Optimization of a coordinated sustainable multi-vendor multi-livestock multi-rancher supply chain for growing products

Ali Fallahi[1], Yeganeh Tehrani [2], and Hadi Mokhtari *[2]

[1] *Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran*

[2] *Department of Industrial Engineering, Faculty of Engineering, University of Kashan, Kashan, Iran*

Abstract.
Inventory management for growing items is crucial in many industries, including agriculture, aquaculture, and animal husbandry. This paper develops a new mathematical model for the inventory management of growing products in a multi-vendor, multi-livestock, multi-rancher supply chain. The possibility of partial backorder shortages is considered, and both backorder and lost sale shortages are possible. To address environmental concerns, the carbon emissions of the system are limited by a direct cap policy. The main objective is to determine the optimal ordering and shortage quantity for each livestock type for each rancher. We incorporate the Hill coordination strategy into our proposed model to provide a centralized decision-making framework. Given the nonlinearity and dimensionality of the model, we propose metaheuristic algorithms as the solution approach. To this end, genetic algorithms, differential evolution, and particle swarm optimization algorithms are designed and implemented for the problem. The input parameters of all algorithms are tuned using Taguchi's design of experiments. We evaluate the performance of these algorithms by solving several numerical instances in small, medium, and large size categories. The experimental results show that the genetic algorithm outperforms the other metaheuristics regarding the quality of solutions. Finally, some suggestions for extending the current study are discussed.

Keywords: Inventory management, livestock, growth, direct cap, metaheuristic algorithms

***Correspond author.** (H. Mokhtari) - Department of Industrial Engineering, Faculty of Engineering, University of Kashan, 6 Km, Ghotbravandi Blvd, Kashan, 87317-53153, Iran

Contact Information

Ali Fallahi: Email address: ali.fallahi10@gmail.com – Mobile phone number: +989130937168

Yeganeh Tehrani: Email address: yeganeh_tehrani@yahoo.com – Mobile phone number: +989034098382

Hadi Mokhtari: Email address: mokhtari_ie@kashanu.ac.ir – Mobile phone number: +989124359296

1. Introduction

Effective inventory management in supply chains is critical for ensuring the efficient delivery of goods and services to customers. Different coordination mechanisms in supply chains have been developed in recent years to achieve more effective inventory management. The developed coordination mechanisms, such as buy-back, revenue sharing, option, consignment stock, and vendor-managed inventory (VMI), are actively used to improve the performance of inventory systems in the supply chain environment [1-5]. One of the most challenging inventory systems is the inventory system of growing products. For the first time, Rezaei [6] addressed the inventory system of growing products, and developed a new economic order quantity model to determine the replenishment policy in a single-echelon system. After this study, the subsequent research tried to bring this basic model to a real-world environment as much as possible. They assumed several realistic assumptions, such as the presence of multi-item [7], quality control aspects [8, 9], discount policy [10], product deterioration [11], trade credits [12, 13], shortage [14], mortality [15], transportation decisions [16], substitution [17], carbon emission [18, 19], etc. However, the coordination of entities in the supply chain of growing products remained unaddressed until Malekitabar, et al. [20] presented a two-echelon supply chain model for the problem. They applied revenue and cost-sharing contracts for coordination in the supply chain. Afterward, more articles focused on this problem. For example, Pourmohammad-Zia, et al. [21] developed an integrated single-vendor single-buyer supply chain model with pricing and inventory decisions. Their model also considered the possibility of deterioration. As one of the few works on three echelon supply chains, Sebatjane and Adetunji [22] studied a farmer-food processor-retailer network with imperfect quality growing products. They suggested a salvage policy to deal with low-quality items. More details on the literature on inventory planning for growing products are also available for interested readers in the review article by Pourmohammad-Zia [23].

On the other hand, there has been a growing interest in developing sustainable supply chain models that consider the environmental impacts of supply chain activities [24-26]. With increasing concerns over climate change and the impact of greenhouse gas emissions on the environment, organizations are under pressure to reduce their carbon footprint. As a result, incorporating carbon emission considerations into supply chain decision-making has become increasingly important. The importance of carbon emission is also so high for the inventory system of growing products, so several recent studies focused on determining optimal replenishment decisions by considering these concerns [27]. For example, Gharaei and Almehdawe [18] derived the optimal ordering policy for growing products under carbon tax regulation. De-la-Cruz-Márquez, et al. [28] also utilized carbon tax regulation to develop a sustainable model for a growing product inventory system with price-sensitive demand. Rana, et al. [29] developed a new model for inventory planning of growing products under payment delays as a financial option. The carbon emission was taken into account by direct accounting regulation. In another study, Zhang, et al. [30] formulated the model by Rezaei [6] under the carbon tax, cap-and-trade, and cap-and-offset regulations.

From the solution methodology viewpoint, researchers have tried to address similar problems using efficient algorithms in recent years. For example, Pasandideh, et al. [31] studied a single-vendor single-buyer supply chain with multiple products, and proposed genetic algorithm (GA) as the solution approach. The integrated model of the problem was derived using VMI contract. Sadeghi, et al. [32] investigated an inventory management problem for a two-layer supply chain with single-vendor and multiple buyers. The demand for considered as a fuzzy parameter in this work. They presented a particle swarm optimization (PSO) algorithm to solve the model. Sadeghi, et al. [33] considered the presence of multiple vendors and multiple buyers in a two-echelon supply chain, and solved the developed model by GA and PSO algorithms. In another work, Sadeghi, et al. [34] extended their previous work by developing a bi-objective model that simultaneously minimizes the total cost and maximizes the reliability for a single-vendor multiple-buyers system. They utilized the non-dominated sorting genetic algorithm-ii (NSGA-II) as the multiobjective version of GA for solving the model. Mousavi, et al. [35] explored a multi-item inventory model with operational constraints and dynamic demand. The GA and SA were the suggested solution approaches in this research. Nia, et al. [36] developed a hybrid genetic and imperialist competitive metaheuristic for the inventory management in a single-vendor single-buyer supply chain, and show the efficiency of the algorithm by making comparison to the GA in the literature. Keramati, et al. [37] proposed a simulation-metaheuristic approach by combining the SA algorithm with a simulation module to solve a multi-product inventory model. In this study, the decisions on supplier selection and inventory classification were also considered. Fallahi, et al. [38] designed differential evolution (DE) and PSO metaheuristics for a multiproduct EOQ model of reusable items. The authors boosted the classical algorithms through the integration of Q-learning algorithm for parameter adaption. Finally, Sadeghi, et al. [39] employed the grey wolf optimizer and whale optimization algorithm as two novel metaheuristic algorithms for a two-layer supply chain of reusable products. Table 1 provides a better clarification of the research gap and novelties of our paper against the previous articles on inventory management of growing products in the literature.

[Please insert Table 1 about here]

To the best of our knowledge, none of the previous works studies a multi-livestock inventory management problem for a multi-vendor multi-buyer supply chain environment under the coordination and carbon emission concerns. As one of the closest works to our paper, Pourmohammad-Zia, et al. [40] formulated the VMI contract for an integrated three-echelon supply chain of growing products. However, they assumed that there is a single entity in each supply chain echelon. In addition, their model determines a single product's replenishment policy, ignoring the carbon emission concerns. In this paper, we fill this research gap and present a novel mathematical model for integrated optimizing the inventory decisions of growing products in a multi-vendor multi-livestock multi-buyer supply chain. Our model considers the direct cap policy for carbon emission of buyers, ensuring that sustainability is integrated into the supply chain. In addition, we consider the possibility of partial backordering shortage, in which both types of the backorder and lost sale shortage are possible. The developed model is a complex large-size nonlinear model. The previous research successfully employed metaheuristic algorithms to solve such constrained inventory management problems in multi-echelon supply chains. Consequently, GA, DE, and PSO are designed and implemented for this goal as three of the well-known and efficient metaheuristics that have shown great potential in solving similar problems in the literature.

The rest of the paper is organized as follows. In Section 2, we define the new inventory management problem, and develop a nonlinear mathematical model. In Section 3, we propose and design GA and DE metaheuristic algorithms as the solution approach. In Section 4, we analyze the performance of algorithms by solving several numerical examples. In Section 5, some managerial insights are discussed based on the computed results. In Section 6, we present the conclusion of the paper, and our suggestions for future research.

2. Problem description and modeling

In this section, we will present the new problem for inventory management of growing products in a two-echelon supply chain, and model this problem mathematically.

2.1 Assumptions

The main assumption of the newly presented inventory model is as follows:

- There is a multi-vendor multi-livestock multi-rancher supply chain.
- The partial backorder shortage is allowed.
- The demand for each type of livestock from each buyer is constant and deterministic.
- The carbon emission is results from the placement of orders and holding of slaughtered live stocks.
- The direct cap approach limits the carbon emission.
- The decision-making is centralized for the supply chain.
- There is no possibility of a discount.

2.2 Notations

The following sets, parameters, and variables are considered in the formulation of the problem:

Sets

K	The set of vendors
J	The set of livestock
I	The set of ranchers

Parameters

AB_{ijk}	The ordering cost of rancher i for the ordering of livestock j from vendor k
AS_{ijk}	The ordering cost of vendor k for the ordering of livestock j by rancher i
h_{ij}	The holding cost of livestock j for rancher k
w_{ij}	The backorder shortage cost of livestock j for rancher i
τ_{ij}	The lost sale shortage cost of livestock j for rancher i
D_{ij}	The demand for livestock j of rancher i
β_{ij}	The portion of backorder shortage for product j of rancher i
p	The unit carbon emission per holding slaughtered livestock
q	The unit carbon emission per each order
T_e	The carbon emission quota for the total ordering carbon emission
T_r	The carbon emission quota for the total inventory holding carbon emission
Cf_j	The feeding cost of livestock j
A_j	The asymptotic weight of livestock j

L	The rate of growth
n	The parameter for the shape of the growth function
b	The constant integration of the growth function

Variables

W_t	The weight of livestock at time t
$f(t)$	The feeding function
$I_{ij}(t)$	The inventory level of livestock j in the warehouse of vendor i in time t
I_{ij}^{max}	The maximum inventory level of livestock j in the warehouse of vendor i
SC_B^k	The total inventory cost of vendor k
BC_B^i	The total inventory cost of rancher i
TC	The total cost of the supply chain under the coordination
Q_{ijk}	The economic quantity of order of livestock j from vendor k for rancher i
b_{ij}	The maximum shortage level of product j for rancher i
g	The growing period of products
T_{ij}	The overall inventory cycle of purchased livestock j from vendor k for rancher i
t_{ij}	The overall shortage cycle of livestock j for rancher i

