IJE TRANSACTIONS A: Basics Vol. 37 No. 07, (July 2024) 1408-1421

International Journal of Engineering

Journal Homepage: www.ije.ir

Bi-objective Economic Production Quantity with Partial Backordering under Uncertainty

M. Najafi[a], A. Ghodratnama*[a], S. H. R. Pasandideh[a], R. Tavakkoli-Moghaddam[b]

[a] *Department of Industrial Engineering, Faculty of Engineering, Kharazmi University, Tehran, Iran*
[b] *School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran*

PAPER INFO

Paper history:
Received 27 August 2023
Received in revised form 22 November 2023
Accepted 27 December 2023

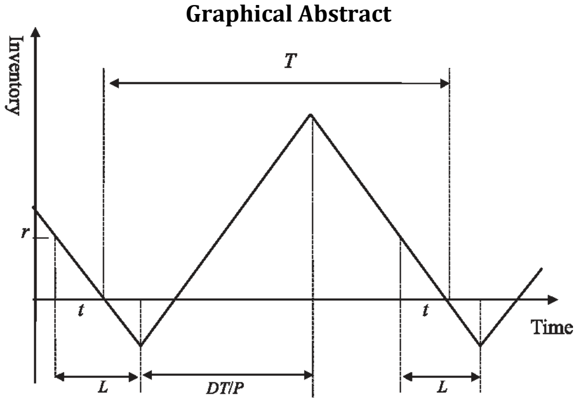
Keywords:
Bi-objective Mathematical Model
Economic Production Quantity
Rework and Shortage
Meta-heuristics
Uncertainty

ABSTRACT

The economic production quantity (EPQ) model considers the production rate, demand rate, setup costs, holding costs, and shortage costs to find the production quantity that minimizes the sum of these costs. The goal is to balance the costs associated with production, holding inventory, and potential shortages. In this paper, two objectives include the costs of production and ordering and others in a separate objective function. In the objectives of the other costs, The cost of storage space as a supply is defined to be minimized. This study considers scrap and reworks in the EPQ model. This inventory model accounts for many items on a single machine. The production capacity is reduced, and there are shortages when only one machine exists. By determining the quantities of the products produced by the manufacturing facility, the storage space for each product, cycle time, and product scarcity, we can reduce both the overall cost and the supply cost of warehouse space due to non-linearity and the inability to solve commercial software in large dimensions, a multi-objective meta-heuristic algorithm, namely the non-dominated sorting genetic algorithm (NSGA-II), is used. The findings are further validated using the non-dominated ranking genetic algorithm (NRGA). Also, the obtained Pareto front is studied with several indicators. To perform these two algorithms at the best condition, we employed the Taguchi approach and related orthogonal arrays and performed algorithms for each array considering several factors. Also, to validate the mathematical model, we used the augmented epsilon-constraint method executed in the GAMS environment. It is clear that GAMS commercial software yields better results; however, these two algorithms are justifiable when the problem becomes bigger. Finally, by performing a sensitivity analysis for these indicators and the objective functions, the behavior of the proposed algorithms is compared and examined in detail. Also, the superior algorithm is chosen using the TOPSIS as a multi-criteria decision-making method. Numerical examples show how the presented model and the proposed algorithms may be used efficiently. A surveying literature review clarifies that the related objective functions, constraints, and solution approaches have not been investigated until now.

doi: *10.5829/ije.2024.37.07a.18*

Graphical Abstract



On-hand inventory for perfect quality items

*Corresponding Author's Email: ghodratnama@khu.ac.ir (A. Ghodratnama)

Please cite this article as: Najafi M, Ghodratnama A, Pasandideh SHR, Tavakkoli-Moghaddam R. Bi-objective Economic Production Quantity with Partial Backordering under Uncertainty. International Journal of Engineering, Transactions A: Basics. 2024;37(07):1408-421.

NOMENCLATURE

d_i	Demand for the *i*-th product	A_i	Setup expense for producing a batch of the *i*-th product
p_i	Production rate of the *i*-th product	ts_i	Duration of machine setup to manufacture the *i*-th product
u_i	Paulty products of the *i*-th product	o_i	Supply cost per unit of storage space
v_i	Product backorders as a percentage of its shortfall	f_i	Space occupied by each unit of the *i*-th product
c_i	Manufacturing cost per unit of the *i*-th product	q_i^s	For each cycle (i = 1, 2,..., n), the size of the manufacturing lot for the *i*-th product
k_i	Reworking the costs per item of the *i*-th product	$T \geq 0$	Cycle's duration
h_i	Price per item and holding time for the *i*-th product	$N \geq 0$	Cycles per year, number
b_i	Backorder price for the *i*-th product	$X_i \geq 0$	Continuous random variable represents the product's storage space.,
sc_i	Scrap percentage of the *i*-th product	$s_i \geq 0$	Overall shortfall amount for the *i*-th product of the cycle
H_i	After normal manufacturing ends, the product's maximum amount of inventory	$H_i^{\max}$	Final amount of available inventory for the product

1. INTRODUCTION

Inventory control is crucial for maintaining inventory levels and minimizing system costs. It ensures that an organization's existing items are available for production, distribution, sales, and engineering operations management departments, considering factors like time, location, quantity, quality, and cost. Inventory control includes raw materials and products stored in the warehouse. Operations management involves designing and managing products, processes, services, and supply chains. It includes strategic, tactical, and operational levels.

Improving competitive power and comprehensiveness in supply chains is essential for efficient systems. Although the economic production quantity model is often used in inventory management and manufacturing, it is crucial to investigate if damaged goods are included in inventory models. This study investigates the economic production quantity (EPQ) model during production periods for different time components, shortages, and lost sales, focusing on production costs, ordering, and storage space.

However, the EPQ model is specifically designed for situations where items are produced or manufactured rather than simply ordered. It is often applied to scenarios where production rates are finite and may vary. The EPQ model takes into account factors such as production rate constraints, setup costs, holding costs, and demand for the product. The key components of the EPQ model include:

- Demand: The rate at which customers are requesting the product.
- Setup (or production) cost: The cost associated with setting up the production process, including the cost of preparing the machinery, changing tools, etc.
- Holding (or carrying) cost: The cost of holding or storing inventory, including expenses related to warehousing, insurance, and potential obsolescence.
- Production rate: The rate at which units are produced.

The goal of the EPQ model is to find the production quantity that minimizes the total cost, taking into consideration the trade-off between setup costs and holding costs. The formula for the Economic Production Quantity is derived based on mathematical optimization techniques, and it helps businesses determine the most cost-effective production quantity to meet demand.

This paper's remaining sections are organized as follows.. Section two scrutinizes the related literature review meticulously. The mathematical formulation of the issue is covered in section 3. The methods for the solutions are in section 4. Section 5 deals with the solution and comparison of numerical instances. The conclusion and some recommendations for more research are included in section 6. References are also included in section 7.

2. LITERATURE REVIEW

Cunha et al. (1) examined the economic production quantity model with partial backordering and a discount for batches of subpar quality. Shah and Vaghela (2) created and refined a flawed production inventory model for time- and effort-dependent demand under inflation and optimum dependability. Taleizadeh et al. (3) developed sustainable economic output quantity models for shortage inventory systems. Al-Salamah (4) investigated how much economic production might be produced in a manufacturing process that included faults and configurable synchronous and asynchronous rework rates. The economic order quantity (EOQ) and EPQ inventory models with two backorder charges were developed by Lin (5) using analytical geometry and algebra. Marchi et al. (6) examined the economic production quantity model. It includes learning production, quality reliability, and energy efficiency.

