left[\left(\sum_{j=2}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=2}^{m} e1_i^j\right) + \left(\sum_{j=3}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=3}^{m} e1_i^j\right)\right]\frac{[\alpha_i^2 + (1-\sigma_i)e1_i^2]}{V_i^2} + \cdots + \left[\sum_{j=m}^{m}\alpha_i^j + (1-\sigma_i)\sum_{j=m}^{m} e1_i^j\right]\frac{[\alpha_i^m + (1-\sigma_i)e1_i^m]}{V_i^m}\right] \tag{B.21}$$

Therefore

$$
\begin{aligned}
&CH \\
&= \sum_{i=1}^{n}\left(\left(\frac{h_i a_i}{2}\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2\right)(T)\right. \\
&+\left(h_i a_i\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j}\right)\right)(T) \\
&+\left(\frac{h_i}{2}\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2\sum_{i=1}^{m}\left(\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j}\right)^2 y_i^j\right)\right)(T) \\
&+\left(h_i\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j}\right.\right. \\
&\left.\left.+\sum_{k=1}^{j-1}\left(\frac{\left[\alpha_i^k+(1-\sigma_i)e1_i^k\right]}{V_i^k}y_i^k\right)\right)\right)(T) \\
&+\left(\frac{h_i D_i}{2}\left(\frac{D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2\right)(T) \\
&+\left(\frac{a_i h_i D_i}{\left(P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)\right)^2}\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right)\right)(T) \\
&+\left(\frac{h_i D_i}{2}\left(\frac{1}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)^2\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j}\right)^2\right)(T) \\
&+\left(\frac{h_i D_i}{\left(P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)\right)^2}\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]}{V_i^j}\right.\right. \\
&\left.\left.+\sum_{k=1}^{j-1}\left(\frac{\left[\alpha_i^k+(1-\sigma_i)e1_i^k\right]}{V_i^k}y_i^k\right)\right)^2\right)(T) + \frac{h_i}{2D_i}\left(\frac{(B_i)^2}{T}\right) + \frac{h_i}{2a_i}\left(\frac{(B_i)^2}{T}\right) \\
&-\left(\frac{h_i a_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)(B_i) \\
&-\left(\frac{h_i}{P_i\,(1-\theta_i)}\sum_{j=1}^{m}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right)\right)(B_i) \\
&-\left(\frac{h_i D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)(B_i) \\
&\left.-\left(\left(\frac{h_i D_i}{P_i\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right)\sum_{j=1}^{\boldsymbol{m}}\left(\frac{\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]y_i^j}{V_i^j}\right)^2\right)(B_i)\right) \\
&+\frac{T}{2}\sum_{i=1}^{n}\frac{h_i\,{D_i}^2}{\left\{\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right] + \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right\}^2}\left[\left((1-\sigma_i)e1_i + \alpha_i\left(1- e2_i^j\right)+\theta_i\right)\right. \\
&\left.+\frac{\left((1-\sigma_i)e1_i^1 + \alpha_i^1\right)^2(1-\gamma_i^1)}{V_i^1} + \frac{\left((1-\sigma_i)e1_i^2 + \alpha_i^2\right)^2(1-\gamma_i^2)}{V_i^2} + \ldots + \frac{\left((1-\sigma_i)e1_i^m + \alpha_i^m\right)^2(1-\gamma_i^m)}{V_i^m}\right]
\end{aligned}
$$

$$+\ \frac{T}{2}\sum_{i=1}^{n} h_i\left[\frac{\alpha_i e2_i^j D_i}{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]\ +\ \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)}\right]$$

$$+\ \frac{T}{2}\sum_{i=1}^{n}\frac{h_i {D_i}^2}{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]\ +\ \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)^2}\left[\left[\left((1-\ \sigma_i)e1_i+\ \alpha_i\right)\right.\right.$$

$$\left.+\left(\sum_{j=2}^{m}\alpha_i^j\ +\ (1-\ \sigma_i)\sum_{j=2}^{m}e1_i^j\right)\right]\frac{\left[\alpha_i^1+(1-\sigma_i)e1_i^1\right]}{V_i^1}$$