2.3 Problem definition

As pointed out before, the first EOQ model for production-inventory management of growing products was developed by Rezaei [6]. In this study, we extend this basic model for a two-echelon supply chain network, including multiple vendors, multiple livestock, and multiple ranchers under coordination. At the beginning of each cycle, rancher i order Q_{ijk} newborn livestock j from the vendor k that can grow. The ordering costs of AB_{ijk} and AS_{ijk} are imposed on the rancher and vendor per each order, respectively. Each newborn livestock weighs w_0 at the time of receiving order. The live stocks are fed, and they grow to a target weight w_1 during a growth period g. To model and measure the growth pattern of products, a suitable growth function should be selected. We utilize the proposed growth function by Richards [41], which is one of the widely used models in the literature [6-8, 10]. The general form of the function is $W_t = A\left[1+be^{-Lt}\right]^{-1/n}$. Each rancher bear feeding costs during the growth period. The feeding cost of products fluctuates over, regarding the weight of livestock. Therefore, a feed intake function should be used to model the feeding behavior of animals. We consider a polynomial feed intake function, with a general form as $f(t) = b_0 + b_1 t + b_2 t^2 + b_3 t^3$ [7, 42]. Based on this feed intake function, the feeding cost of the growing livestock is calculated as $Cf_j Q_{ijk} \int_0^g b_0 + b_1 t + b_2 t^2 + b_3 t^3 \, dt$. After attaining the target weight, the livestock are slaughtered, and used to satisfy the demand of customers. The inventory holding cost h_{ij} results from the holding of slaughtered livestock. We consider the possibility of the shortage in the inventory system. In addition, we assume a partial backorder shortage to establish a more realistic decision support system. More specifically, β_{ij} portion of the shortage is backorder, and $(1-\beta_{ij})$ is lost sale. Furthermore, in recent years, environmental concerns and sustainability have become crucial factors in supply chain management. The carbon footprint of supply chains has been identified as a significant source of greenhouse gas emissions, contributing to climate change. As such, we take into account the carbon emission concerns in the proposed inventory system of growing products. In this study, we consider carbon emissions from both ordering and holding of livestock, which are denoted by T_e and T_r. To limit and reduce the carbon emissions, we will apply the direct cap policy, which sets a maximum allowable level of carbon emissions [43-45]. The goal of each rancher is to determine to main decisions as: (1) optimal ordering quantity of each newborn livestock and (2) optimal shortage quantity of each newborn livestock. Figure 1 shows the inventory level diagram for livestock in the warehouse of a rancher.

[Please insert Figure 1 about here]

2.4 Mathematical modeling

Based on the defined problem, the cost components of the inventory system can be formulated as follows [5, 32, 46]:

- The total ordering cost of livestock for each rancher:

$$\sum_{K=1}^{K}\sum_{j=1}^{J}\left[\frac{AB_{ijk} D_{ij}}{Q_{ijk} + (1-\beta_{ij}) b_{ij}}\right] \qquad \forall i \epsilon \mathrm{I} \tag{1}$$

- The total ordering cost of livestock for each vendor:

$$\sum_{k=1}^{k}\sum_{i=1}^{i}\sum_{j=1}^{j}\left[\frac{AS_{ijk}\,D_{ij}}{Q_{ijk}\left(1-\beta_{ij}\right)b_{ij}}\right] \qquad \forall k \in \mathrm{K} \tag{2}$$

- The total holding cost of slaughtered livestock for each rancher:

$$\sum_{K=1}^{K}\sum_{j=1}^{j}A_j\left[1+be^{-Lt}\right]^{-1/n}\left[\frac{h_{ij}\left[Q_{ijk}-\beta_{ij}b_{ij}\right]^2}{2\left(Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}\right)}\right] \qquad \forall i \in \mathrm{I} \tag{3}$$

- The total backorder shortage cost of slaughtered livestock for each rancher:

$$\sum_{K=1}^{K}\sum_{j=1}^{j}\left[\frac{w_{ij}\,\beta_{ij}\,b_{ij}^2}{2\left(Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}\right)}\right] \qquad \forall i \in \mathrm{I} \tag{4}$$

- The total lost sale shortage cost of slaughtered livestock for each rancher:

$$\sum_{K=1}^{K}\sum_{j=1}^{j}\left[\frac{\tau_{ij}\left(1-\beta_{ij}\right)b_{ij}\,D_{ij}}{Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}}\right] \qquad \forall i \in \mathrm{I} \tag{5}$$

- The total feeding cost of livestock before slaughtering for each rancher:

$$\sum_{K=1}^{K}\sum_{j=1}^{j}\left[Cf_j\,Q_{ijk}\int_0^g b_0+b_1t+b_2t^2+b_3t^3\,dt\right] \qquad \forall i \in \mathrm{I} \tag{6}$$

In addition, the shortage cycle and overall inventory cycles can be formulated as follows:

$$T_{ij}=\left(\sum_{k=1}^{K}Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}\right)/\,D_{ij} \tag{7}$$

$$t_{ij}=b_{ij}\,/\,D_{ij} \tag{8}$$

Based on the modeled cycles, the total cost of i^{th} rancher can be formulated as follows:

$$\begin{aligned}BC_B^i=&\sum_{K=1}^{K}\sum_{j=1}^{j}\left[\frac{AB_{ijk}\,D_{ij}}{Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}}\right]+\sum_{K=1}^{K}\sum_{j=1}^{j}\left[\frac{h_{ij}\left[Q_{ijk}-\beta_{ij}b_{ij}\right]^2}{2\left(Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}\right)}\right]\times A_j\left[1+be^{-Lt}\right]^{-1/n}\\&+\sum_{K=1}^{K}\sum_{j=1}^{j}\left[\frac{w_{ij}\,\beta_{ij}\,b_{ij}^2}{2\left(Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}\right)}\right]+\sum_{K=1}^{K}\sum_{j=1}^{j}\left[\frac{\tau_{ij}\left(1-\beta_{ij}\right)b_{ij}\,D_{ij}}{Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}}\right]+\sum_{K=1}^{K}\sum_{j=1}^{j}\left[Cf_j\,Q_{ijk}\int_0^g b_0+b_1t+b_2t^2+b_3t^3\,dt\right]\end{aligned} \qquad \forall i \in \mathrm{I} \tag{9}$$

Also, the vendors bear the ordering cost, which is calculated as follows:

$$SC_B^k = \sum_{i=1}^{i}\sum_{j=1}^{j}\left[\frac{AS_{ijk}\,D_{ij}}{Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}}\right] \qquad \forall k \in \mathrm{K} \tag{10}$$

As mentioned before, the goal of the present problem is to provide a coordinated decision-making framework. To this end, we use the proposed mechanism by Hill [47]. Under this coordination scheme, the vendors and ranchers are not responsible for their inventory decisions and costs. In this situation, the inventory decisions are made by a central unit in such a way that the overall cost of the supply chain is minimized. Therefore, the integrated objective function of the system can be formulated as follows:

$$TC = \sum_{k=1}^{K}\sum_{i=1}^{I}\left(\begin{array}{l}\sum_{j=1}^{J}\left[\frac{AS_{ijk}\,D_{ij}}{Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}}\right]+\sum_{j=1}^{J}\left[\frac{AB_{ijk}\,D_{ij}}{Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}}\right]+\left[\sum_{j=1}^{J}\left[\frac{h_{ij}\left[Q_{ijk}-\beta_{ij}b_{ij}\right]^2}{2\left(Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}\right)}\right]\times A_j\left[1+be^{-Lt}\ \right]^{-1/n}\right] \\ +\sum_{j=1}^{J}\left[\frac{w_{ij}\,\beta_{ij}\,b_{ij}^2}{2\left(Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}\right)}\right]+\sum_{j=1}^{J}\left[\frac{\tau_{ij}\left(1-\beta_{ij}\right)b_{ij}\,D_{ij}}{Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}}\right]+\sum_{j=1}^{J}\left[Cf_j\,Q_{ijk}\int_0^g b_0+b_1t+b_2t^2+b_3t^3\,dt\right]\end{array}\right) \tag{11}$$

The above objective function is subjected to the following constraints:

$$\sum_{j=1}^{J}\left[\frac{q\,D_{ij}}{Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}}\right] \le T_e \qquad \forall i \in I, k \in K \tag{12}$$

$$\sum_{j=1}^{J}\left[\frac{p\left[Q_{ijk}-\beta_{ij}b_{ij}\right]^2}{2\left(Q_{ijk}+\left(1-\beta_{ij}\right)b_{ij}\right)}\right] \le T_r \qquad \forall i \in \mathrm{I}, \forall k \in K \tag{13}$$

$$Q_{ijk}, b_{ij} \ge 0 \qquad \forall i \in I,\ j \in J, \forall k \in K \tag{14}$$

Constraints (12) specify the upper limit on the total carbon emission from ordering newborn livestock. Also, Constraints (13) show the limited quota for the carbon emission from the holding of slaughtered livestock. Note that we adopted the approach by Gharaei and Almehdawe [18] for the separate formulation of ordering and holding emission constraints. Finally, the types of decision variables are expressed via constraints (14).

3. Solution approach

The developed model for the proposed inventory system, as a generalization of the presented model by Khalilpourazari and Pasandideh [7], is a large-dimension constrained nonlinear programming mathematical model. Therefore, the classical exact algorithms or commercial solvers are not efficient in solving these problems. In other words, the challenges in employing classical exact algorithms or commercial solvers for solving these problems stem from the nonlinear nature of the objective function and constraints. The nonlinearity of the equations governing our model introduces a multitude of local optimum points within the solution space. These local optima can mislead classical algorithms into converging prematurely to suboptimal solutions. As a result, achieving the global optimum becomes a challenging task [7]. Also, the sustainability constraint embedded in our model creates a complex interplay between decision variables, making it difficult for classical algorithms to efficiently explore the entire feasible region in nonlinear space. Metaheuristics are among the widely used approaches that are used for these problems in the literature. These algorithms showed great power in solving the models for multi-product inventory systems. To solve the presented new mode, we design GA, DE, and PSO metaheuristics as three of the most powerful evolutionary algorithms in the literature. These algorithms are among the well-established and widely used metaheuristic algorithms, particularly in solving complex continuous optimization problems. They are known for their simplicity and ease of understanding, making them accessible to researchers and practitioners. This simplicity facilitates straightforward implementation and experimentation. These algorithms also have a reasonable number of input parameters, and this flexibility allows us to fine-tune their configurations for our specific problem by the statistical methods.

3.1 Genetic algorithm

GA was introduced by Holland [48] as one of the earliest metaheuristic algorithms for the first time in 1960. The significant power of GA in solving optimization problems in different fields made the algorithm one of the most popular evolutionary search techniques.

Stage 1: GA starts the search process with random initialization of N_{pop} population of solutions. Each individual in the population is known as a chromosome and includes a set of genes that represents the problem variables. GA evolves the generated initial solutions using two main operators, crossover and mutation, which are explained in the next stages. The operators are continually applied unit a termination criterion is met for the algorithm.

Stage 2: The crossover operator is the main responsible for generating solutions and evolving chromosomes in GA. This stage includes performing the crossover operators to generate new individuals. Different crossover operators are used for GA. In this research, we utilize the double-point crossover operator. This operator is commonly used in GAs to combine genetic material from two parent individuals to create new offspring individuals. The double-point crossover operator selects two random points on the parent chromosomes and exchanges the genetic material between the selected points to produce the corresponding offspring chromosomes. The two points are randomly selected to ensure diversity in the offspring population. The number of generated chromosomes by crossover is determined by a crossover rate parameter P_c.

Stage 3: In this stage, the algorithm applies a mutation operator to improve the diversity of solutions. We utilize the swap mutation operator for this goal. This operator randomly selects two positions in an individual's genotype and swaps the values at these positions. This results in the creation of a new solution that is similar to the original but has some differences in the order of the elements. The probability of applying the swap mutation can be controlled by a mutation rate parameter P_m. A higher mutation rate will result in more mutations and more solution space exploration but may also lead to premature convergence or loss of good solutions.