The EOQ and EPQ inventory models with partial backorder issues were studied by Thinakaran et al. (7). For the economic output quantity and the joint economic lot size, Zavanella et al. (8) considered energy. To resolve an EPQ model for an inefficient manufacturing process, De et al. (9) employed a game-based approach and a neutrosophic fuzzy method. Ganesan and Uthayakumar (10) created EPQ models for a flawed manufacturing system that considers warm-up production runs, shortages during hybrid maintenance periods, and partial backordering.

Guha and Bose (11) presented the EPQ in Batch Manufacturing with imperfect quality and non-destructive acceptance sampling. Insights from an EPQ model were investigated by Hauck et al. (12) on the impact of early inspection on the functionality of production systems. According to Kalantari and Taleizadeh (13), mathematical modeling may be used to find the best replacement for failed items in an EPQ model with several shipments. In 2020, that was formulated by Nobil et al. (14). An economic production quantity inventory model with discrete delivery orders, common production standards, and budgetary constraints for several items produced on a single machine. The Development and Solvency were reported by Rahaman et al. (15).

The production model in terms of amount under arbitrary commands both with and without deterioration. Artificial bee colony optimization was the basis for Rahman et al.'s (16) synergetic analysis of the fractional-order economic production quantity model. Economic output quantity models that are sensitive to forecasting maintenance and modified later on. An economic production quantity model for three tiers of work was designed by using the Weibull distribution degradation and shortage. Bose and Guha (17) looked at the economic production lot size under the conditions of low quality, online inspection, and inspection errors. A comment on the cost comparison method used to address the EOQ and EPQ concerns.

Using fuzzy geometric programming (GP), different fuzzification and defuzzification approaches, and an unconstrained multi-item model, Kalaiarasi et al. (18) claim that this model was optimized. Moghdani et al. (19) considered a multi-item fuzzy economic production quantity model with multiple deliveries. Shekhawat et al. (20) looked at the EPQ model for deteriorating items with a Weibulleterioration rate throughout the finite time horizon. To solve a multi-product, single-machine EPQ inventory model utilizing GP mode, Kalaiarasi et al. (21) employed Python.

A quantitative model of economic production with a fluctuating energy price has been constructed. Nobil et al. (22) considered a setup time/cost function for a multi-product imperfect manufacturing system and an economic production quantity inventory model. Priyan et al. (23) examined a cleaner EPQ inventory model with synchronous and asynchronous rework procedures and investments in green technologies. Edalatpour et al. (24) integrated sustainability concerns with pricing and inventory decisions for degrading items. Also, some researchers studied other aspects of EPQ issue (25-27). Figure 1 illustrates the number of papers published in this regard.

According to the considerable literature on the economic production quantity model, it is evident that interest in the topic of an incomplete production system, a problem that affects real-world manufacturing, is still expanding. Since the model created in this work is challenging to solve analytically, Pareto fronts are discovered using a non-dominated sorting genetic algorithm (NSGA-II) and the non-dominated ranking genetic algorithm (NRGA).

3. PROPOSED MATHEMATICAL MODEL

In this section, the assumptions of the proposed model are explained first, and then the parameters and variables are defined; in the following, the problem's mathematical model and the constraints' definition are discussed in detail.

3. 1. Assumptions The main assumptions of the proposed model are as follows:

- Considering a manufacturing system with flawed production procedures.
- At a rate of u_i; $i=1,2,\ldots,n$ every cycle, incomplete objects of n various sorts are produced, and among these goods, the SC_i part is considered to be junk, while the other portion may be reworked.
- In each cycle, most parameters are seen as unknown.
- The parameters are produced randomly using uniform distributions in the respective ranges for various issues.
- One machine is used to make all of the goods.
- All things are believed to have a certain cycle length, $T_1=T_2=\ldots=T_n=T$.
- Assume that the number of goods produced corresponds to a corresponding demand in every cycle, with a production rate of pi per cycle.

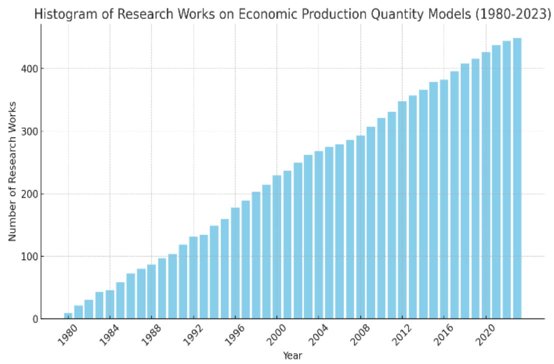


Figure 1. Histogram representing the number of papers published in this regard

- It is assumed that the number of goods produced, which corresponds to a corresponding demand, d_i, in every cycle, has a production rate of p_i per cycle for the *i*-th item.
- At the end of the rework period, we expect that all reworkable objects will be reworked, and one part will be left behind as scrap
- We expect all of the reworkable objects to be reworked, and a m_i part will be left behind as scrap after the rework period.
- Producers use the same resource for production and rework simultaneously.
- The budget and capacity for the standard manufacturing system are limited, and some shortfalls are being backordered. The fundamental concept of the EPQ inventory model with the rework process is that the production rate of fewer defectives must always be more than or equal to the demand (28). The production cycle length is determined as the average of the good and incomplete item production up times (t_i^1 and t_i^5, respectively), the reworking time (t_i^2), and the good and incomplete item production downtimes (t_i^3 and t_i^4 , respectively).

$$T=\sum_{j=1}^{5} t_i^j \tag{1}$$

Figure 2 depicts the cycle time for each product since all products are produced on a single machine with a limited capacity. This has led to the following equations:

$$t_i^1=\frac{q_i^s}{p_i}-\frac{v_i s_i}{(1-u_i-sc_i)p_i-d_i} \tag{2}$$

$$t_i^1=u_i\frac{q_i^s}{p_i} \tag{3}$$

$$t_i^3=\frac{H_i^{\max}}{d_i}=\left(\frac{(1-sc_i-m_i u_i)}{d_i}-\frac{(1+u_i)}{p_i}\right)q_i^s-\frac{v_i s_i}{d_i} \tag{4}$$

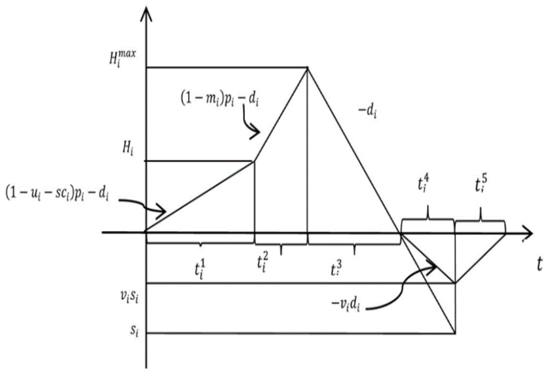


Figure 2. On-hand inventory for perfect quality items

$$t_i^4=\frac{s_i}{d_i} \tag{5}$$

$$t_i^5=\frac{v_i s_i}{(1-u_i-sc_i)p_i-d_i} \tag{6}$$

Thus, based on Equation 1, the length of the cycle for a single product is as follows:

$$H_i=\left((1-u_i-sc_i)p_i-d_i\right)\frac{q_i^s}{p_i}-v_i s_i \tag{7}$$

$$H_i^{\max}=H_i+u_i\left((1-m_i)p_i-d_i\right)\frac{q_i^s}{p_i}$$

$$H_i^{\max}=\left((1-sc_i-m_i u_i)p_i-(1+u_i)d_i\right)\frac{q_i^s}{p_i}-v_i s_i \tag{8}$$

$$T=\frac{(1-v_i)s_i+(1-sc_i-m_i u_i)q_i^s}{d_i} \tag{9}$$

$$q_i^s=\frac{Td_i-(1-v_i)s_i}{(1-sc_i-m_i u_i)} \tag{10}$$

3. 2. Function of the Total Cost

The following describes the model's total cost function:

$$TC=CA+CP+CR+CH+CB+CL \tag{11}$$

3. 2. 1. Setup Cost

A setup costs and occurs N times annually.

