Int. J. Nonlinear Anal. Appl. In Press, 1–26
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2025.37224.5428

Sustainable Supply Chain Model in Plant-Based Milk Production Based on the Internet of Things

Najmeddin Hosseini[a,*], Ahmad Jafarnejad Chaghoshi[b], Nima Garosi Mokhtarzadeh[a]

[a]*Department of Industrial Management, Kish International Campus, University of Tehran, Island, Iran*

[b]*Department of Industrial Management, University of Tehran, Tehran, Iran*

[c] *Department of Technology and Innovation Management, University of Tehran, Tehran*

(Communicated by Majid Eshaghi Gordji)

Abstract

This research has designed a framework for a sustainable supply chain in plant-based milk production based on the Internet of Things. The aim of this research is to present a multi-objective model for the supply chain that uses Internet of Things tools to control and optimize internal network processes. In this system, in order to prevent product spoilage and improve the distribution network, the inventory level in distribution centers is monitored using Internet of Things technology. To solve the complex problems of this supply chain, GAMS software and the hybrid NSGA-II algorithm will be used. The results of the presented research showed that the hybrid algorithm presented (NSGA II algorithm with initial solutions obtained from the MOSA algorithm) has better solutions but takes longer to solve than the NSGA II algorithm. The results of the research have shown that the proposed hybrid algorithm (combining the NSGA-II algorithm with the initial solutions obtained from the MOSA algorithm) has been able to provide better solutions compared to the NSGA-II algorithm alone, but the problem solving time in this hybrid algorithm is longer than that of the NSGA-II algorithm. This increase in solution time is due to the additional processes performed in the hybrid algorithm. Initially, the MOSA algorithm is run independently to generate initial solutions, which requires more time. After that, these initial solutions are given to the NSGA-II algorithm to be improved using evolutionary operations and lead to a set of best solutions. In addition to requiring more time, this process also adds more complexity to the problem. In contrast, the NSGA-II algorithm alone, without the need to generate initial solutions and without this additional process, is able to provide a shorter solution time for the problem. However, the hybrid algorithm using the initial solutions obtained from the MOSA algorithm has been able to achieve more optimal solutions, because these solutions serve as a more optimal starting point for searching the solution space and allow faster access to more optimal regions. This feature improves the quality of the results and proves that the combination of these two algorithms can have a better performance in terms of solution quality, especially in complex and multi-objective problems. As a result, despite the increase in solution time, the hybrid algorithm can be a good choice for problems where the quality of solutions is important, while the NSGA-II algorithm alone may be a better option for problems with time constraints

Keywords: (Sustainable supply chain, plant milk, Internet of things, NSGA II algorithm, MOSA algorithm)
2022 MSC: O20, O10

*Corresponding author
Email addresses: nhoseini@ut.ac.ir (Najmeddin Hosseini), jafarnjd@ut.ac.ir (Ahmad Jafarnejad Chaghoshi), mokhtarzadeh@ut.ac.ir (Nima Garosi Mokhtarzadeh)

Received: March 2025 *Accepted:* April 2022

1 Introduction

With 75% of the world's adult population being lactose intolerant, it is not surprising that plant-based milk is the best-selling product in the entire plant-based segment. But it is not just about lactose intolerance. People prefer plant-based milk to cow's milk for a variety of reasons. This could be due to their nutritional value, animal welfare, lower environmental impact, prevention of dairy allergies, or simply personal preference [7]. This highlights the need to address issues related to plant-based milk production. In this study, a plant-based milk production supply chain is examined. In this system, in order to prevent corruption and improve the distribution network, inventory levels are monitored at distribution centers using the Internet of Things, and if a certain amount of inventory is reached, then orders are placed. The ordering is done by the Internet of Things technology and is determined based on the transmitted information of the intelligent routing in the distribution and logistics network in that period and based on the same information, the production planning and receipt of raw materials and packaging items from each supplier is carried out taking into account sustainability criteria. To solve the problem, GAMS software and the NSGAII (Non-dominated Sorting Genetic Algorithm) algorithm will be used. In fact, in this study, the development of a plant milk production supply chain network has been built using optimization modeling tools.