Stage 4: In the last stage, which is the selection stage, the newly generated chromosomes resulting from the crossover and mutation operations are combined with the previous population. Then, the fitness of each chromosome is evaluated using the fitness function. The best chromosomes with higher fitness values are selected for the next iteration. This process of selection is often referred to as "survival of the fittest," as only the chromosomes with the best fitness values are allowed to pass on their genetic material to the next generation.

3.2 Differential evolution algorithm

DE, as a population-based evolutionary algorithm, was developed by Storn and Price [49] for the first time. This algorithm has been extensively used in solving real-world optimization problems in several fields, such as medical decision-making, inventory management, scheduling, etc. [50, 51]. The DE algorithm comprises four main stages as below:

Stage 1: The DE algorithm generates a random initial population with size N_{pop} to start the search process. Consider iteration $k=1$ of DE algorithm to solve a D dimensional minimization problem. Then, the initial population can be expressed as follows:

$$\phi = \left\{ z_{1,1}, z_{1,2}, \ldots, z_{1,N_{pop}} \right\} \tag{15}$$

The DE algorithm targets all individuals in population ϕ for replacement in each iteration. The DE algorithm utilizes mutation, crossover, and acceptance operators for this goal. These operators are continuously applied until the algorithm reaches a stopping criterion, such as the maximum number of iterations.

Stage 2: Consider the k^{th} iteration of the DE algorithm. DE applies the mutation operator for each individual $z_{i,k}$ as follows:

$$z_{i,k} = z_{\alpha,k} + F\left(z_{\beta,k} - z_{\gamma,k}\right) \tag{16}$$

Where $\alpha, \beta, \gamma \epsilon \left\{1, \ldots, N_{pop}\right\}$ are three randomly selected different individuals from the population. In addition, $F \epsilon \left[0, 2\right]$ is the mutation scale factor, an input parameter of the DE algorithm.

Stage 3: In this stage, a crossover operator is applied to compute a trial vector $x_{i,k}$ through the combination of the target vector $z_{i,k}$ and mutant vector $z_{i,k}$ as follows:

$$x_{i,k}^{j} = \begin{cases} \hat{z}_{i,k}^{j} & if\ R_j \le P_c\ or\ j = I_j \\ z_{i,k}^{j} & o.w \end{cases} \tag{17}$$

where j denotes the j^{th} variable of the problem. Moreover, P_c is the crossover probability, and R_j is a random number in $\left[0,1\right]$ interval. Also, $I_j\left[1, D\right]$ is a random integer number that ensures a minimum difference between the target vector and trial vector.

Stage 4: After the generation of the corresponding trial vector of each individual, the acceptance operator is applied. In this stage, the objective functions of the trial vector and target vector are compared to choose the new individual for the next iteration. The comparison of trial and target vectors are as follows:

$$z_{i,k+1} = \begin{cases} x_{i,k} & if\ f\left(x_{i,k}\right) \le f\left(z_{i,k}\right) \\ z_{i,k} & o.w \end{cases} \tag{18}$$

3.3 Particle swarm optimization algorithm

Kennedy and Eberhart [52] developed PSO as a nature-inspired metaheuristic algorithm. The PSO algorithm drew inspiration from the collective interactions observed in groups of birds or fish. In PSO, the algorithm assigns the term particle to each individual member within the group, and the entire assembly of particles is referred to as a swarm. The algorithm has shown great potential in solving complex real-world optimization problems during the past two decades, and the researchers successfully applied the algorithm to problems in various areas such as production scheduling, medical decision-making etc. [53, 54]. There are four main stages in the implementation of PSO for an optimization problem:

Stage 1: In the PSO algorithm, each particle is recognized by its position and velocity. Consider $x_{i,j}^{k}$ and $v_{i,j}^{k}$ as the position and velocity vector corresponding to the variable j in the iteration k of the algorithm. Similar to GA and DE, the PSO also starts the search process by random generation of an initial population with a predetermined size N_{pop}. The algorithm assigns the randomly generated values to the position of particles. Also, the initial velocity for each particle is set to zero. The PSO algorithm focuses on all particles in the population for replacement in each iteration. More specifically, the algorithm uses two main operators, velocity update and position update, for this goal. This will be done until a termination criterion, like the maximum number of iterations, is reached.

Stage 2: Consider the k^{th} iteration of the PSO algorithm. PSO updates the velocity of the variable j for particle i using the following equation:

$$v_{i,j}^{k} = v_{i,j}^{k-1} + c_1 r_1 \left(G_{best} - x_{i,j}^{k-1}\right) + c_2 r_2 \left(P_{best} - x_{i,j}^{k-1}\right) \tag{19}$$

where r_1 and r_2 are two random numbers that are generated from $[0,1]$ interval. Here, P_{best} and G_{best} are the personal best position of particle i and the global best position for all particles, respectively. Also, c_1 and c_2 are two weight factors that determine the level of impact for the social and individual cognition.

Stage 3: In this stage, the algorithm uses the updated velocity and previous position information to update the position for each particle based on the below equation:

$$x_{i,j}^{k} = x_{i,j}^{k-1} + v_{i,j}^{k} \tag{20}$$

The implementation of the explained metaheuristic algorithms requires a solution representation scheme. To this end, four matrixes are used, which include the order quantity, shortage quantity, overall inventory cycle, and overall shortage cycle. Finally, the flowcharts of the proposed GA, DE, and PSO metaheuristics are depicted in Figure 2.

[Please insert Figure 2 about here]

4. Computational results

In this section, we analyze the performance of proposed metaheuristic algorithms to solve the nonlinear mathematical model of the problem. All the algorithms are coded and run on a personal computer with Intel(R) Core (TM) i7-6700HQ CPU @ 2.60GHz processor and 8 GB RAM. The input value of problem parameters is randomly generated from the specified ranges in Table 2. The considered ranges are mostly adapted from the papers by Rezaei [6] and Mokhtari and Rezvan [5].

[Please insert Table 2 about here]

As the performance of metaheuristics is highly affected by the value of their input parameters, a suitable approach should be used to determine parameters of GA, DE and PSO algorithms. A wide variety of approaches has been used for this goal in literature. Fractional factorial design, response surface methodology, and Taguchi design of experiments are some examples of the techniques used to calibrate metaheuristics [55, 56]. Taguchi's design of experiments is among the widely used methods in recent years. This method divides the algorithm's affecting factors into signal and noise factors. Then, Taguchi tries to determine the optimal level of signal factors in such a way that the response's performance is optimized. Taguchi uses the signal-to-noise ratio for this goal, which is as follows for a minimization problem:

$$\frac{S}{N} = -10\log\left(\frac{\sum_{j=1}^{k} y_j^2}{k}\right) \tag{21}$$

where y_j is the value of the objective function in replication j, and k is the total number of replications. One of the main benefits of Taguchi is the use of orthogonal arrays instead of full factorial design. Therefore, the optimal parameter levels of algorithms can be determined by expanding a minimum level of computational cost. We consider three levels for parameters of GA, DE, and PSO with the details presented in Table 3. The selection of input levels is performed using a trial-and-error procedure [33]. This methodology is firmly rooted in Taguchi's philosophy, emphasizing the importance of integrating quality design principles right from the inception of production, rather than addressing them later in the process.

[Please insert Table 3 about here]

For GA and DE algorithms, L^9 orthogonal arrays are used to determine the experiments. Also, L^{27} are employed to specify PSO's experiments. Each experiment is run in five replications. The obtained main effect plots from the signal-to-noise ratios are shown in Figure 3. In this figure, the level with the highest means of the signal-to-noise ratio is the optimal level for each parameter.

[Please insert Figure 3 about here]

After parameter tuning, the performance of algorithms should be compared. We consider four performance measures for the comparison. These measures include average objective function, CPU time, RPD, and RDI. The RPD and RDI are two error measures, which are computed as follows:

$$RDI = \frac{Met_{sol} - Best_{sol}}{Worst_{sol} - Best_{sol}} \tag{22}$$

$$RPD = \frac{Met_{sol} - Best_{sol}}{Best_{sol}} \tag{23}$$

where Met_{sol} is the obtained solution by the metaheuristic algorithm. Moreover, $Best_{sol}$ and $Worst_{sol}$ are the best and worst obtained solutions by all metaheuristics. The dimension of the three instance categories is presented in Table 4.

[Please insert Table 4 about here]

For each category, 10 examples are generated using the defined range in Table 2. The convergence curve of algorithms for a single instance in each category is shown in Figure 4. The detailed obtained results of metaheuristic algorithms for small, medium, and large categories are summarized in Tables 5 to 7. For the small instances, we also solved the problem using the CONOPT solver in GAMS.

[Please insert Figure 4 about here]

[Please insert Table 5 about here]

[Please insert Table 6 about here]

[Please insert Table 7 about here]

Based on the results, the gap of the best obtained solutions to the computed solution by the CONOP solver is relatively low. Moreover, the solver cannot find solutions for some instances. This is due to the nonlinearity of the equations in the model. As can be seen, GA has a better performance than DE and PSO in terms of the average objective function measure. This difference is more significant in small-sized instances, where the average objective function of DE, as the second-best algorithm, is about seven percent lower than GA. The difference in the quality of solutions is also reflected in average RDI and RPD measures. In all categories, GA reaches lower values of RDI and RPD. Despite this difference, DE needs less time to calculate the solutions. As can be seen, the CPU time of DE is less than GA and PSO for all examples. The CPU time difference is highlighted more in small and large-sized instances. Figures 5 and 6 demonstrate the comparison between the average objective function and CPU time of the algorithms for the solved problems in each size, respectively.

[Please insert Figure 5 about here]

[Please insert Figure 6 about here]

The boxplot of the average objective function and average CPU time measures are also presented in Figure 7 to provide better insights.

[Please insert Figure 7 about here]

In Figures 7(a) and 7(b), we present boxplots illustrating the distribution of average objective function and CPU time values for small instances, respectively. As can be seen, GA significantly outperforms DE and PSO regarding average objective function. For CPU time measure, DE exhibits the lowest median value, indicating its superior computation speed. However, it's worth noting that DE demonstrates a narrower spread of values for the average objective function, suggesting greater robustness. In medium instances, GA again achieves the lowest median objective function value, as shown in Figure 7(c). Moving on to computational efficiency, Figure 7(d) depicts boxplots representing the average CPU time for solving medium instances. Based on this plot, the DE algorithm demonstrates the shortest computation times, followed by GA and PSO. Note that the difference in CPU time of PSO with the other algorithms in medium instances is less than in small instances. In Figures 7(e) and X(f), we provide the boxplots for large instances. These boxplots reveal that GA maintains its competitive edge in terms of solution quality. As expected, DE remains the most efficient option for large instances in terms of CPU time.

5. Managerial insights and practical implications

The primary objective of our framework is to provide a decision support system tailored for the intricate realm of inventory management for growing products in supply chains. More specifically, this system caters to the unique demands of handling multiple growing products within two-level multi-vendor multi-rancher supply chains. Importantly, we have integrated the direct cap mechanism, a widely recognized emissions reduction strategy, into our framework, thereby enhancing its realism and applicability. For industrial managers, the consideration of emissions reduction policies, such as the direct cap, has emerged as an indispensable tool. These policies not only align with global sustainability imperatives but also serve as vital instruments for achieving sustainability within their industries. Our developed framework, in this context, emerges as a critical enabler, empowering managers to incorporate both economic and environmental dimensions of sustainability into their inventory management strategies.