As a result, the yearly setup cost is as follows:

$$CA=\sum_{i=1}^{n} NA_i \tag{12}$$

$$N=\frac{1}{T} \tag{13}$$

3. 2. 2. Production Cost

The sum of the total production cost is the sum of the production costs per unit and quantity per period for all *i*-th commodities, respectively of manufacturing annually:

$$CP=\frac{1}{T}\sum_{i=1}^{n} c_i q_i^s \tag{14}$$

3. 2. 3. Rework Cost

k_i reflects how much of the *i*-th product has to be changed. The sum of the rework cost per unit of the *i*-th product is what is referred to as the annual rework cost. The year of work may be determined by multiplying the total cost by N. The cost of this shared insurance is as follows:

$$CR=\frac{1}{T}\sum_{i=1}^{n}k_iu_iq_i^s \tag{15}$$

3. 2. 4. Holding Cost The holding costs of the inventory system for independent and collaborative production strategies expressed in Equation 16 as shown in Figure 1,

$$CH=\frac{1}{T}\sum_{i=1}^{n}h_i\left[\frac{H_i}{2}\left(t_i^1\right)+\frac{H_i+H_i^{\max}}{2}\left(t_i^2\right)+\frac{H_i^{\max}}{2}\left(t_i^3\right)\right] \tag{16}$$

3. 2. 5. Backorder Cost Related to Figure 1, expressions 17 and 18 indicate the back-ordered and lost selling expenditures for each cycle.

$$CB=\frac{1}{2T}\sum_{i=1}^{n}b_iv_is_i\left(t_i^4+t_i^5\right) \tag{17}$$

$$CL=\frac{1}{2T}\sum_{i=1}^{n}l_i\left(1-v_i\right)s_i \tag{18}$$

3. 2. 6. Lost Sale Cost As a consequence, the model's objective function is as follows:

$$\begin{aligned}TC&=CA+CP+CR+CH+CB+CL\\&=\frac{1}{T}\sum_{i=1}^{n}A_i+\frac{1}{T}\sum_{i=1}^{n}c_iq_i^s+\frac{1}{T}\sum_{i=1}^{n}k_iu_iq_i^s\\&+\frac{1}{T}\sum_{i=1}^{n}h_i\left[\frac{H_i}{2}\left(t_i^1\right)+\frac{H_i+H_i^{\max}}{2}\left(t_i^2\right)+\frac{H_i^{\max}}{2}\left(t_i^3\right)\right]\\&+\frac{1}{2T}\sum_{i=1}^{n}b_iv_is_i\left(t_i^4+t_i^5\right)+\frac{1}{2T}\sum_{i=1}^{n}l_i\left(1-v_i\right)s_i\end{aligned} \tag{19}$$

3. 5. Cost of Storage Space as a Supply The supply cost of a warehouse is determined as the product of the supply cost per storage space and the continuous random variables X_i representing the storage area of a particular product, respectively.

$$G=\sum_{i=1}^{n}o_iX_i \tag{20}$$

3. 6. Constraints The related constraints are as follows:

3. 6. 1. Capacity Constraint In collaborative production systems that include rework, the combined production, rework, and setup times need to be less than the cycle time. In our issue, T must be less than or equal to $\sum_{i=1}^{n}\left(t_i^1+t_i^2+t_i^5\right)+\sum_{i=1}^{n}ts_i$. As a result, this is the capacity-constrained model:

$$\sum_{i=1}^{n}\left(t_i^1+t_i^2+t_i^5\right)+\sum_{i=1}^{n}ts_i\le T \tag{21}$$

Equations 2, 3, and 6 give the capacity constraint model as follows:

$$\sum_{i=1}^{n}\left(1+u_i\right)\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_iu_i\right)p_i}+\sum_{i=1}^{n}ts_i\le T \tag{22}$$

3. 6. 2. Budget Constraint Given that the entire budget is W, the manufacturing quantity is represented, and the i-th is reworked $u_iq_i^s$. The current budget constraint is as follows:

$$\sum_{i=1}^{n}\left(c_iq_i^s+k_iu_iq_i^s\right)\le W \tag{23}$$

3. 6. 3. Service Level Constraint For the service level constraint, the i-th product's annual demand, safety margin for allowable shortage, period-by-period shortfall amount, and several periods are, in that order: S_i, d_i, SL, and N. The current service level limitation is as follows:

$$\sum_{i=1}^{n}\frac{s_i}{Td_i}\le SL \tag{24}$$

Constraints 22-24.

3. 6. 4. Warehouse-Space Constraint There is definite room in the warehouse to keep the goods.

$$f_iH_i^{\max}\le X_i \tag{25}$$

3. 7. Final Model

$$\begin{aligned}MinZ&=\frac{1}{T}\sum_{i=1}^{n}A_i+\frac{1}{T}\sum_{i=1}^{n}C_i\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_iu_i\right)}+\\&\frac{1}{T}\sum_{i=1}^{n}k_iu_i\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_iu_i\right)}+\frac{1}{T}\sum_{i=1}^{n}h_i\\&\left[\begin{array}{l}\frac{1}{2}\left(\left(\left(1-u_i-sc_i\right)p_i-d_i\right)\left(\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_iu_i\right)p_i}\right)-v_is_i\right)\\\left(\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_iu_i\right)p_i}-\frac{v_is_i}{\left(1-u_i-sc_i\right)p_i-d_i}\right)+\\\left(\left(\begin{array}{l}\left(\left(1-0.5u_i-sc_i-0.5m_iu_i\right)-\left(1+0.5u_i\right)\frac{d_i}{p_i}\right)\\\left(\frac{\left(Td_i-\left(1-v_i\right)s_i\right)}{\left(\left(1-sc_i-m_iu_i\right)p_i\right)}\right)\end{array}\right)-v_is_i\right)\\\left(u_i\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_iu_i\right)p_i}\right)\\\frac{1}{2d_i}\left(\left(\left(\left(1-sc_i-m_iu_i\right)p_i-\left(1+u_i\right)d_i\right)\right)\left(\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_iu_i\right)p_i}\right)-v_is_i\right)^2\end{array}\right]\\&+\frac{1}{2T}\sum_{i=1}^{n}b_iv_is_i\left(\frac{s_i}{d_i}+\frac{v_is_i}{\left(1-u_i-sc_i\right)p_i-d_i}\right)+\frac{1}{2T}\sum_{i=1}^{n}l_i\left(1-v_i\right)s_i\end{aligned} \tag{26}$$

$$Min\ G=\sum_{i=1}^{n} o_i X_i$$

s.t.