$$+\left[\left(\sum_{j=2}^{m}\alpha_i^j\ +\ (1-\ \sigma_i)\sum_{j=2}^{m}e1_i^j\right)+\left(\sum_{j=3}^{m}\alpha_i^j\ +\ (1-\ \sigma_i)\sum_{j=3}^{m}e1_i^j\right)\right]\frac{\left[\alpha_i^2+(1-\sigma_i)e1_i^2\right]}{V_i^2}+\cdots$$

$$\left.+\left[\sum_{j=m}^{m}\alpha_i^j\ +\ (1-\ \sigma_i)\sum_{j=m}^{m}e1_i^j\right]\frac{\left[\alpha_i^m+(1-\sigma_i)e1_i^m\right]}{V_i^m}\right] \qquad \text{(B.22)}$$

Appendix C: Determining the machine capacity constraint

$$\sum_{i=1}^{n}\left(t_i^0+t_i^1+t_i^2+t_i^3+\ldots+t_i^{m+1}\right)+\sum_{i=1}^{n}S_i\ \le T \qquad \text{(C.1)}$$

Based on Eqs (1) - (5), we have:

$$\sum_{i=1}^{n}\left(\frac{Q_i}{P_i}+\frac{\left[\alpha_i^1+(1-\sigma_i)e1_i^1\right]Q_i}{V_i^1\,P_i}+\frac{\left[\alpha_i^2+(1-\sigma_i)e1_i^2\right]Q_i}{V_i^2\,P_i}+\cdots+\ \frac{\left[\alpha_i^m+(1-\sigma_i)e1_i^m\right]Q_i}{V_i^m\,P_i}\right)+\sum_{i=1}^{n}S_i\ \le T \qquad \text{(C.2)}$$

$$\sum_{i=1}^{n}\frac{Q_i}{P_i}\left(1+\frac{\left[\alpha_i^1+(1-\sigma_i)e1_i^1\right]}{V_i^1}+\frac{\left[\alpha_i^2+(1-\sigma_i)e1_i^2\right]}{V_i^2}+\cdots+\ \frac{\left[\alpha_i^m+(1-\sigma_i)e1_i^m\right]}{V_i^m}\right)+\sum_{i=1}^{n}S_i\ \le T \qquad \text{(C.3)}$$

By Inserting Eq. (14), the Machine capacity constraint is determined as follow.

$$\sum_{i=1}^{n}\frac{D_i\,T}{\left(\left[(1-\sigma_i)(1-e1_i)+\alpha_i e2_i^j\right]\ +\ \sum_{j=1}^{m}\gamma_i^j\left[\alpha_i^j+(1-\sigma_i)e1_i^j\right]\right)P_i}\left(1+\frac{\left[\alpha_i^1+(1-\sigma_i)e1_i^1\right]}{V_i^1}\right.$$
$$\left.+\ \frac{\left[\alpha_i^2+(1-\sigma_i)e1_i^2\right]}{V_i^2}+\cdots+\ \frac{\left[\alpha_i^m+(1-\sigma_i)e1_i^m\right]}{V_i^m}\right)+\sum_{i=1}^{n}S_i\ \le T \qquad \text{(C.4)}$$

Declarations

Conflict of interest: The authors declare that they have no conflict of interest.

References

[1] Ahmadianfar, I., Bozorg-Haddad, O., & Chu, X. (2020). Gradient-based optimizer: A new Metaheuristic optimization algorithm. Information Sciences, 540, 131-159.

[2] Al-Salamah, M. (2019). Economic production quantity in an imperfect manufacturing process with synchronous and asynchronous flexible rework rates. Operations Research Perspectives, 6, 100103.

[3] Barzoki, M. R., Jahanbazi, M., & Bijari, M. (2011). Effects of imperfect products on lot sizing with work in process inventory. Applied Mathematics and Computation, 217(21), 8328-8336.

[4] Beheshti Fakher, H., Nourelfath, M., & Gendreau, M. (2017). A cost minimisation model for joint production and maintenance planning under quality constraints. International Journal of Production Research, 55(8), 2163-2176.