The developed mathematical model for the problem belongs to the category of constrained nonlinear programming models. These models, characterized by their intricate nonlinear objective functions and constraints, pose formidable challenges for classical exact algorithms and commercial solvers. Recognizing this, we turn to the realm of metaheuristic algorithms, a domain that has demonstrated remarkable promise in tackling similar complex problems. In our endeavor to provide managers with a versatile solution methodology, we have designed Genetic Algorithms GA, DE, and PSO metaheuristics. These algorithms stand out for their capabilities in addressing continuous optimization problems. To showcase the practicality of our approach, we have undertaken a comprehensive series of experiments, spanning from small-scale instances to larger ones. The results obtained from these experiments present a clear recommendation for managers: GA emerges as the algorithm of choice.

This algorithm computes solutions with lower objective values across a diverse range of problem sizes. While it's important to note that GA excels in optimizing objectives, it does come with a marginally higher computational time, roughly 35% longer than DE in various examples. However, it's worth emphasizing that the computational time of all these algorithms remains reasonably low, making this difference, in practical terms, potentially negligible for managers.

6. Conclusion

This research developed a new model for sustainable, coordinated inventory planning of growing items in a multi-vendor multi-livestock multi-rancher supply chain. To establish the coordination between the vendors and ranchers, the Hill coordination mechanism was employed as one of the well-known and efficient strategies. In the proposed model, the partial backorder shortage of products was allowed for ranchers. In addition, carbon emission from ordering and inventory holding was taken into account via the direct cap policy. This new problem was formulated using a constrained nonlinear programming mathematical model. The goal of the proposed single-objective model was to determine the optimal ordering and shortage quantity of livestock for each rancher so that the total cost of the entire supply chain is minimized. The nonlinearity and dimension of the model prompted us to utilize the metaheuristic algorithms as the solution approach of the problem. To this end, GA, DE, and PSO metaheuristic algorithms were designed and implemented for the problem. To ensure the algorithms efficiency, the parameters were calibrated by the Taguchi method. Extensive analysis of results was provided by solving several numerical instances in small, medium, and large size categories. To this end, 10 numerical examples were randomly generated for each category, and solved by the algorithms. The results indicated that GA is more powerful than DE and PSO in terms of solution quality. More specifically, GA reaches solutions with a lower total cost in three considered instance categories. However, DE showed a better performance concerning the CPU time measure. Although DE computes the solutions in less time, the difference between its CPU time and GA's CPU time is insignificant. Finally, we presented some managerial insights based on the computed results.

The current study can be extended in several ways by future research. Here, we modeled the system considering a deterministic decision-making environment. Uncertainty modeling by using a proper approach such as robust optimization or chance-constrained programming is a direction to extend the current paper. Moreover, other state-of-art metaheuristic algorithms can be implemented for the problem, and their performance can be discussed with the developed GA and DE. Also, other limitations on the system's resources, such as the total available budget or available space, can be addressed in future research.

References

[1] Asadkhani, J., Fallahi, A., and Mokhtari, H., "A sustainable supply chain under VMI-CS agreement with withdrawal policies for imperfect items," *Journal of Cleaner Production,* **376**, pp. 134098, (2022), doi: doi.org/10.1016/j.jclepro.2022.134098.

[2] Haji, A., Afzalabadi, M., and Rasoul, R., "Pricing and inventory decisions in a vendor managed inventory system with revenue sharing contract," *Uncertain Supply Chain Management,* **6** (3), pp. 299-320, (2018), doi: doi.org/10.5267/j.uscm.2017.11.001.

[3] Hou, J., Zeng, A. Z., and Zhao, L., "Coordination with a backup supplier through buy-back contract under supply disruption," *Transportation Research Part E: Logistics and Transportation Review,* **46** (6), pp. 881-895, (2010), doi: doi.org/10.1016/j.tre.2010.03.004.

[4] Cai, J., Hu, X., Han, Y., et al., "Supply chain coordination with an option contract under vendor-managed inventory," *International Transactions in Operational Research,* **23** (6), pp. 1163-1183, (2016), doi: doi.org/10.1111/itor.12172.

[5] Mokhtari, H. and Rezvan, M. T., "A single-supplier, multi-buyer, multi-product VMI production-inventory system under partial backordering," *Operational Research,* **20** (1), pp. 37-57, (2020), doi: doi.org/10.1007/s12351-017-0311-z.

[6] Rezaei, J., "Economic order quantity for growing items," *International Journal of Production Economics,* **155**, pp. 109-113, (2014), doi: doi.org/10.1016/j.ijpe.2013.11.026.

[7] Khalilpourazari, S. and Pasandideh, S. H. R., "Modeling and optimization of multi-item multi-constrained EOQ model for growing items," *Knowledge-Based Systems,* **164**, pp. 150-162, (2019), doi: doi.org/10.1016/j.knosys.2018.10.032.

[8] Sebatjane, M. and Adetunji, O., "Economic order quantity model for growing items with imperfect quality," *Operations Research Perspectives,* **6**, pp. 100088, (2019), doi: doi.org/10.1016/j.orp.2018.11.004.

[9] Alfares, H. K. and Afzal, A. R., "An economic order quantity model for growing items with imperfect quality and shortages," *Arabian Journal for Science and Engineering,* **46**, pp. 1863-1875, (2021), doi: doi.org/10.1007/s13369-020-05131-z.

[10] Sebatjane, M. and Adetunji, O., "Economic order quantity model for growing items with incremental quantity discounts," *Journal of Industrial Engineering International,* **15**, pp. 545-556, (2019), doi: doi.org/10.1007/s40092-019-0311-0.
[11] Mokhtari, H., Salmasnia, A., and Asadkhani, J., "A new production-inventory planning model for joint growing and deteriorating items," *International Journal of Supply and Operations Management,* **7** (1), pp. 1-16, (2020), doi: doi.org/10.22034/ijsom.2020.1.1.
[12] Mahato, C., De, S. K., and Mahata, G. C., "Joint pricing and inventory management for growing items in a supply chain under trade credit," *Soft Computing,* **25** (11), pp. 7271-7295, (2021), doi: doi.org/10.1007/s00500-021-05635-2.
[13] Mittal, M. and Sharma, M., "Economic ordering policies for growing items (poultry) with trade-credit financing," *International Journal of Applied and Computational Mathematics,* **7**, pp. 1-11, (2021), doi: doi.org/10.1007/s40819-021-00973-z.
[14] Nobil, A. H., Sedigh, A. H. A., and Cárdenas-Barrón, L. E., "A generalized economic order quantity inventory model with shortage: case study of a poultry farmer," *Arabian Journal for Science and Engineering,* **44** (3), pp. 2653-2663, (2019), doi: doi.org/10.1007/s13369-018-3322-z.
[15] Pourmohammadzia, N., "Economic order quantity for growing items in the presence of mortality," *Journal of Supply Chain Management Science,* **3** (3-4), pp. 82-95, (2022), doi: doi.org/10.18757/jscms.2022.6843.
[16] Sebatjane, M. and Adetunji, O., "Optimal inventory replenishment and shipment policies in a three-echelon supply chain for growing items with expiration dates," *Opsearch,* **59** (3), pp. 809-838, (2022), doi: doi.org/10.1007/s12597-022-00572-6.
[17] Mokhtari, H., Salmasnia, A., and Fallahi, A., "Economic Production Quantity Under Possible Substitution: A Scenario Analysis Approach," *International Journal of Industrial Engineering & Production Research,* Research **33** (1), pp. 1-17, (2022), doi: doi.org/10.22068/ijiepr.33.1.2.
[18] Gharaei, A. and Almehdawe, E., "Optimal sustainable order quantities for growing items," *Journal of Cleaner Production,* **307**, pp. 127216, (2021), doi: doi.org/10.1016/j.jclepro.2021.127216.
[19] Fallahi, A., Azimi-Dastgerdi, M., and Mokhtari, H., "A sustainable production-inventory model joint with preventive maintenance and multiple shipments for imperfect quality items," *Scientia Iranica,* **30** (3), pp. 1204-1223, (2023), doi: doi.org/10.24200/sci.2021.55927.4475.
[20] Malekitabar, M., Yaghoubi, S., and Gholamian, M. R., "A novel mathematical inventory model for growing-mortal items (case study: Rainbow trout)," *Applied Mathematical Modelling,* **71**, pp. 96-117, (2019), doi: doi.org/10.1016/j.apm.2019.02.007.
[21] Pourmohammad-Zia, N., Karimi, B., and Rezaei, J., "Dynamic pricing and inventory control policies in a food supply chain of growing and deteriorating items," *Annals of Operations Research,* pp. 1-40, (2021), doi: doi.org/10.1007/s10479-021-04239-1.
[22] Sebatjane, M. and Adetunji, O., "Optimal inventory replenishment and shipment policies in a four-echelon supply chain for growing items with imperfect quality," *Production & Manufacturing Research,* **8** (1), pp. 130-157, (2020), doi: doi.org/10.1080/21693277.2020.1772148.
[23] Pourmohammad-Zia, N., "A review of the research developments on inventory management of growing items," *Journal of Supply Chain Management Science,* **2** (3-4), pp. 71-84, (2021), doi: doi.org/10.18757/jscms.2021.6122.
[24] Fallahi, A., Shahidi-Zadeh, B., and Niaki, S. T. A., "Unrelated parallel batch processing machine scheduling for production systems under carbon reduction policies: NSGA-II and MOGWO metaheuristics," *Soft Computing,* **27** (22), pp. 17063-17091, (2023), doi: doi.org/10.1007/s00500-023-08754-0.
[25] Amani Bani, E., Fallahi, A., Varmazyar, M., et al., "Designing a sustainable reverse supply chain network for COVID-19 vaccine waste under uncertainty," *Computers & Industrial Engineering,* **174**, pp. 108808, (2022), doi: doi.org/10.1016/j.cie.2022.108808.
[26] Kochakkashani, F., Kayvanfar, V., and Haji, A., "Supply chain planning of vaccine and pharmaceutical clusters under uncertainty: The case of COVID-19," *Socio-Economic Planning Sciences,* **87**, pp. 101602, (2023), doi: doi.org/10.1016/j.seps.2023.101602.
[27] Khan, M. A.-A., Cárdenas-Barrón, L. E., Treviño-Garza, G., et al., "Integrating prepayment installment, pricing and replenishment decisions for growing items with power demand pattern and non-linear holding cost under carbon regulations," *Computers & Operations Research,* **156**, pp. 106225, (2023), doi: doi.org/10.1016/j.cor.2023.106225.

[28] De-la-Cruz-Márquez, C. G., Cárdenas-Barrón, L. E., and Mandal, B., "An inventory model for growing items with imperfect quality when the demand is price sensitive under carbon emissions and shortages," *Mathematical Problems in Engineering,* **2021**, pp. 1-23, (2021), doi: doi.org/10.1155/2021/6649048.

[29] Rana, K., Singh, S. R., Saxena, N., et al., "Growing items inventory model for carbon emission under the permissible delay in payment with partially backlogging," *Green Finance,* **3** (2), pp. 153-174, (2021), doi: doi.org/10.3934/GF.2021009.