$$\sum_{i=1}^{n}\left(c_i+k_i u_i\right)\left(\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_i u_i\right)}\right)\le W \tag{27}$$

$$f_i\left(\left(\left(1-sc_i-m_i u_i\right)p_i-\left(\left(1+u_i\right)d_i\right)\right)\left(\frac{Td_i-\left(1-v_i\right)s_i}{\left(1-sc_i-m_i u_i\right)p_i}\right)-v_i s_i\right)\le X_i \tag{28}$$

4. SOLUTION APPROACHES

In terms of approaches to solutions For reactive berth allocation and scheduling at maritime container ports in reaction to disturbances, Dulebenets (29) took into consideration a dispersed memetic optimizer. An adaptable polyploid memetic algorithm was also considered by Dulebenets (30) for truck scheduling at a cross-docking terminal. In multi-objective settings, Pasha et al. (31) used a factory-in-a-box to look at precise and metaheuristic algorithms for the vehicle routing issue. Singh and Pillay (32) considered analyzing ant-based pheromone spaces for generating perturbative meta-heuristics. The development of precise and heuristic optimization techniques for safety enhancement projects at level crossings under competing goals was considered by Singh et al. (33). Chen and Tan (34) provide a quick, self-adaptive, efficient fireworks approach for large-scale optimization. An effective multi-objective metaheuristic algorithm for the sustainable harvest planning issue was considered by Fathollahi-Fard et al. (35).

4. 1. Augmented Epsilon-Constraint Method

The augmented epsilon-constraint method is a technique used in multi-objective optimization to handle constraints in the optimization process. Multi-objective optimization involves optimizing multiple conflicting objectives simultaneously, and constraints are conditions that must be satisfied for a solution to be considered feasible.

In the augmented epsilon-constraint method, the idea is to transform the constrained multi-objective optimization problem into an unconstrained one by introducing additional variables and constraints. The method is particularly useful when dealing with problems where finding feasible solutions is challenging.

Here's a basic overview of the augmented epsilon-constraint method:

- **Original Problem:** Let's say you have a multi-objective optimization problem with objectives $f_1(x), f_2(x),..., f_m(x)$, and constraint functions, $g_1(x), g_2(x),..., g_p(x)$ where x is the vector of decision variables.
- **Introduce Slack Variables:** Introduce slack variables $(\varepsilon_1, \varepsilon_2,..., \varepsilon_p)$ to represent the violation of each constraint. These slack variables are non-negative and measure how much a solution violates a particular constraint.
- **Transform Constraints:** Transform the original constraints into equality constraints using the slack variables. The transformed constraints may look like this:

$g_i(x)+\varepsilon_i=0 \quad i=1,2,...,p$ where $\varepsilon_i \ge 0$

- **Augmented Objective Function:** Modify the objective functions to penalize violations of constraints. The augmented objective function may include a penalty term that depends on the slack variables:
- $F(x)=\left(f_1(x), f_2(x),..., f_m(x)\right), \lambda_1 \in 1, \lambda_2 \in 2,..., \lambda_p \in p$

  Here, $\lambda_1, \lambda_2,..., \lambda_p$ are penalty coefficients.
- **Optimization:** Solve the augmented unconstrained problem using a multi-objective optimization algorithm. The algorithm seeks to optimize the augmented objective function, and the penalty terms encourage the optimization process to find solutions that minimize violations of constraints.
- **Post-Processing:** After obtaining solutions from the augmented problem, analyze the trade-offs between conflicting objectives and check the values of slack variables to ensure constraint satisfaction.

The augmented epsilon-constraint method helps convert a constrained multi-objective optimization problem into a form that can be addressed by standard multi-objective optimization algorithms. This approach allows for a more flexible and efficient exploration of the solution space in the presence of constraints (36).

Hybrid multiobjective optimization problems involve single-objective optimization using multicriteria decision-making methods and single-objective evolutionary algorithms (SOEA) like NSGA-II, NRGA, and MOPSO for finding Pareto optimal fronts in a single simulation run.

4. 2. NSGA-II

Deb et al. (37) developed the NSGA-II, a GA-based multi-objective optimization technique. They created a random parent population of size (*nPop*), assessed objective values, and sorted using the nondomination method. They selected two individuals and formed a new offspring population with *nPop* + *n* sizes. The NSGA-II implementation produced non-dominated Pareto-optimum solutions.

4. 3. NRGA

The NRGA works identically to NSGA-II, except for selecting and reproducing the parents in the mating pool. Before using the Pareto-based population-ranking approach, one of the fronts is chosen using the ranked-based roulette wheel (RBRW) selection operator, initially proposed by Al Jadaan et al. (38, 39). The Pseudo-code of the NSGA-II stated in Figure 3.

The next step is to choose one option from the candidate front using the same method. Consequently, the chances of picking a solution inside the best non-dominated set of the initial front are greatest. In contrast, solutions inside a set of the second front have lower probability, and so on.

4. 4. Algorithms' Characteristics

4. 4. 1. Chromosome Structure Chromosomes are gene collections arranged in a specific order. Designing an appropriate chromosomal structure is crucial for algorithm execution. The study's chromosomal solution is represented as a 2n matrix, displaying storage space, low supply, shortfall, and storage locations for each commodity. Figure 4 shows this graphically.

4. 5. Algorithms' Mechanism This part explains the crossover, mutation, evaluation function, and stop criteria of the two algorithms' four key characteristics.

4. 5. 1. Crossover Operator A crossover process involves DNA switching between parent chromosomes, creating better chromosomes and favorable genes for offspring. This study generates offspring using linear chromosome vectors and arithmetic crossover operators with a random weighting factor

$$offspring_1 = \alpha \times parent_1 + (1-\alpha) \times parent_2 \quad (29)$$

$$offspring_2 = (1-\alpha) \times parent_1 + \alpha \times parent_2 \quad (30)$$

4. 5. 2. Mutation Operator Mutation preserves genetic diversity in populations, protecting against data loss. In this regard, firstly, we chose a normal random variable, then by substituting in Expression 32 regarding γ between zero and one, we obtain β after that if β is positive Expression 34 is regarded else Expression 36 is taken into account. Note that Upperbound is one and lowerbound is zero; they are updated during algorithm processing.

---

1. Create *Npop* random solutions (Initialization)
2. Determine the values of the objective function for the first solutions.
3. Determine rankings for the solutions using Goldberg's ranking method
4. Determine the crowding distance.
5. While stop requirements are not met
    a. Create the mating pool and add individuals to it using binary tournament selection.
    b. To the mating pool, apply the crossover and mutation operators.
    c. To the new solutions' objective function values.
    d. Combine the existing populace with the newly developed solutions.
    e. Determine rank using the Goldberg's ranking method.
    f. Determine the crowding distance.
    g. Group people and choose superior options.