The Internet of Things, through the combination of sensors, cloud technologies and big data analysis, enables real-time monitoring of production processes [4]. An IoT-equipped factory can collect real-time data from devices and machinery and send it to intelligent decision-making systems. Some of the main benefits of these systems include: Improved quality control: Using sensors, data related to the quality of raw materials and finished products is monitored. If the product quality falls outside the standard range, the system can issue an alert and automatically adjust the process [5]. Reduced production downtime: IoT systems can predict equipment failures and perform preventive maintenance measures by analyzing sensor data. This leads to increased efficiency of production processes and reduced repair costs [6]. Supply chain optimization: Using IoT data, manufacturers can accurately predict demand and adjust production according to market needs. This prevents overproduction or underproduction [11].

IoT provides AI systems with the necessary information by collecting real-time data from different stages of the supply chain. Some important IoT applications in this field include: Management of raw material supply farms: Smart sensors can record information on water consumption, plant health, and soil nutrient levels and send it to AI algorithms. These systems provide farmers with detailed recommendations on optimizing water and nutrient use, which helps increase productivity and reduce environmental impact [1]. Monitoring transportation and distribution: GPS systems and temperature and humidity sensors in vehicles transporting plant milk provide information on the storage conditions of the product along the route. AI can analyze this data and suggest corrective actions in case of risk of spoilage or adverse changes in transportation conditions [2]. Optimizing energy consumption in factories: AI systems can monitor energy consumption in plant-based milk factories and provide optimization solutions. This technology will reduce operating costs and reduce greenhouse gas emissions.

Artificial intelligence, along with the Internet of Things, can fundamentally transform the sustainable plant-based milk supply chain. These technologies contribute to greater sustainability in this industry by optimizing resource consumption, reducing waste, increasing quality, and reducing costs. Demand forecasting, intelligent quality control, optimal logistics management, and increasing agricultural productivity are among the most important benefits of combining AI and IoT in the plant-based milk supply chain.

Based on the reviews and studies conducted on previous research, some research gaps and suggestions for further research were extracted in these articles, and the relevant explanations are provided below:

Table 1: Research background and unanswered challenges

Article title, authors' names and journal	Research Gaps Mentioned
Applications of Internet of Things in the Food Supply Chain: a Literature Review Reza Tavakkoli-Moghaddam. et al. [13] Journal of Applied Research on Industrial Engineering	According to the summary of statistics provided, it is observed that most of the articles published in reputable journals have a Q1 impact factor and the highest number of citations is 806 articles. Elsevier publishes the most IoT articles in FSC with topics such as food control, simulation, blockchain and food transportation. By studying the articles, it is observed that less than 12% of the articles in the field of mathematical modeling, optimization and simulation of IoT in FSC are done, and the rest of them present operational strategies and impact of IoT This review aims to fill this knowledge gap by systematically analyzing the existing literature on the application of blockchain.
Assessing blockchain and IoT technologies for agricultural food supply chains in Africa: A feasibility analysis Andrews Tang. et al. [12] Review Article in Science Direct	and IoT in the food supply chain assesses their technical, economic, legal and operational feasibility and provides valuable insights to the stakeholders and suggests directions for future research and implementation strategies. Although many studies may be mentioned in RSC and waste management, limited works discuss the design of CPWs supply chains. Above and beyond all other considerations, the literature review revealed that the number of studies that have developed a mathematical model for CPWs is small with many shortcomings.
The IoT-enabled sustainable reverse supply chain for COVID-19 Pandemic Wastes (CPW) Behzad Mosallanezhad. et al. [10] Engineering Applications of Artificial Intelligence (2023)	We examined the role of digital technologies in solving some of the emerging problems surrounding urban food security and reviewed the
How Important are Digital Technologies for Urban Food Security? A Framework for Supply Chain Integration using IoT Soujanya Mantravadia, Jagjit Singh Srai [8] 4th International Conference on Industry 4.0 and Smart Manufacturing	Research Gaps Mentioned

In this research, various innovations have been proposed: One of the important issues that has been raised in recent years is the improvement of meta-heuristic algorithm solutions. Given that such algorithms do not have exact solutions, therefore, in this research, an attempt has been made to improve the NSGAII algorithm.