[30] Zhang, Y., Li, L.-y., Tian, X.-q., et al., "Inventory management research for growing items with carbon-constrained," in *2016 35th Chinese Control Conference (CCC)*, 2016: IEEE, pp. 9588-9593, doi: doi.org/10.1109/ChiCC.2016.7554880.

[31] Pasandideh, S. H. R., Niaki, S. T. A., and Nia, A. R., "A genetic algorithm for vendor managed inventory control system of multi-product multi-constraint economic order quantity model," *Expert Systems with Applications,* **38** (3), pp. 2708-2716, (2011), doi: doi.org/10.1016/j.eswa.2010.08.060.

[32] Sadeghi, J., Sadeghi, S., and Niaki, S. T. A., "Optimizing a hybrid vendor-managed inventory and transportation problem with fuzzy demand: an improved particle swarm optimization algorithm," *Information Sciences,* **272**, pp. 126-144, (2014), doi: doi.org/10.1016/j.ins.2014.02.075.

[33] Sadeghi, J., Mousavi, S. M., Niaki, S. T. A., et al., "Optimizing a multi-vendor multi-retailer vendor managed inventory problem: Two tuned meta-heuristic algorithms," *Knowledge-Based Systems,* **50**, pp. 159-170, (2013), doi: doi.org/10.1016/j.knosys.2013.06.006.

[34] Sadeghi, J., Sadeghi, S., and Niaki, S. T. A., "A hybrid vendor managed inventory and redundancy allocation optimization problem in supply chain management: An NSGA-II with tuned parameters," *Computers & Operations Research,* **41**, pp. 53-64, (2014), doi: doi.org/10.1016/j.cor.2013.07.024.

[35] Mousavi, S. M., Hajipour, V., Niaki, S. T. A., et al., "Optimizing multi-item multi-period inventory control system with discounted cash flow and inflation: two calibrated meta-heuristic algorithms," *Applied Mathematical Modelling,* **37** (4), pp. 2241-2256, (2013), doi: doi.org/10.1016/j.apm.2012.05.019.

[36] Nia, A. R., Far, M. H., and Niaki, S. T. A., "A hybrid genetic and imperialist competitive algorithm for green vendor managed inventory of multi-item multi-constraint EOQ model under shortage," *Applied Soft Computing,* **30**, pp. 353-364, (2015), doi: doi.org/10.1016/j.asoc.2015.02.004.

[37] Keramati, F., Mokhtari, H., and Fallahi, A., "Joint decisions on inventory classification, supplier selection and production policy for a multi-item EPQ inventory system under uncertainty," *Journal of Quality Engineering and Production Optimization,* **7** (2), pp. 186-204, (2022), doi: doi.org/10.22070/jqepo.2023.16975.1247.

[38] Fallahi, A., Amani Bani, E., and Niaki, S. T. A., "A constrained multi-item EOQ inventory model for reusable items: Reinforcement learning-based differential evolution and particle swarm optimization," *Expert Systems with Applications,* **207**, pp. 118018, (2022), doi: doi.org/10.1016/j.eswa.2022.118018.

[39] Sadeghi, A. H., Amani Bani, E., Fallahi, A., et al., "Grey Wolf Optimizer and Whale Optimization Algorithm for Stochastic Inventory Management of Reusable Products in a two-level Supply Chain," *IEEE Access,* **12**, (2023), doi: doi.org/10.1109/ACCESS.2023.3269292.

[40] Pourmohammad-Zia, N., Karimi, B., and Rezaei, J., "Food supply chain coordination for growing items: A trade-off between market coverage and cost-efficiency," *International Journal of Production Economics,* **242**, pp. 108289, (2021), doi: doi.org/10.1016/j.ijpe.2021.108289.

[41] Richards, F. J., "A flexible growth function for empirical use," *Journal of Experimental Botany,* **10** (2), pp. 290-301, (1959), doi: doi.org/10.1093/jxb/10.2.290.

[42] Goliomytis, M., Panopoulou, E., and Rogdakis, E., "Growth curves for body weight and major component parts, feed consumption, and mortality of male broiler chickens raised to maturity," *Poultry Science,* **82** (7), pp. 1061-1068, (2003), doi: doi.org/10.1093/ps/82.7.1061.

[43] Chen, X., Benjaafar, S., and Elomri, A., "The carbon-constrained EOQ," *Operations Research Letters,* **41** (2), pp. 172-179, (2013), doi: doi.org/10.1016/j.orl.2012.12.003.

[44] Sherafati, M., Bashiri, M., Tavakkoli-Moghaddam, R., et al., "Achieving sustainable development of supply chain by incorporating various carbon regulatory mechanisms," *Transportation Research Part D: Transport and Environment,* **81**, pp. 102253, (2020), doi: doi.org/10.1016/j.trd.2020.102253.

[45] Toptal, A., Özlü, H., and Konur, D., "Joint decisions on inventory replenishment and emission reduction investment under different emission regulations," *International Journal of Production Research,* **52** (1), pp. 243-269, (2014), doi: doi.org/10.1080/00207543.2013.836615.

[46] Taleizadeh, A. A., Niaki, S. T. A., and Barzinpour, F., "Multiple-buyer multiple-vendor multi-product multi-constraint supply chain problem with stochastic demand and variable lead-time: a harmony search algorithm," *Applied Mathematics and Computation,* **217** (22), pp. 9234-9253, (2011), doi: doi.org/10.1016/j.amc.2011.04.001.

[47] Hill, R. M., "The single-vendor single-buyer integrated production-inventory model with a generalised policy," *European Journal of Operational Research,* **97** (3), pp. 493-499, (1997), doi: doi.org/10.1016/S0377-2217(96)00267-6.

[48] Holland, J. H., *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT Press, 1992.

[49] Storn, R. and Price, K., "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization,* **11**, pp. 341-359, (1997), doi: doi.org/10.1023/A:1008202821328.

[50] Fallahi, A., Mahnam, M., and Niaki, S. T. A., "A discrete differential evolution with local search particle swarm optimization to direct angle and aperture optimization in IMRT treatment planning problem," *Applied Soft Computing,* **131**, pp. 109798, (2022), doi: doi.org/10.1016/j.asoc.2022.109798.

[51] Onwubolu, G. and Davendra, D., "Scheduling flow shops using differential evolution algorithm," *European Journal of Operational research,* **171** (2), pp. 674-692, (2006), doi: doi.org/10.1016/j.ejor.2004.08.043.

[52] Kennedy, J. and Eberhart, R., "Particle swarm optimization," in *Proceedings of ICNN'95-International Conference on Neural Networks*, 1995, vol. 4: ieee, pp. 1942-1948, doi: doi.org/10.1109/ICNN.1995.488968.

[53] Keshavarz, T., Salmasi, N., and Varmazyar, M., "Flowshop sequence-dependent group scheduling with minimisation of weighted earliness and tardiness," *European Journal of Industrial Engineering,* **13** (1), pp. 54-80, (2019), doi: doi.org/10.1504/EJIE.2019.097920.

[54] Fallahi, A., Mahnam, M., and Niaki, S. T. A., "Direct aperture optimization for intensity modulated radiation therapy: Two calibrated metaheuristics and liver cancer case study," *International Journal of Industrial Engineering and Production Research,* **33** (2), pp. 1-14, (2022), doi: doi.org/10.22068/ijiepr.33.2.4.

[55] Costa, C. B., Maciel, M. R. W., and Maciel Filho, R., "Factorial design technique applied to genetic algorithm parameters in a batch cooling crystallization optimisation," *Computers & Chemical Engineering,* **29** (10), pp. 2229-2241, (2005), doi: doi.org/10.1016/j.compchemeng.2005.08.005.

[56] Najafi, A. A., Niaki, S. T. A., and Shahsavar, M., "A parameter-tuned genetic algorithm for the resource investment problem with discounted cash flows and generalized precedence relations," *Computers & Operations Research,* **36** (11), pp. 2994-3001, (2009), doi: doi.org/10.1016/j.cor.2009.01.016.

Figures Captions

Figure 1: The inventory level diagram for livestock j in the warehouse of rancher i

Figure 2: The flowcharts of the proposed metaheuristic algorithms

Figure 3: The signal-to-noise ratio main effect plots

Figure 4: The convergence curve of metaheuristics in three categories of numerical instances

Figure 5: Average objective function of the metaheuristic algorithms in different sizes of numerical instances

Figure 6: Average CPU time of the metaheuristic algorithms in different sizes of numerical instances

Figure 7: The boxplot of the average objective function and CPU time measures

Tables Captions

Table 1: The novelties of current work against the previous research in the literature of inventory models for growing products

Table 2: The range of input parameters of numerical instances

Table 3: Three considered levels for the parameter calibration of metaheuristics

Table 4: The dimension of three categories of numerical instances

Table 5: The computational results of algorithms for small instances

Table 6: The computational results of algorithms for medium instances

Table 7: The computational results of algorithms for large instances

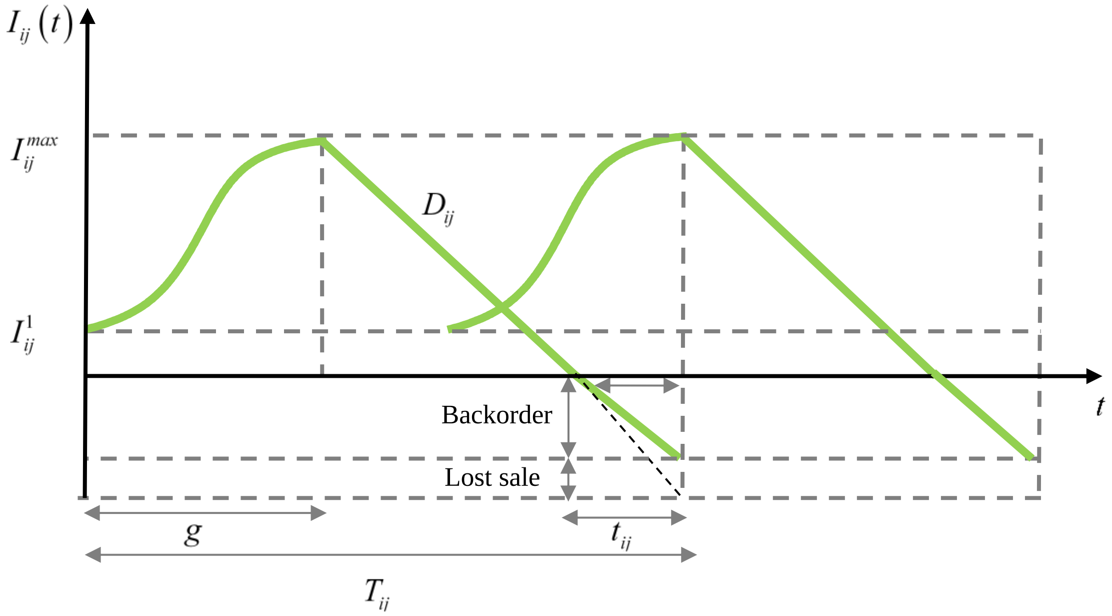


Figure 1: The inventory level diagram for livestock j in the warehouse of rancher i