Stop while

Figure 3. Pseudo-code of the NSGA-II

$$\begin{matrix} X_i \\ s_i \end{matrix} \begin{bmatrix} X_1\, X_2 \ldots X_n \\ s_1\ s_2 \ldots\ s_n \end{bmatrix}$$

Figure 4. Structure of a chromosome

$$z \sim Normal(0,1) \quad (31)$$

$$\beta = \tanh(\gamma \times z) \quad (32)$$

$$if \quad \beta \geq 0 \quad (33)$$

$$offspring = parent + \beta \times (Uperbound - parent) \quad (34)$$

$$else \quad (35)$$

$$offspring = parent + \beta \times (parent - Lowerbound) \quad (36)$$

4. 5. 3. Controlled Elitism In the NSGA-II, elitism is regulated to balance maintaining high-quality solutions and preserving diversity in the population. This is crucial because excessive elitism can lead to premature convergence (where the algorithm converges to sub-optimal solutions early). In contrast, insufficient elitism can lead to a loss of good solutions.

4. 5. 4. Evaluation The fitness of chromosomes in each generation is assessed using the optimization model's objective function. However, the inventory model has four limitations, making synthetic chromosomes unlikely. A penalty function is used to increase the likelihood of constraint violations, defining the penalty and fitness function.

$$Penalty\ function = \begin{cases} \sum_{i=1}^{4} \frac{E_i}{4} & If\ the\ chromosome \in \text{infeasible region} \\ 0 & If\ the\ chromosome \in \text{feasible region} \end{cases} \quad (37)$$

Fitness function = Penalty function + Objective function

4. 5. 5. Stopping Criterion A stopping condition in the MOEA leads to Pareto-optimal solutions, eliminating the need for mutation and crossover operators. After a certain number of generations, algorithms are believed to be finished. Finding an early perfect solution requires both the NSGA-II and NRGA algorithms in MATLAB 8.20 and statistical techniques.

5. NUMERICAL RESULTS

In this step, twenty test problems are regarded, ranging from the size of the two products to twenty products. Also, related parameters are generated randomly following uniform distribution between associated bounds. Table 1 displays the ranges' of numerical data. Firstly, we report solutions obtained by GAMS commercial software for all twenty test problems as the best-quality solutions. For the augmented epsilon-constraint method, this approach outperforms two other solution approaches regarding the quality metric index. Thus, we did not consider it. The number of Pareto front solutions oscillates between 400 and 500. However, for other criteria, no specific trend exists. However, the Pareto front curve is the best compared to other methods, and convexity is outward.

To perform in the best condition regarding two metaheuristic approaches. We chose six factors for each of the three levels. Table 3 shows the factors and related levels. Then, using Minitab software, we generated orthogonal arrays, and for specific levels, we obtained the metrics mentioned above. We obtained weighted values after normalizing the solution and determining positive or negative indexes. At last, the distance from the maximum value of weighted sum values is computed. The distance from the maximum as the best solution should be minimized. Tables 4 to 5 show the numerical results, and Figures 5 to 6 show the optimum level of regarded factors (less response value is better).

We used the Topsis approach, a multi-attribute decision-making approach, to decide which algorithm was better. Results show that the NRGA outperforms the NSGA-II regarding related indexes. For further information refer to literture [32].

TABLE 1. Parameter range

A ~uniform [500 1900]	*m* ~uniform [0.02 0.04]	*b* ~uniform [5 33]
c ~uniform [6 34]	*u* ~uniform [0.05 0.25]	*l* ~uniform [1 29]
d ~uniform [150 1000]	*k* ~uniform [1 15]	*ts* ~uniform [0.0003 0.0007]
v ~uniform [0.5 0.7]	*h* ~uniform [2 30]	
sc~uniform[0.045 0.065]	*p* ~uniform [5000 12000]	*f* ~uniform [2 5]
		o ~uniform [3 10]
W=20000000 *SL*=0.99999		

TABLE 2. Problems vs. obtained indexes related to the augmented epsilon-constraints method executed by GAMS commercial software

Problem	Pareto front metric		Objective function mean values		Computational time
	No. of non-dominated solutions	Spacing metric	Mean z_1	Mean z_2	
1	429	6.0161	33020.6	11022.1	711
2	491	6.2039	68276.68	7271.221	291
3	329	0.08952	109109	13461.52	229
4	492	5.7082	126659.1	17500.58	270
5	490	6.0684	156536.3	20625.78	256
6	389	7.2765	190277.9	34970.78	271
7	490	3.927	223838.8	32008.53	309
8	492	4.1316	225652.9	25609.38	345
9	491	3.1453	206231.8	30771.12	318
10	2	0	151504.9	145595.4	37
11	1	0	176127	1119.57	20
12	1	0	205488.2	1190.5	20
13	1	0	177252.2	1432.92	20
14	Infeasible				
15	492	2.1907	351029.9	49065.39	338
16	490	1.856	401399.8	54128.24	435
17	489	2.0345	449593.3	59183.4	539
18	491	1.8456	424752	77587.54	533
19	489	1.563	521761.7	62907.14	417
20	490	1.6378	470140	62391.54	332

TABLE 3. Considered structural parameters and levels of both algorithms

Factors	Levels		
	1	2	3
nPop	30	50	70
MaxIt	30	50	70
pCrossover	0.3	0.5	0.7
pMutation	0.3	0.5	0.7
Gamma	0.3	0.5	0.7
ControlledElitisimParameter	0.3	0.5	0.7

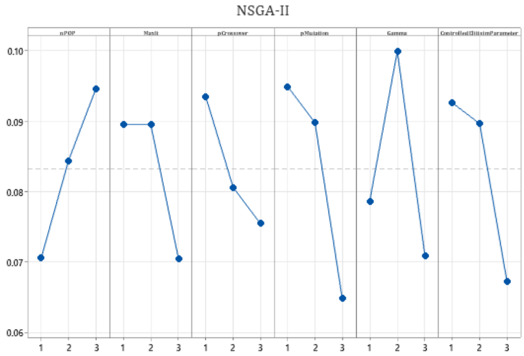


Figure 5. Levels vs. response values related to the NSGA-II algorithm (less response value is better)

TABLE 4. Orthogonal arrays vs. obtained indexes related to the NSGA-II

Array	Pareto front metric			Objective function mean values		Computational time
	Number of non-dominated solutions	Spacing metric	QM	Mean z_1	Mean z_2	
1	18	2.18	1	5.37E+06	1.73E+05	2.385609
2	6	1.92	0	5.75E+06	1.51E+05	2.827144
3	6	1.35	1	4.59E+06	1.56E+05	2.868053
4	7	1.25	0	3.65E+06	1.27E+05	5.840853
5	7	1.74	0	3.39E+06	1.22E+05	6.133792
6	4	0.44	1	2.00E+06	7.32E+04	6.862424
7	4	0.94	1	2.19E+06	7.37E+04	12.0195
8	2	0	0	1.21E+06	3.96E+04	12.77704
9	1	0	1	6.85E+05	2.38E+04	13.75613
10	15	0.57	0	3.51E+06	1.27E+05	10.614
11	1	0	1	2.37E+06	8.86E+04	11.75421
12	12	1.226	0	4.49E+06	1.47E+05	13.51858
13	19	2.38	0	4.20E+06	1.52E+05	13.30175
14	7	1.59	1	3.28E+06	1.03E+05	14.68792
15	6	0.93	0	2.80E+06	8.92E+04	15.39808
16	4	0.63	1	2.54E+06	8.74E+04	18.23107
17	5	1.75	0	3.07E+06	1.09E+05	17.88389
18	1	0	0	1.48E+06	4.86E+04	21.27136
19	7	1.47	1	3.87E+06	1.37E+05	18.79754
20	2	0	0.22	3.17E+06	1.11E+05	19.69584
21	4	1.076	0.2	3.00E+06	1.01E+05	19.88637
22	1	0	1	2.31E+06	7.88E+04	29.65523
23	4	1.21	0	2.21E+06	7.79E+04	29.67642
24	2	0	0	1.42E+06	5.02E+04	32.4939
25	11	1.13	0	3.13E+06	1.05E+05	32.45348
26	4	0.066	0	2.38E+06	8.18E+04	34.08444
27	2	0	1	9.92E+05	3.22E+04	42.43041