In previous research in the field of plant milk, no research has been done on production planning, order planning, and distribution network; especially in this research, in addition to production, supplier, and distribution planning, the discussion of the Internet of Things in the plant milk supply chain has been raised.

In this system, a supply chain distribution network has been designed, based on which, by means of the Internet of Things, when the distributor's inventory level reaches and the warning system is sent, the routing and logistics network are updated, and production and supply planning is carried out. This issue has not been seen in previous research either.

A multi-objective SC network is presented in the current work that simultaneously optimizes the total network cost, transportation risk, environmental impact, CO2 emission, and job opportunity. Also, the IoT platform empowers the model to obtain the required information as model input and is constrained by sustainability aspects to provide a nature-friendly SC network.

2 Research method

In this study, the data collection method was carried out in a library manner. Also, in order to provide information, random data is used to solve random problems, and for data analysis, GAMS software was used in small dimensions and metaheuristic algorithms in MATLAB software were used in large dimensions. Finally, sensitivity analysis was also performed and the efficiency of the presented algorithm was examined in comparison with other methods.

The research steps in this study are as follows:

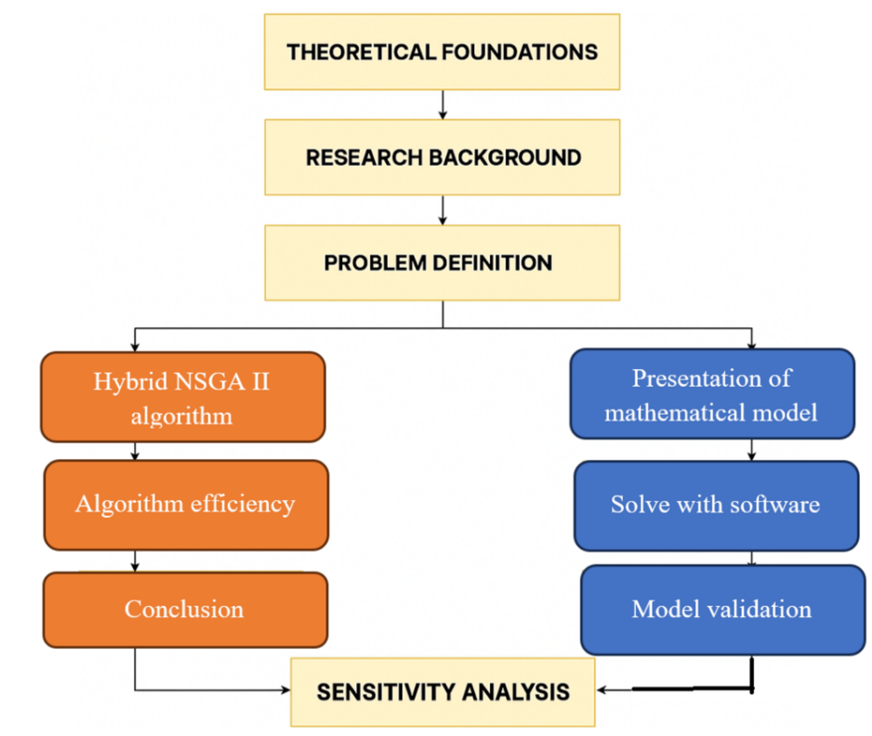

Figure 1: Problem Schematic

2.1 Mathematical model

The mathematical model of the problem is defined as follows.