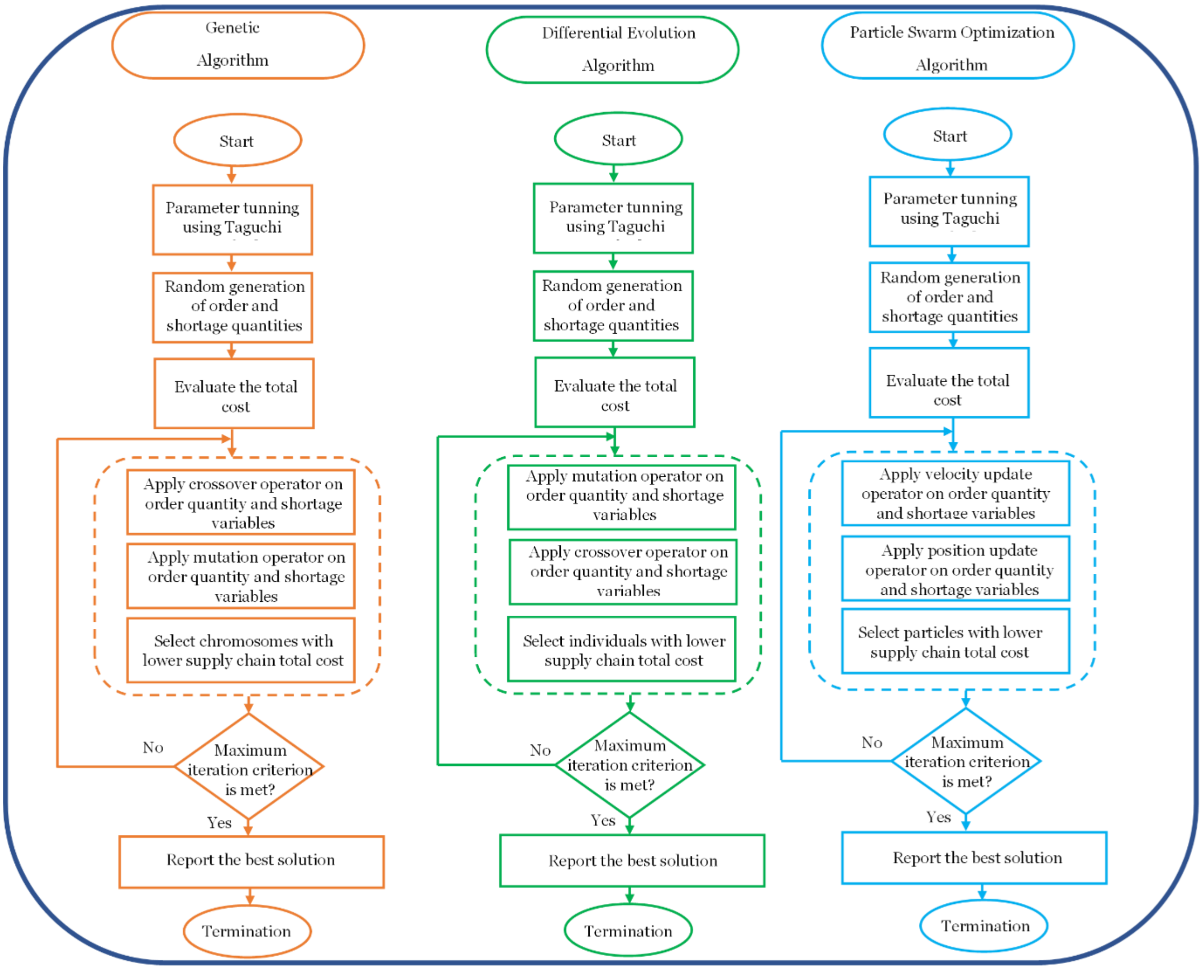
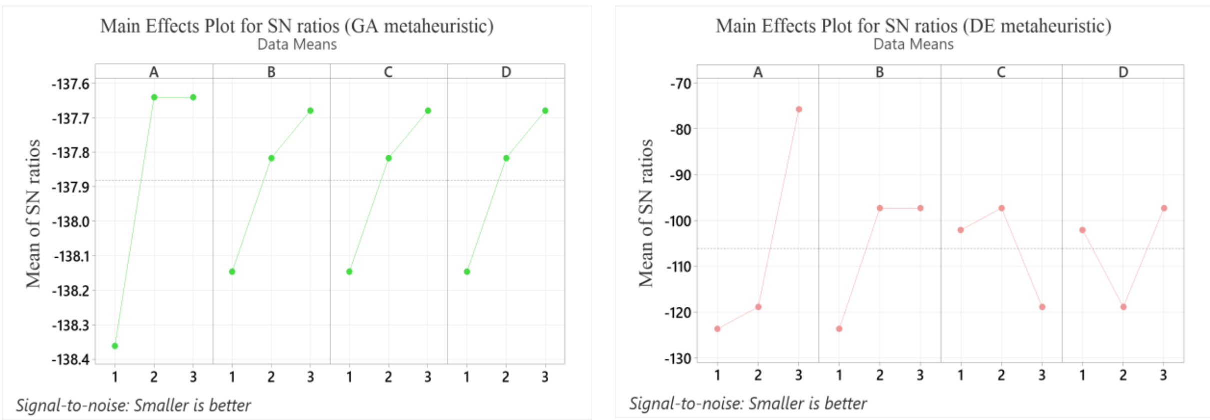


Figure 2: The flowcharts of the proposed metaheuristic algorithms



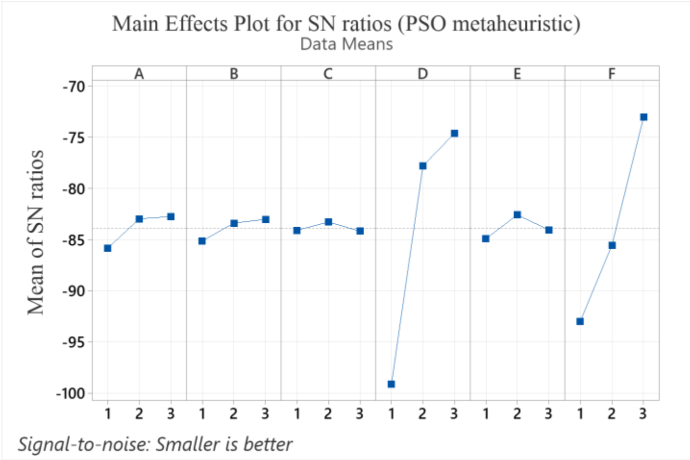
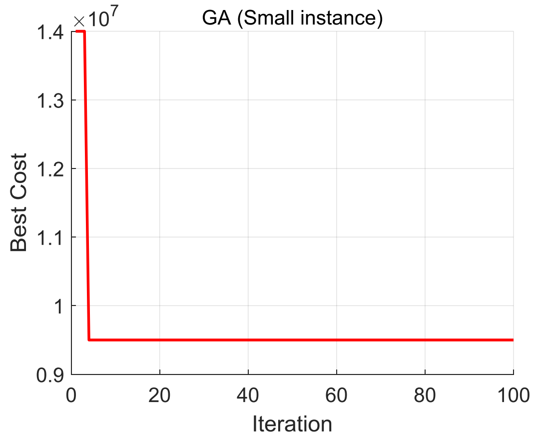
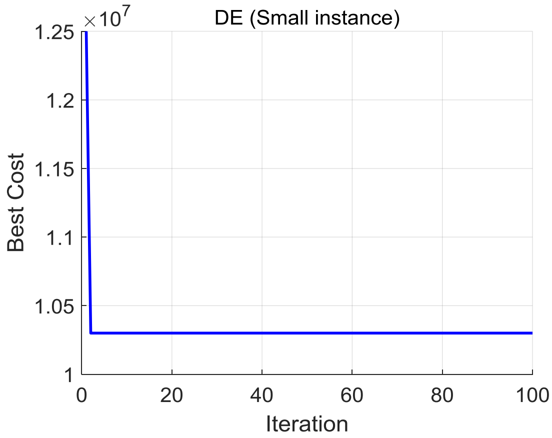


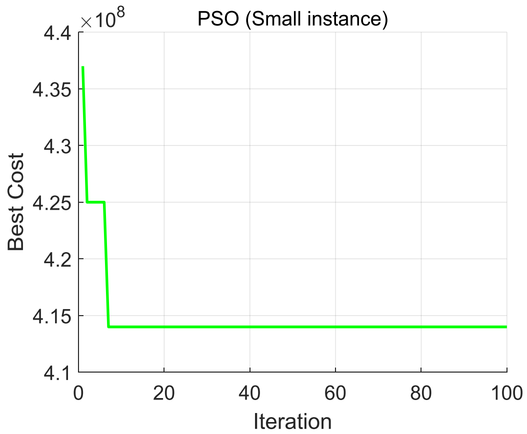
Figure 3: The signal-to-noise ratio main effect plots