TABLE 5. Orthogonal arrays vs. indexes related to the NRGA

Array	Pareto front metric			Objective function mean values		Computational time
	Number of non-dominated solutions	Spacing metric	QM	Mean z_1	Mean z_2	
1	21	2.14	0	7.01E+05	1.88E+05	2.628553
2	12	1.39	1	4.66E+05	1.40E+05	2.805269
3	8	0.88	0	6.22E+05	1.92E+05	2.751699
4	16	0.913	1	2.71E+05	9.74E+04	6.375257
5	2	0	1	3.38E+05	1.02E+05	7.816881
6	8	2.39	0	4.64E+05	1.44E+05	6.523603
7	3	1.7	0	3.10E+05	9.70E+04	14.80842
8	2	0	1	1.06E+05	3.51E+04	13.41777
9	1	0	0	7.08E+04	2.39E+04	15.72957
10	5	1.11	1	2.93E+05	1.04E+05	12.59192
11	3	0.006	0	4.04E+05	1.32E+05	15.59124
12	4	0.88	1	2.96E+05	1.03E+05	13.4974
13	12	1.53	1	4.04E+05	1.26E+05	17.41597
14	11	1.43	0	3.94E+05	1.24E+05	15.31773
15	1	0	1	2.31E+05	7.77E+04	15.9429
16	2	0	0	2.65E+05	9.03E+04	18.27887
17	1	0	1	1.79E+05	6.76E+04	23.95663
18	2	0	1	7.97E+04	2.63E+04	32.82435
19	22	0.71	0	4.53E+05	1.54E+05	22.75148
20	8	1.42	0.77	2.95E+05	1.14E+05	24.62209
21	4	1.92	0.8	3.01E+05	9.90E+04	21.58282
22	4	0.94	0	2.67E+05	9.48E+04	34.73219
23	1	0	1	1.10E+05	3.49E+04	42.13635
24	1	0	1	9.14E+04	2.80E+04	40.36455
25	8	0.55	1	2.41E+05	8.30E+04	37.56112
26	2	0	1	1.13E+05	3.77E+04	39.72398
27	2	0	0	1.19E+05	3.82E+04	53.15664

TABLE 6. Problems vs. obtained indexes related to the NSGA-II

Problem	Pareto front metric			Objective function mean values		Computational time
	Number of non-dominated solutions	Spacing metric	QM	Mean z_1	Mean z_2	
1	3	1.5089	1	1.72E+05	6.97E+03	11.99048
2	10	0.75766	0.21622	9.80E+05	1.27E+04	11.09406
3	11	0.88645	0.44	4.25E+06	1.28E+04	13.00945
4	13	0.80206	0.20588	3.90E+06	2.34E+04	12.14553
5	12	1.0162	0.083333	4.89E+06	2.37E+04	13.08017
6	9	0.87584	0.29167	6.70E+06	3.27E+04	11.87017
7	11	0.95535	0.19355	9.09E+06	3.91E+04	11.83453
8	11	0.84226	0.91667	6.78E+06	31495.19	12.17597
9	14	1.1263	0.34286	7.07E+06	46441.94	12.5615

10	10	0.69828	0.34783	9.34E+06	55771.99	12.04777
11	11	1.0172	1	1.11E+07	59469.65	12.2284
12	11	1.1176	1	1.19E+07	59824.82	12.26202
13	10	0.79429	1	1.37E+07	67954.2	12.0311
14	12	0.71965	1	2.00E+07	73020.04	12.36023
15	9	1.364	0.29412	2.52E+07	90003.51	11.13272
16	12	1.8151	0.90909	2.67E+07	103629.3	11.90267
17	12	0.7592	0	2.27E+07	106882.5	11.68452
18	10	0.61745	1	1.89E+07	115649.1	12.14915
19	10	1.2073	0	3.96E+07	112046.6	11.08706
20	9	0.37443	0.28	3.46E+07	126621.2	11.36034

TABLE 7. Problems vs. obtained indexes related to the NRGA

Problem	Pareto front metric			Objective function mean values		Computational time
	Number of non-dominated solutions	Spacing metric	QM	Mean z_1	Mean z_2	
1	6	1.4405	0	17220.22	6971.906	34.33418
2	36	2.2765	0.78378	97960.15	12584.02	20.96036
3	36	0.93509	0.56	415115.9	13117.35	20.67863
4	35	1.4365	0.79412	394099.5	22696.6	21.08452
5	35	1.4198	0.91667	485820.9	23265.65	21.11902
6	34	0.83329	0.70833	678311.7	31882.03	21.18444
7	35	0.95884	0.80645	882258.1	39061.81	21.10709
8	33	1.5107	0.083333	671786.3	32377.67	21.0122
9	35	0.86176	0.65714	680924.6	47891.94	20.27082
10	35	1.0508	0.65217	950517.3	56985.32	20.27982
11	34	1.4061	0	1135262	66045.51	20.00956
12	33	1.1839	0	1231447	64292.26	20.86337
13	36	1.9886	0	1462616	77532.3	19.67027
14	29	2.0254	0	2152912	91681.01	19.55916
15	33	2.5172	0.70588	2604750	89019.46	19.9761
16	34	2.4407	0.090909	2675543	109897.8	19.36939
17	7	2.3533	1	1949359	88772.28	20.74752
18	15	1.9437	0	2098233	115609.2	19.69793
19	3	1.1994	1	3200771	104068.1	20.23629
20	32	2.1377	0.72	2996309	134638.6	18.99066

TABLE 8. Topsis final score decision matrix

	Indexes					
Alternatives	No. of non-dominated solutions	Spacing metric	QM	Mean z_1	Mean z_2	Computational time
NSGA-II	0.35	0.52	0.74	0.72	0.7	0.495
NRGA	0.93	0.86	0.67	0.69	0.72	0.86

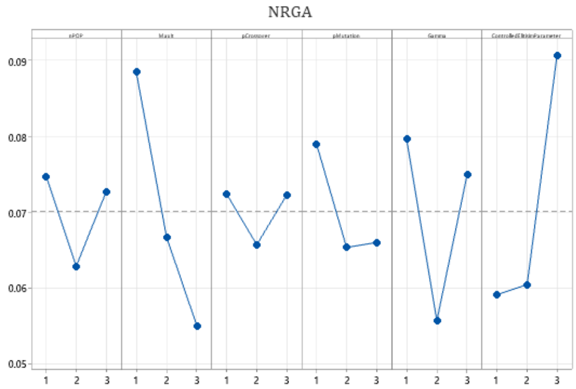


Figure 6. Levels vs. response values related to the NRGA algorithm (less response value is better)

Figure 7 shows the obtained Pareto front by three approaches. Solving problem shows that the best quality is related to the GAMS commercial software (the augmented epsilon-constraint). The NSGA-II yields more Pareto solutions; however, NRGA solutions are closest to the ideal point (origin). As a whole, the conflict between objective functions is strongly evident.