[5] Cheikhrouhou, N., Sarkar, B., Ganguly, B., Malik, A. I., Batista, R., & Lee, Y. H. (2018). Optimization of sample size and order size in an inventory model with quality inspection and return of defective items. Annals of Operations Research, 271(2), 445-467.

[6] Chiu, S. W., Chiu, Y. S. P., & Yang, J. C. (2012). Combining an alternative multi-delivery policy into economic production lot size problem with partial rework. Expert Systems with Applications, 39(3), 2578-2583.
[7] Chiu, S. W., Wang, S. L., & Chiu, Y. S. P. (2007). Determining the optimal run time for EPQ model with scrap, rework, and stochastic breakdowns. European Journal of Operational Research, 180(2), 664-676.
[8] Chiu, Y. P. (2003). Determining the optimal lot size for the finite production model with random defective rate, the rework process, and backlogging. Engineering optimization, 35(4), 427-437.
[9] Chiu, Y. S. P., Chen, K. K., Cheng, F. T., & Wu, M. F. (2010). Optimization of the finite production rate model with scrap, rework and stochastic machine breakdown. Computers & mathematics with applications, 59(2), 919-932.
[10] Fadlil, I. N., Novitasari, R., & Jauhari, W. A. (2020, April). Sustainable economic production quantity model with rework and product return policy. In AIP Conference Proceedings (Vol. 2217, No. 1, p. 030078). AIP Publishing LLC.
[11] Faris, H., Aljarah, I., Al-Betar, M. A., & Mirjalili, S. (2018). Grey wolf optimizer: a review of recent variants and applications. Neural computing and applications, 30(2), 413-435.
[12] Gharaei, A., Hoseini Shekarabi, S. A., & Karimi, M. (2019). Modelling and optimal lot-sizing of the replenishments in constrained, multi-product and bi-objective EPQ models with defective products: Generalised cross decomposition. International Journal of Systems Science: Operations & Logistics, 1-13.
[13] Haji, A., Sikari, S. S., & Shamsi, R. (2009). The effect of inspection errors on the optimal batch size in reworkable production systems with scraps. International Journal of Product Development, 10(1-3), 201-216.
[14] Hayek, P. A., & Salameh, M. K. (2001). Production lot sizing with the reworking of imperfect quality items produced. Production planning & control, 12(6), 584-590.
[15] Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. Future generation computer systems, 97, 849-872.
[16] Hou, K. L. (2007). An EPQ model with setup cost and process quality as functions of capital expenditure. Applied Mathematical Modelling, 31(1), 10-17.
[17] Hsu, J. T., & Hsu, L. F. (2013). Two EPQ models with imperfect production processes, inspection errors, planned backorders, and sales returns. Computers & Industrial Engineering, 64(1), 389-402.
[18] Hwang, C. L., & Yoon, K. (1981). Methods for multiple attribute decision making. In Multiple attribute decision making (pp. 58-191). Springer, Berlin, Heidelberg.
[19] Jamal, A. M. M., Sarker, B. R., & Mondal, S. (2004). Optimal manufacturing batch size with rework process at a single-stage production system. Computers & industrial engineering, 47(1), 77-89.
[20] Kang, C. W., Ramzan, M. B., Sarkar, B., & Imran, M. (2018). Effect of inspection performance in smart manufacturing system based on human quality control system. The International Journal of Advanced Manufacturing Technology, 94(9-12), 4351-4364.
[21] Kang, C. W., Ullah, M., & Sarkar, B. (2018). Optimum ordering policy for an imperfect single-stage manufacturing system with safety stock and planned backorder. The International Journal of Advanced Manufacturing Technology, 95(1-4), 109-120.
[22] Kang, C. W., Ullah, M., Sarkar, B., Hussain, I., & Akhtar, R. (2017). Impact of random defective rate on lot size focusing work-in-process inventory in manufacturing system. International Journal of Production Research, 55(6), 1748-1766.
[23] Khalilpourazari, S., Pasandideh, S. H. R., & Niaki, S. T. A. (2016). Optimization of multi-product economic production quantity model with partial backordering and physical constraints: SQP, SFS, SA, and WCA. Applied Soft Computing, 49, 770-791.
[24] Korda, N., Szorenyi, B., & Li, S. (2016, June). Distributed clustering of linear bandits in peer to peer networks. In International conference on machine learning (pp. 1301-1309). PMLR.
[25] Li, S., & Kar, P. (2015). Context-aware bandits. arXiv preprint arXiv:1510.03164.
[26] Li, S., Gentile, C., & Karatzoglou, A. (2016). Graph clustering bandits for recommendation. arXiv preprint arXiv:1605.00596.
[27] Li, S., Karatzoglou, A., & Gentile, C. (2016, July). Collaborative filtering bandits. In Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (pp. 539-548).
[28] Liao, G. L., Chen, Y. H., & Sheu, S. H. (2009). Optimal economic production quantity policy for imperfect process with imperfect repair and maintenance. European Journal of Operational Research, 195(2), 348-357.
[29] Mahadik, K., Wu, Q., Li, S., & Sabne, A. (2020, June). Fast distributed bandits for online recommendation systems. In Proceedings of the 34th ACM International Conference on Supercomputing (pp. 1-13).
[30] Mehrabian, A. R., & Lucas, C. (2006). A novel numerical optimization algorithm inspired from weed colonization. Ecological informatics, 1(4), 355-366.
[31] Mellouk, L., Aaroud, A., Boulmalf, M., Zine-Dine, K., & Benhaddou, D. (2020). Development and performance validation of new parallel hybrid cuckoo search–genetic algorithm. Energy Systems, 11(3), 729-751.
[32] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. Advances in engineering software, 69, 46-61.
[33] Nobil, A. H., Afshar Sedigh, A. H., & Cárdenas-Barrón, L. E. (2018). Multi-machine economic production quantity for items with scrapped and rework with shortages and allocation decisions. Scientia Iranica, 25(4), 2331-2346.