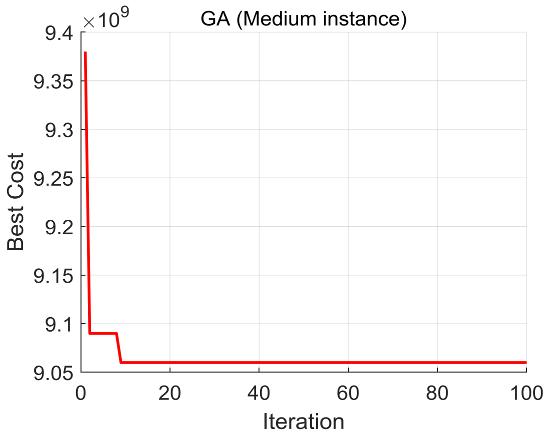
(a): GA-small instance



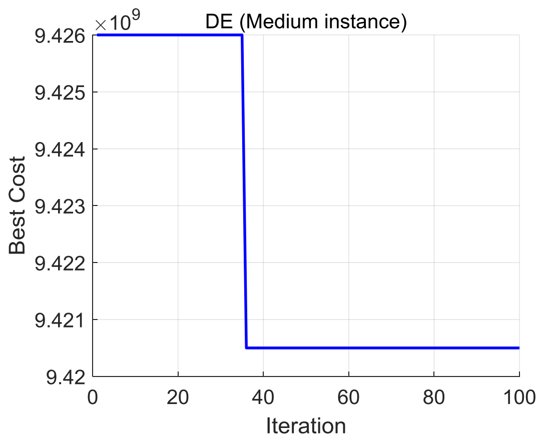
(b): DE-small instance



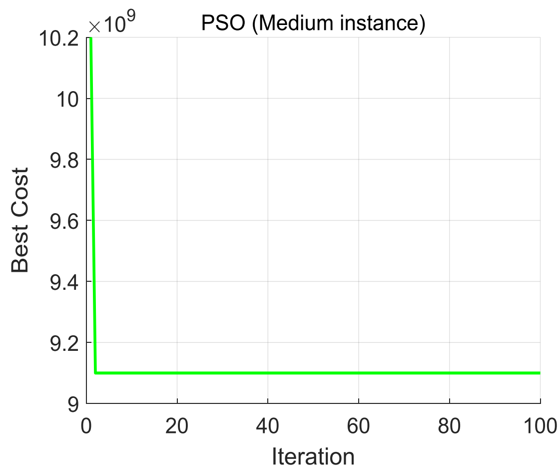
(c): PSO-small instance



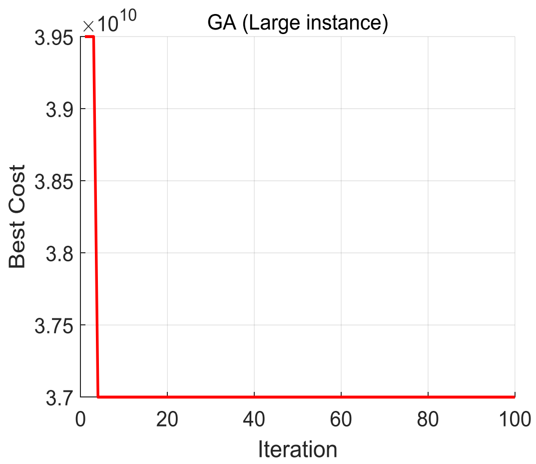
(d): GA-medium instance



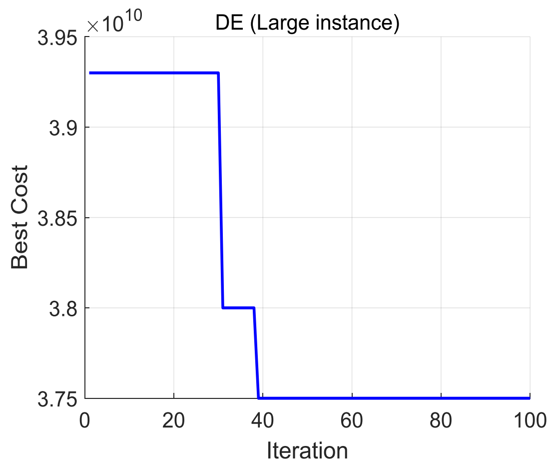
(e): DE-medium instance



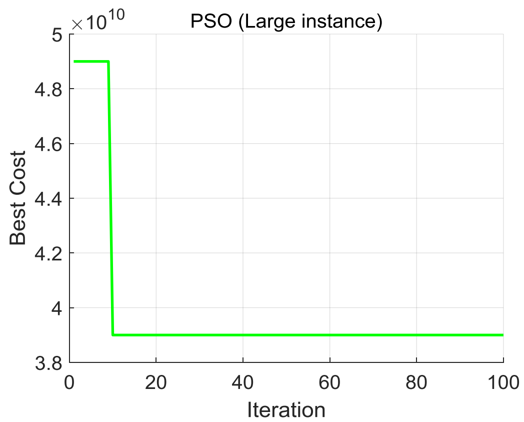
(f): PSO-medium instance



(g): GA-large instance



(h): DE-large instance



(i): PSO-large instance

Figure 4: The convergence curve of metaheuristics in three categories of numerical instances

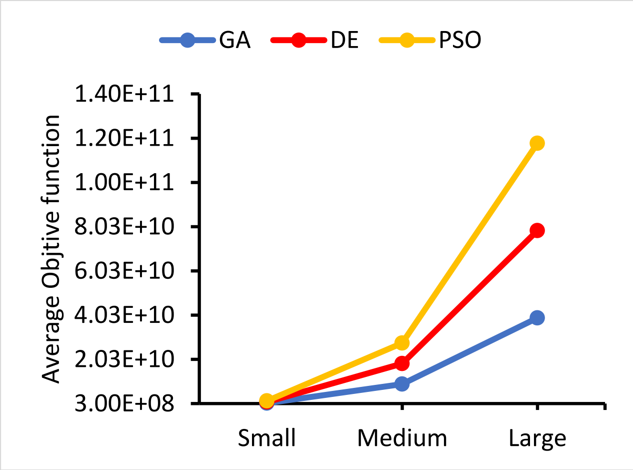


Figure 5: Average objective function of the metaheuristic algorithms in different sizes of numerical instances

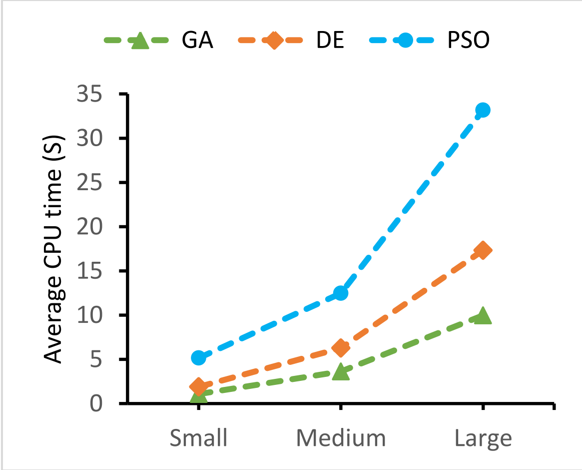
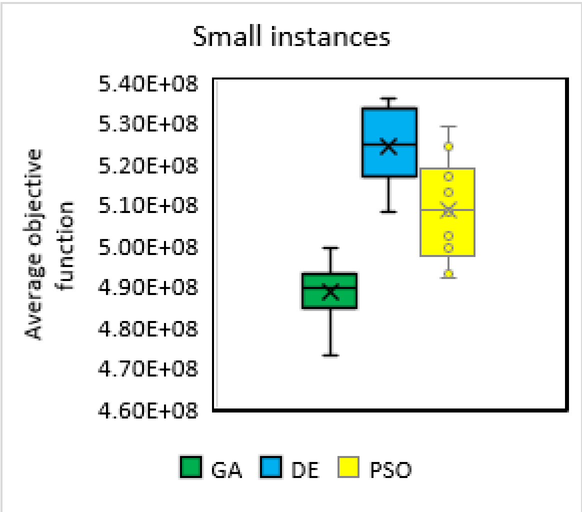
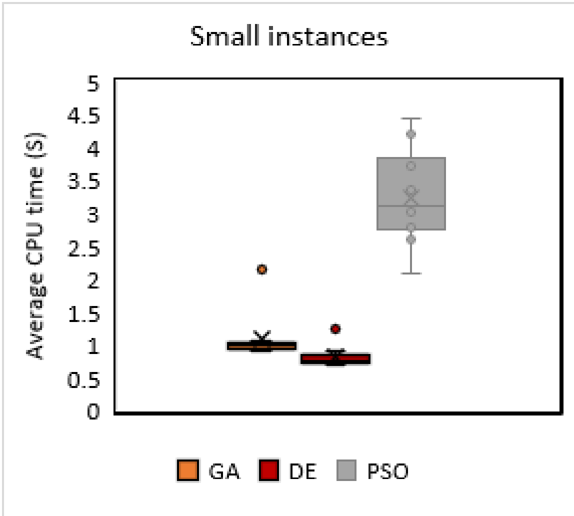


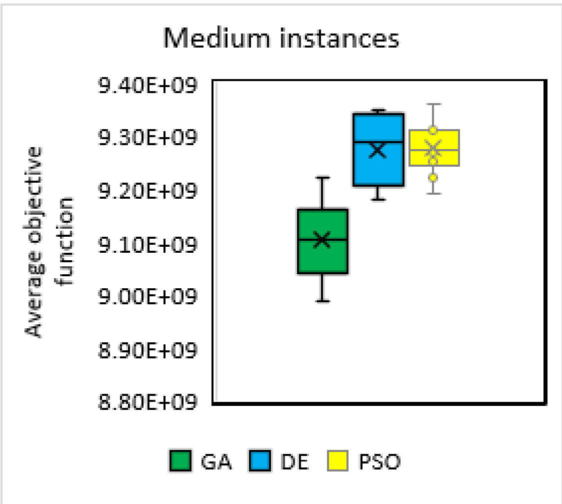
Figure 6: Average CPU time of the metaheuristic algorithms in different sizes of numerical instances



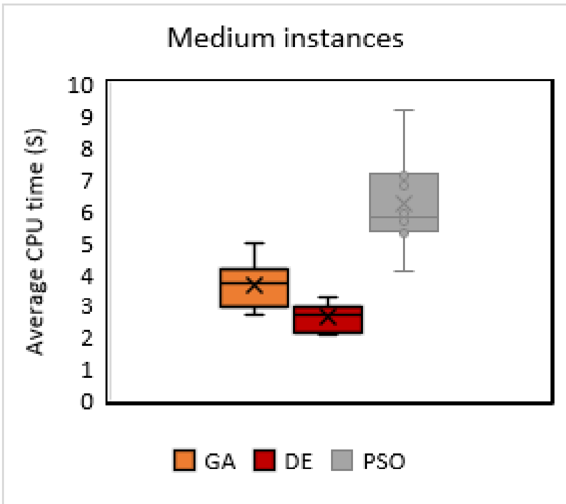
(a): Average objective function boxplot-small instances



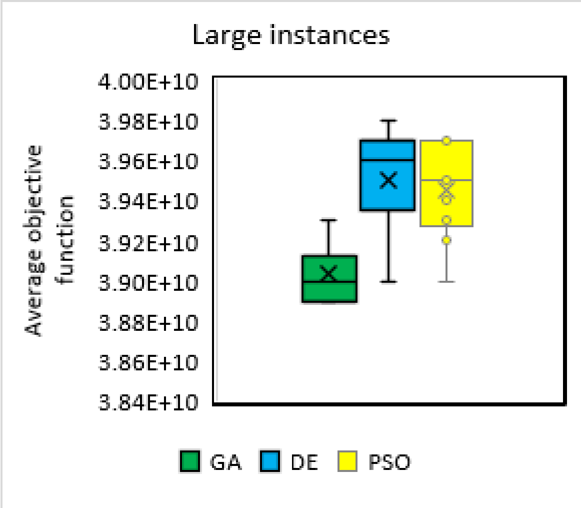
(b): Average CPU time boxplot-small instances



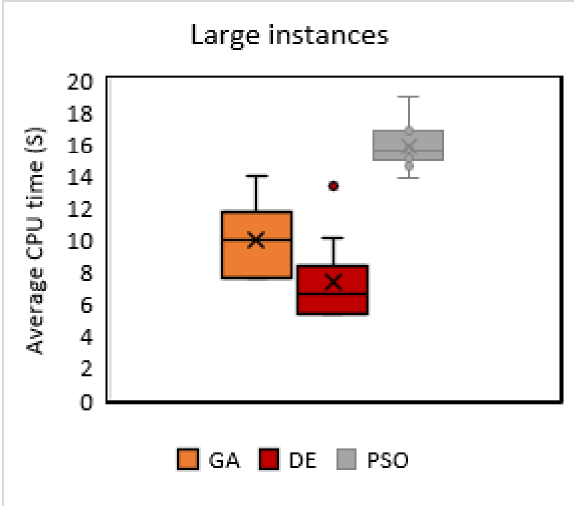
(c): Average objective function boxplot-medium instances



(d): Average CPU time boxplot-medium instances



(e): Average objective function boxplot-large instances



(f): Average CPU time boxplot-large instances

Figure 7: The boxplot of the average objective function and CPU time measures

Table 1: The novelties of current work against the previous research in the literature of inventory models for growing products