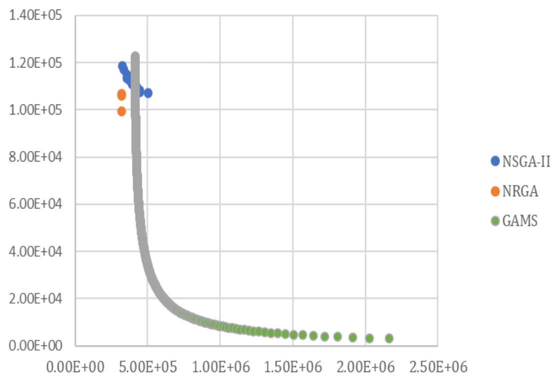


Figure 7. Obtained Pareto front regarding two meta-heuristic algorithms (NSGA-II and NRGA) and GAMS (augmented epsilon-constraint method)

6. CONCLUDING REMARKS AND SUGGESTIONS

This study utilized a partial back ordering, rework, and garbage EPQ model to minimize manufacturing facility costs and warehouse space supply costs. We investigate two separate objective functions regarding operational constraints. Regarding the research literature gap, we contributed our novelty (developing the mathematical model and using solution approaches). First, the best solutions obtained by GAMS software were presented using the augmented epsilon-constraint method. Due to the complexity of the related model, The model was solved using the NSGA-II and NRGA approaches. Various Pareto front indexes were considered, and we tuned the structural parameters of these two algorithms using the Taguchi method. Numerous test problems were investigated and reported all of them meticulously. The obtained Pareto front confirms the conflict between objective functions. Topsis method was used to specify the best approaches. As a whole, GAMS yields the best quality solutions. However, in a large size, the two aforementioned algorithms are justifiable.

Future research should consider using various meta-heuristics, comparing performance metrics, accounting for uncertain parameters, and considering multi-product systems with multiple stages and product limits.

7. REFERENCES

1. Cunha LRA, Delfino APS, dos Reis KA, Leiras A. Economic production quantity (EPQ) model with partial backordering and a discount for imperfect quality batches. International Journal of Production Research. 2018;56(18):6279-93. https://doi.org/10.1080/00207543.2018.1445878
2. Shah NH, Vaghela CR. Imperfect production inventory model for time and effort dependent demand under inflation and maximum reliability. International Journal of Systems Science: Operations & Logistics. 2018;5(1):60-8. https://doi.org/10.1080/23302674.2016.1229076
3. Taleizadeh AA, Soleymanfar VR, Govindan K. Sustainable economic production quantity models for inventory systems with shortage. Journal of cleaner production. 2018;174:1011-20. https://doi.org/10.1016/j.jclepro.2017.10.222
4. Al-Salamah M. Economic production quantity in an imperfect manufacturing process with synchronous and asynchronous flexible rework rates. Operations research perspectives. 2019;6:100103. https://doi.org/10.1016/j.orp.2019.100103
5. Lin SS-C. Note on "The derivation of EOQ/EPQ inventory models with two backorders costs using analytic geometry and algebra". Applied Mathematical Modelling. 2019;73:378-86. https://doi.org/10.1016/j.apm.2019.04.025
6. Marchi B, Zanoni S, Jaber MY. Economic production quantity model with learning in production, quality, reliability and energy efficiency. Computers & Industrial Engineering. 2019;129:502-11. https://doi.org/10.1016/j.cie.2019.02.009
7. Thinakaran N, Jayaprakas J, Elanchezhian C. Survey on inventory model of EOQ & EPQ with partial backorder problems. Materials Today: Proceedings. 2019;16:629-35. https://doi.org/10.1016/j.matpr.2019.05.138
8. Zavanella LE, Marchi B, Zanoni S, Ferretti I. Energy considerations for the economic production quantity and the joint economic lot sizing. Journal of business economics. 2019;89(7):845-65. https://doi.org/10.1007/S11573-019-00933-6
9. De SK, Nayak PK, Khan A, Bhattacharya K, Smarandache F. Solution of an EPQ model for imperfect production process under game and neutrosophic fuzzy approach. Applied Soft Computing. 2020;93:106397. https://doi.org/10.1016/j.asoc.2020.106397
10. Ganesan S, Uthayakumar R. EPQ models for an imperfect manufacturing system considering warm-up production run, shortages during hybrid maintenance period and partial backordering. Advances in Industrial and Manufacturing Engineering. 2020;1:100005. https://doi.org/10.1016/j.aime.2020.100005

11. Guha A, Bose D. A note on "Economic production quantity in batch manufacturing with imperfect quality, imperfect inspection, and destructive and non-destructive acceptance sampling in a two-tier market". Computers & Industrial Engineering. 2020;146:106609. https://doi.org/10.1016/j.cie.2020.106609

12. Hauck Z, Rabta B, Reiner G. Impact of early inspection on the performance of production systems–insights from an EPQ model. Applied Mathematical Modelling. 2022;107:670-87. https://doi.org/10.1016/j.apm.2022.03.003

13. Kalantari SS, Taleizadeh AA. Mathematical modelling for determining the replenishment policy for deteriorating items in an EPQ model with multiple shipments. International Journal of Systems Science: Operations & Logistics. 2020;7(2):164-71. https://doi.org/10.1080/23302674.2018.1542753

14. Nobil AH, Sedigh AHA, Cárdenas-Barrón LE. A multiproduct single machine economic production quantity (EPQ) inventory model with discrete delivery order, joint production policy and budget constraints. Annals of Operations Research. 2020;286:265-301. https://doi.org/10.1007/S10479-017-2650-9

15. Rahaman M, Mondal SP, Shaikh AA, Ahmadian A, Senu N, Salahshour S. Arbitrary-order economic production quantity model with and without deterioration: generalized point of view. Advances in Difference Equations. 2020;2020:1-30. https://doi.org/10.1186/S13662-019-2465-X/FULLTEXT.HTML

16. Rahaman M, Mondal SP, Shaikh AA, Pramanik P, Roy S, Maiti MK, et al. Artificial bee colony optimization-inspired synergetic study of fractional-order economic production quantity model. Soft Computing. 2020;24:15341-59. https://doi.org/10.1007/S00500-020-04867-Y

17. Bose D, Guha A. Economic production lot sizing under imperfect quality, on-line inspection, and inspection errors: Full vs. sampling inspection. Computers & Industrial Engineering. 2021;160:107565. https://doi.org/10.1016/j.cie.2021.107565

18. Kalaiarasi K, Daisy S, Sumathi M. WITHDRAWN: Solving a multi product single machine EPQ inventory model with GP mode: By using python. Elsevier; 2021.

19. Moghdani R, Sana SS, Shahbandarzadeh H. Multi-item fuzzy economic production quantity model with multiple deliveries. Soft Computing. 2020;24(14):10363-87. https://doi.org/10.1007/S00500-019-04539-6

20. Shekhawat S, Rathore H, Sharma K. Economic production quantity model for deteriorating items with weibull deterioration rate over the finite time horizon. International Journal of Applied and Computational Mathematics. 2021;7:1-21. https://doi.org/10.1007/S40819-021-00972-0

21. Kalaiarasi K, Begum MS, Sumathi M. WITHDRAWN: Optimization of unconstrained multi-item (EPQ) model using fuzzy geometric programming with varying fuzzification and defuzzification methods by applying python. Elsevier; 2021.