[34] Nobil, A. H., Sedigh, A. H. A., & Cárdenas-Barrón, L. E. (2016). A multi-machine multi-product EPQ problem for an imperfect manufacturing system considering utilization and allocation decisions. Expert Systems with Applications, 56, 310-319.
[35] Nobil, A. H., Sedigh, A. H. A., & Cárdenas-Barrón, L. E. (2020). A multiproduct single machine economic production quantity (EPQ) inventory model with discrete delivery order, joint production policy and budget constraints. Annals of Operations Research, 286(1), 265-301.
[36] Ouyang, L. Y., & Chang, C. T. (2013). Optimal production lot with imperfect production process under permissible delay in payments and complete backlogging. International Journal of Production Economics, 144(2), 610-617.
[37] Pasandideh, S. H. R., Niaki, S. T. A., & Gharaei, A. (2015b). Optimization of a multiproduct economic production quantity problem with stochastic constraints using sequential quadratic programming. Knowledge-Based Systems, 84, 98-107.
[38] Pasandideh, S. H. R., Niaki, S. T. A., Nobil, A. H., & Cárdenas-Barrón, L. E. (2015a). A multiproduct single machine economic production quantity model for an imperfect production system under warehouse construction cost. International Journal of Production Economics, 169, 203-214.
[39] Pirayesh, M., & Poormoaied, S. (2015). GPSO-LS algorithm for a multi-item EPQ model with production capacity restriction. Applied Mathematical Modelling, 39(17), 5011-5032.
[40] Porteus, E. L. (1986). Optimal lot sizing, process quality improvement and setup cost reduction. Operations research, 34(1), 137-144.
[41] Sarkar, B., & Saren, S. (2016). Product inspection policy for an imperfect production system with inspection errors and warranty cost. European Journal of Operational Research, 248(1), 263-271.
[42] Sarkar, B., Cárdenas-Barrón, L. E., Sarkar, M., & Singgih, M. L. (2014). An economic production quantity model with random defective rate, rework process and backorders for a single stage production system. Journal of Manufacturing Systems, 33(3), 423-435.
[43] Sarkar, B., Chaudhuri, K., & Moon, I. (2015). Manufacturing setup cost reduction and quality improvement for the distribution free continuous-review inventory model with a service level constraint. Journal of Manufacturing Systems, 34, 74-82.
[44] Sarkar, B., Sett, B. K., & Sarkar, S. (2018). Optimal production run time and inspection errors in an imperfect production system with warranty. Journal of Industrial & Management Optimization, 14(1), 267-282.
[45] Taheri-Tolgari, J., Mirzazadeh, A., & Jolai, F. (2012). An inventory model for imperfect items under inflationary conditions with considering inspection errors. Computers & Mathematics with Applications, 63(6), 1007-1019.
[46] Taleizadeh, A. A., Kalantari, S. S., & Cárdenas-Barrón, L. E. (2015). Determining optimal price, replenishment lot size and number of shipments for an EPQ model with rework and multiple shipments. Journal of Industrial and Management Optimization, 11(4), 1059-1071.
[47] Taleizadeh, A. A., Niaki, S. T. A., & Najafi, A. A. (2010). Multiproduct single-machine production system with stochastic scrapped production rate, partial backordering and service level constraint. Journal of Computational and Applied Mathematics, 233(8), 1834-1849.