Research	Year	Number of vendors		Number of buyers		Number of products		Shortage type		Carbon emission	Coordination	Solution approach
		Single	Multiple	Single	Multiple	Single	Multiple	Backorder	Lost sale			
Rezaei [6]	2014	×	×	✓	×	✓	×	×	×	×	×	Heuristic
Zhang, et al. [30]	2016	×	×	✓	×	✓	×	×	×	✓	×	Analytical
Khalilpourazari and Pasandideh [7]	2019	×	×	✓	×	×	✓	×	×	×	×	Metaheuristic
Sebatjane and Adetunji [8]	2019	×	×	✓	×	✓	×	×	×	×	×	Heuristic
Sebatjane and Adetunji [10]	2019	×	×	✓	×	✓	×	×	×	×	×	Heuristic
Nobil, et al. [14]	2019	×	×	✓	×	✓	×	✓	×	×	×	Heuristic
Malekitabar, et al. [20]	2019	✓	×	✓	×	✓	×	×	×	×	✓	Heuristic
Mokhtari, et al. [11]	2020	×	×	✓	×	✓	×	×	×	×	×	Metaheuristic
Sebatjane and Adetunji [22]	2020	✓	×	✓	×	✓	×	×	×	×	×	Heuristic
Alfares and Afzal [9]	2021	×	×	✓	×	✓	×	✓	×	×	×	Analytical
Mahato, et al. [12]	2021	✓	×	✓	×	✓	×	×	×	×	✓	Analytical
Mittal and Sharma [13]	2021	×	×	✓	×	✓	×	×	×	×	×	Analytical
Pourmohammad-Zia, et al. [21]	2021	✓	×	✓	×	✓	×	×	×	×	✓	Heuristic
Gharaei and Almehdawe [18]	2021	×	×	✓	×	✓	×	✓	×	✓	×	Metaheuristic
De-la-Cruz-Márquez, et al. [28]	2021	×	×	✓	×	✓	×	✓	×	✓	×	Heuristic
Rana, et al. [29]	2021	×	×	✓	×	✓	×	✓	✓	✓	×	Analytical
Pourmohammadzia [15]	2022	×	×	✓	×	✓	×	×	×	×	×	Heuristic
Sebatjane and Adetunji [16]	2022	✓	×	✓	×	✓	×	×	×	×	✓	Heuristic
Current research	2024	×	✓	×	✓	×	✓	✓	✓	✓	✓	Metaheuristic

Table 2: The range of input parameters of numerical instances

Parameter	Range	Parameter	Range
AB_{ijk}	$\sim U(50,100)$	p	$\sim U(0.10, 0.12)$
AS_{ijk}	$\sim U(50,100)$	T_e	$\sim U(12,15)$
h_{ij}	$\sim U(0.002, 0.005)$	T_r	$\sim U(12,15)$
w_{ij}	$\sim U(4,9)$	Cf_j	$\sim U(0.007, 0.015)$
τ_{ij}	$\sim U(1,3)$	A_j	$\sim U(670.2, 671.2)$
D_{ijk}	$\sim U(5000,13000)$	k	0.036
β_{ij}	$\sim U(0,1)$	n	–0.0087
q	$\sim U(0.10, 0.12)$	b	–0.043

Table 3: Three considered levels for the parameter calibration of metaheuristics

Algorithm	Parameter	Parameter name	Level 1	Level 2	Level 3
GA	A	Max_{it}	50	75	100
	B	N_{pop}	20	25	30
	C	P_c	0.7	0.8	0.9
	D	P_m	0.1	0.2	0.3
DE	A	Max_{it}	50	75	100
	B	N_{pop}	20	25	30
	C	F	0.7	0.75	0.8
	D	P_c	0.1	0.2	0.3
PSO	A	Max_{it}	50	75	100
	B	N_{pop}	20	25	30
	C	c_1	1	1.50	2
	D	c_2	1.50	2	2.50
	E	w	0.90	0.95	0.99
	F	w_{damp}	0.95	0.99	1

Table 4: The dimension of three categories of numerical instances

Instance size	Number of vendors	Number of livestock	Number of ranchers
Small	6	8	6
Medium	15	20	15
Large	25	30	25

Table 5: The computational results of algorithms for small instances

Instance	Average total cost			Average RDI			Average RPD			Average CPU time			% Gap
	GA	DE	PSO	GA	DE	PSO	GA	DE	PSO	GA	DE	PSO	
#S1	**4.73E+08**	5.25E+08	5.02E+08	**0.16**	0.94	0.59	**0.02**	0.13	0.08	0.93	**0.74**	3.03	2.03
#S2	**4.92E+08**	5.36E+08	4.99E+08	**0.35**	0.81	0.55	**0.07**	0.17	0.08	1.05	**0.89**	2.60	3.09
#S3	**4.86E+08**	5.34E+08	5.08E+08	**0.44**	0.80	0.68	0.14	0.25	**0.10**	1.00	**0.82**	3.36	8.06
#S4	**4.87E+08**	5.22E+08	5.24E+08	**0.29**	0.66	0.61	**0.06**	0.13	0.09	1.02	**0.77**	3.08	7.06
#S5	**4.96E+08**	5.08E+08	5.09E+08	**0.50**	0.65	0.61	**0.09**	0.11	0.09	0.92	**0.71**	3.71	–
#S6	**4.89E+08**	5.10E+08	5.13E+08	**0.33**	0.52	0.76	**0.08**	0.13	0.11	0.93	**0.79**	4.19	4.08
#S7	**4.90E+08**	5.19E+08	5.29E+08	**0.34**	0.62	0.86	**0.11**	0.19	0.12	1.01	**0.73**	4.45	–
#S8	**4.80E+08**	5.24E+08	5.17E+08	**0.41**	0.77	0.82	0.14	0.22	**0.12**	0.89	**0.69**	2.79	5.02
#S9	**4.91E+08**	5.33E+08	4.93E+08	**0.34**	0.67	0.45	0.09	0.19	**0.06**	0.97	**0.74**	2.09	3.03
#S10	4.99E+08	5.28E+08	**4.92E+08**	**0.42**	0.77	0.45	0.08	1.14	**0.06**	2.13	**1.23**	3.15	2.07
Average	**4.88E+08**	5.24E+08	5.09E+08	**0.36**	0.72	0.64	**0.09**	0.27	0.09	1.09	**0.81**	3.25	4.30

Table 6: The computational results of algorithms for medium instances

Instance	Average total cost			Average RDI			Average RPD			Average CPU time		
	GA	DE	PSO	GA	DE	PSO	GA	DE	PSO	GA	DE	PSO
#M1	**9.10E+09**	9.34E+09	9.28E+09	**0.40**	0.76	0.58	**0.03**	0.05	0.05	2.69	**2.06**	6.74
#M2	**9.11E+09**	9.35E+09	9.25E+09	**0.37**	0.77	0.74	**0.02**	0.04	0.04	2.91	**2.08**	7.11
#M3	**9.02E+09**	9.27E+09	9.31E+09	**0.31**	0.70	0.79	**0.03**	0.06	0.05	2.93	**2.15**	5.32
#M4	**8.99E+09**	9.34E+09	9.27E+09	**0.33**	0.90	0.58	**0.02**	0.06	0.04	3.77	**2.67**	5.63
#M5	**9.05E+09**	9.21E+09	9.32E+09	**0.30**	0.45	0.54	**0.03**	0.05	0.05	4.93	**3.15**	9.14
#M6	**9.12E+09**	9.31E+09	9.19E+09	**0.40**	0.79	0.48	**0.02**	0.04	0.03	4.26	**3.25**	5.89
#M7	**9.22E+09**	9.31E+09	9.26E+09	**0.34**	0.57	0.80	**0.02**	0.04	0.04	3.57	**2.70**	4.05

#M8	**9.10E+09**	9.20E+09	9.36E+09	**0.48**	0.60	0.77	**0.04**	0.06	0.05	3.80	**2.69**	5.42
#M9	**9.19E+09**	9.23E+09	9.31E+09	**0.51**	0.68	0.73	0.06	0.07	**0.05**	4.06	**2.73**	7.32
#M10	**9.15E+09**	9.18E+09	9.22E+09	**0.47**	0.48	0.58	**1.03**	1.04	1.04	3.49	**2.85**	5.26
Average	**9.11E+09**	9.27E+09	9.28E+09	**0.39**	0.67	0.66	**0.13**	0.15	0.14	3.64	**2.63**	6.19

Table 7: The computational results of algorithms for large instances

Instance	Average total cost			Average RDI			Average RPD			Average CPU time		
	GA	DE	PSO	GA	DE	PSO	GA	DE	PSO	GA	DE	PSO
#L1	**3.89E+10**	3.97E+10	3.95E+10	**0.32**	0.76	0.65	**0.02**	0.04	0.03	11.53	**7.08**	16.77
#L2	**3.90E+10**	3.98E+10	3.93E+10	**0.25**	0.72	0.54	**0.01**	0.03	0.03	13.93	**13.34**	13.86
#L3	**3.90E+10**	3.94E+10	3.90E+10	0.50	0.73	**0.40**	**0.02**	0.03	0.02	10.78	**7.07**	14.58
#L4	**3.91E+10**	3.97E+10	3.97E+10	**0.24**	0.62	0.74	**0.01**	0.02	0.04	12.37	**10.05**	15.79
#L5	**3.90E+10**	3.96E+10	3.95E+10	**0.39**	0.74	0.62	**0.02**	0.03	0.03	10.00	**7.75**	15.46
#L6	**3.89E+10**	3.96E+10	3.92E+10	**0.37**	0.76	0.50	**0.02**	0.04	0.03	10.02	**6.29**	15.14
#L7	3.93E+10	**3.92E+10**	3.95E+10	**0.39**	0.36	0.65	**0.02**	0.02	0.03	8.17	**5.80**	15.73
#L8	**3.89E+10**	3.90E+10	3.94E+10	**0.24**	0.31	0.62	0.02	**0.01**	0.03	7.63	**5.36**	15.17
#L9	**3.92E+10**	3.94E+10	3.97E+10	**0.21**	0.35	0.76	**0.00**	0.01	0.04	7.62	**5.36**	18.96
#L10	**3.90E+10**	3.96E+10	3.97E+10	**0.10**	0.70	0.74	1.00	1.02	1.04	7.69	**5.33**	17.10
Average	**3.90E+10**	3.95E+10	3.95E+10	**0.30**	0.61	0.62	**0.11**	0.13	0.13	9.97	**7.34**	15.86

Biography

Ali Fallahi received his MSc degree in Industrial Engineering from Department of Industrial Engineering, Sharif University of Technology, in 2022. His research interests are in the area of inventory control, supply chain management, and optimization in healthcare. Ali has published several papers in reputable international journals such as Computers & Industrial Engineering, Expert Systems with Applications, Applied Soft Computing, Journal of Cleaner Production, IEEE Access, and Soft Computing.

Yeganeh Tehrani received her BSc degree in Industrial Engineering from Department of Industrial Engineering, University of Kashan, in 2022. Her research interests are in the area of inventory planning and supply chain optimization.

Hadi Mokhtari is currently an Associate Professor of Industrial Engineering at University of Kashan, Iran. His current research interests include the applications of operations research and artificial intelligence techniques to the areas of project scheduling, production scheduling, manufacturing supply chains, and engineering economic problems. He has also published several papers in international journals such as Computers & Operations Research, International Journal of Production Research, Applied Soft Computing, Journal of Cleaner Production, Neurocomputing, International Journal of Advanced Manufacturing Technology, IEEE Transactions on Engineering Management, and Expert Systems with Applications