22. Nobil AH, Niaki STA, Niaki SAA, Cárdenas-Barrón LE. An economic production quantity inventory model for multi-product imperfect production system with setup time/cost function. Revista de la Real Academia de Ciencias Exactas, Físicas y Naturales Serie A Matemáticas. 2022;116:1-23. https://doi.org/10.1007/S13398-021-01182-5

23. Priyan S, Mala P, Palanivel M. A cleaner EPQ inventory model involving synchronous and asynchronous rework process with green technology investment. Cleaner Logistics and Supply Chain. 2022;4:100056. https://doi.org/10.1016/j.clscn.2022.100056

24. Edalatpour MA, Mirzapour Al-e-Hashem SMJ, Fathollahi-Fard AM. Combination of pricing and inventory policies for deteriorating products with sustainability considerations. Environment, Development and Sustainability. 2023:1-41. https://doi.org/10.1007/S10668-023-02988-6

25. Poursoltan L, Seyedhosseini S, Jabbarzadeh A. An extension to the economic production quantity problem with deteriorating products considering random machine breakdown and stochastic repair time. International Journal of Engineering, Transactions B: Applications. 2020;33(8):1567-78. 10.5829/IJE.2020.33.08B.15

26. Deiranlou M, Dehghanian F, Pirayesh MA. The simultaneous effect of holding safety stock and purchasing policies on the economic production quantity model subject to random machine breakdown. International Journal of Engineering, Transactions B: Applications. 2019;32(11):1643-55. https://doi.org/10.5829/ije.2019.32.11b.16

27. Jain M, Sharma G, Rathore S. Economic production quantity models with shortage, price and stock-dependent demand for deteriorating items. International Journal of Engineering, Transactions A: Basics. 2007;20(2):159-68.

28. Pattnaik S, Nayak MM, Abbate S, Centobelli P. Recent trends in sustainable inventory models: A literature review. Sustainability. 2021;13(21):11756. https://doi.org/10.3390/SU132111756

29. Dulebenets MA. A Diffused Memetic Optimizer for reactive berth allocation and scheduling at marine container terminals in response to disruptions. Swarm and Evolutionary Computation. 2023;80:101334. https://doi.org/10.1016/J.SWEVO.2023.101334

30. Dulebenets MA. An Adaptive Polyploid Memetic Algorithm for scheduling trucks at a cross-docking terminal. Information Sciences. 2021;565:390-421. https://doi.org/10.1016/J.AEI.2022.101623

31. Pasha J, Nwodu AL, Fathollahi-Fard AM, Tian G, Li Z, Wang H, et al. Exact and metaheuristic algorithms for the vehicle routing problem with a factory-in-a-box in multi-objective settings. Advanced Engineering Informatics. 2022;52:101623. https://doi.org/10.1145/3583131.3590367

32. Singh E, Pillay N, editors. A study of ant-based pheromone spaces for generation perturbative hyper-heuristics. Proceedings of the Genetic and Evolutionary Computation Conference; 2023.

33. Singh P, Pasha J, Moses R, Sobanjo J, Ozguven EE, Dulebenets MA. Development of exact and heuristic optimization methods for safety improvement projects at level crossings under conflicting objectives. Reliability Engineering & System Safety. 2022;220:108296. https://doi.org/10.1016/J.SWEVO.2023.101314

34. Chen M, Tan Y. SF-FWA: A Self-Adaptive Fast Fireworks Algorithm for effective large-scale optimization. Swarm and Evolutionary Computation. 2023;80:101314. https://doi.org/10.1016/J.COR.2023.106304

35. Fathollahi-Fard AM, Tian G, Ke H, Fu Y, Wong KY. Efficient multi-objective metaheuristic algorithm for sustainable harvest planning problem. Computers & Operations Research. 2023;158:106304.

36. Greco S, Figueira J, Ehrgott M. Multiple criteria decision analysis: Springer; 2016.

37. Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation. 2002;6(2):182-97. 10.1109/4235.996017

38. Al Jadaan O, Rao CR, Rajamani L, editors. Solving constrained multi-objective optimization problems using non-dominated ranked genetic algorithm. 2009 Third Asia International Conference on Modelling & Simulation; 2009: IEEE. 10.1109/AMS.2009.38

39. Al Jadaan O, Rao C, Rajamani L, editors. Parametric study to enhance genetic algorithm performance, using ranked based roulette wheel selection method. International conference on multidisciplinary information sciences and technology (InSciT2006); 2006.

Persian Abstract

چکیده

مدل مقدار تولید اقتصادی (EPQ) نرخ تولید، نرخ تقاضا، هزینه‌های راه‌اندازی، هزینه‌های نگهداری و هزینه‌های کمبود را در نظر می‌گیرد تا مقدار تولیدی را بیابد که مجموع این هزینه‌ها را به حداقل برساند .هدف متعادل کردن هزینه های مرتبط با تولید، نگهداری موجودی و کمبودهای احتمالی است .در این مقاله، دو هدف شامل هزینه های تولید و سفارش مورد نظر است در هدف دوم هزینه فضای ذخیره‌سازی قرار است تا به حداقل برسد .این مطالعه ضایعات و کار مجدد را در مدل EPQ در نظر می‌گیرد .مدل موجودی برای اقلام در یک ماشین مورد نظر است .ظرفیت تولید کاهش می‌یابد و زمانی که فقط یک دستگاه وجود داشته باشد کمبودهایی وجود دارد .با تعیین مقدار محصولات تولید شده توسط واحد تولیدی، فضای ذخیره سازی برای هر محصول، زمان چرخه و کمیاب محصول، می توان هم هزینه کلی و هم هزینه تامین فضای انبار و کاهش داد . برای حل از نرم افزار تجاری و در ابعاد بزرگ، از یک الگوریتم فراابتکاری چندهدفه، یعنی الگوریتم ژنتیک مرتب سازی غیر غالب (NSGA-II) استفاده شده است .یافته‌ها با استفاده از الگوریتم ژنتیک رتبه‌بندی غیرمسلط (NRGA) اعتبار بیشتری دارند .همچنین جبهه پارتو به دست آمده با چند شاخص مورد مطالعه قرار می گیرد .برای اجرای این دو الگوریتم در بهترین شرایط، از رویکرد تاگوچی و آرایه‌های متعامد مربوطه استفاده.شد همچنین برای اعتبارسنجی مدل ریاضی، از روش محدودیت اپسیلون ادغام شده، اجرا شده در محیط GAMS استفاده کردیم .واضح است که نرم افزار تجاری نتایج بهتری به همراه دارد، اما استفاده از این دو الگوریتم زمانی قابل توجیه هستند که ابعاد بزرگتر شود .در نهایت با انجام تحلیل حساسیت این شاخص‌ها و توابع هدف، رفتار الگوریتم‌های پیشنهادی با جزئیات مقایسه و بررسی می‌شود .همچنین الگوریتم برتر با استفاده از روش TOPSIS به عنوان یک روش تصمیم گیری چند معیاره انتخاب شده است .مثال های عددی نشان می دهد که چگونه مدل ارائه شده و الگوریتم های پیشنهادی ممکن است به طور موثر مورد استفاده قرار گیرند .بررسی ادبیات روشن می‌کند که توابع هدف مرتبط، محدودیت‌ها و رویکردهای راه‌حل تا کنون مورد بررسی قرار نگرفته‌اند.