[48] Taleizadeh, A. A., Sadjadi, S. J., & Niaki, S. T. A. (2011). Multiproduct EPQ model with single machine, backordering and immediate rework process. European Journal of Industrial Engineering, 5(4), 388-411.
[49] Taleizadeh, A. A., Sarkar, B., & Hasani, M. (2019a). Delayed payment policy in multi-product single-machine economic production quantity model with repair failure and partial backordering. Journal of Industrial & Management Optimization, 13(5), 1.
[50] Taleizadeh, A. A., Wee, H. M., & Jalali-Naini, S. G. (2013). Economic production quantity model with repair failure and limited capacity. Applied Mathematical Modelling, 37(5), 2765-2774.
[51] Taleizadeh, A. A., Yadegari, M., & Sana, S. S. (2019b). Production models of multiple products using a single machine under quality screening and reworking policies. Journal of Modelling in Management.
[52] Taleizadeh A.A., Cárdenas-Barrón L.E., Biabani J., Nikousokhan, R., (2012). "Multi products single machine EPQ model with immediate rework process," International Journal of Industrial Engineering Computations, 3(2): 93-102.
[53] Taleizadeh A.A., Moghadasi H., Niaki S.T.A., Eftekhari A.K., (2009). "An EOQ-Joint Replenishment Policy to Supply Expensive Imported Raw Materials with Payment in Advance," Journal of Applied Science, 8(23): 4263-4273.
[54] Taleizadeh A.A., Najafi A.A., Niaki S.T.A., (2010). "Multi Product EPQ Model with Scraped Items and limited Production Capacity," International Journal of Science and Technology (Scientia Iranica) Transaction E, 17 (1): 58-69.
[55] Taleizadeh A.A., Kalantary S.S., Cárdenas-Barrón L.E., (2016). Pricing and Lot sizing for an EPQ Inventory Model with Rework and Multiple Shipments, TOP, 24:143-155.
[56] Tayyab, M., & Sarkar, B. (2016). Optimal batch quantity in a cleaner multi-stage lean production system with random defective rate. Journal of cleaner production, 139, 922-934.
[57] Wee, H. M., & Widyadana, G. A. (2012). Economic production quantity models for deteriorating items with rework and stochastic preventive maintenance time. International Journal of Production Research, 50(11), 2940-2952.

[58] Wee, H. M., Wang, W. T., & Cárdenas-Barrón, L. E. (2013). An alternative analysis and solution procedure for the EPQ model with rework process at a single-stage manufacturing system with planned backorders. Computers & Industrial Engineering, 64(2), 748-755.
[59] Yang, X. S. (2010). Nature-inspired metaheuristic algorithms. Luniver press.
[60] Yeniay, Ö. (2005). Penalty function methods for constrained optimization with genetic algorithms. Mathematical and computational Applications, 10(1), 45-56.
[61] Yoo, S. H., Kim, D., & Park, M. S. (2009). Economic production quantity model with imperfect-quality items, two-way imperfect inspection and sales return. International Journal of Production Economics, 121(1), 255-265.
[62] Yoo, S. H., Kim, D., & Park, M. S. (2012). Inventory models for imperfect production and inspection processes with various inspection options under one-time and continuous improvement investment. Computers & Operations Research, 39(9), 2001-2015.