Parameters:

- Fix_d: Fixed cost of distributor d (in Rials)
- $cost_k$: Rental cost of vehicle k (in Rials)
- A_{ijk}: Binary parameter; equal to 1 if vehicle k is allowed to travel from node i to node j, otherwise 0.
- dx_{ij}: Distance from node i to node j (in meters)
- cst_k: Travel cost of vehicle per unit distance and per unit weight (in Rials)
- cp_j: Capacity of the smart bin at customer location j (in liters)
- cap_k: Capacity of vehicle k (in liters)
- cpd_d: Capacity of collection center d (in liters)
- d_{jd}: Consumption amount of location j in period t (in liters)
- $scap_s$: Capacity of supplier s (in liters)

- α: Raw material consumption coefficient
- $Fcas_f$: Production amount during regular hours of manufacturer f (in liters)
- Re_s: Ordering cost from supplier s (in Rials)
- iv_f: Inventory holding cost per period for manufacturer f (in Rials)
- TR_{df}: Cost of sending one unit of product from manufacturer f to distributor d (in Rials)
- Tx_{fs}: Cost of sending one unit of raw material from supplier s to manufacturer f (in Rials)

Decision Variables:

- E_{fst}: Binary variable; equal to 1 if manufacturer f orders from supplier s in period t, otherwise 0.
- R_{fst}: Order quantity of manufacturer f from supplier s in period t (in liters)
- ie_{ft}: End-of-period inventory of manufacturer f in period t (in liters)
- is_{ft}: Beginning-of-period inventory of manufacturer f in period t (in liters)
- Y_{dft}: Quantity of demand sent from manufacturer f to distributor d in period t (in liters)
- Op_{ft}: Production amount of manufacturer f in period t after working hours (in liters)
- X_{ijkd}: Binary variable; equal to 1 if vehicle k, belonging to distributor d in period t, travels from node i to node j, otherwise 0.
- hst_{jt}: Unfulfilled service time for customer j in period t (in minutes)
- F_d: Binary variable; equal to 1 if distributor d is used, otherwise 0.
- θ_j: Cost of excess load transportation on the vehicle (in Rials)
- L_{jt}: Load amount on the vehicle in period t after leaving node j (in liters)
- de_{jt}: Product amount in the smart bin at customer location j in period t (in liters)
- Lo_{kdt}: Amount of product that distributor d must supply in period t (in liters)
- jEt_{jt}: Delivery time of goods to customer j in period t

Objective function :

$$\begin{aligned}\min Z_1 =& \sum_{d\in D} F_d \times Fix_d + \sum_{d\in D}\sum_{k\in K}\sum_{j\in c}\sum_{i\in C} X_{ijdk}\times cost_k + \sum_{i\in w}\sum_{k\in K}\sum_{j\in w}\sum_{d\in D}\sum_{t\in T} X_{ijdkt}\times ct_{ijk} + \sum_{j\in c}\sum_{t\in T}\theta_{jt}\\ &+ \sum_{f\in F}\sum_{s\in S}\sum_{t\in T} E_{fst}\times RE_s + \sum_{f\in F}\sum_{t\in T} ie_{ft}\times iv_f + \sum_{f\in F}\sum_{d\in D}\sum_{t\in T} Y_{fdt}\times tR_{df} + \sum_{f\in F}\sum_{s\in S}\sum_{t\in T} R_{fst}\times tx_{fs}\\ \min Z_2 =& \sum_{i\in w}\sum_{k\in K}\sum_{j\in w}\sum_{d\in D}\sum_{t\in T} X_{ijKdt}\times Q_{ijk}\end{aligned}$$

In the first objective function, the distributor's fixed cost, the vehicle's fixed cost, the transportation cost between the distributor and customers, the cost of additional cargo on the vehicle, the cost of purchasing from the supplier, the cost of ending inventory, the ordering cost, and the transportation cost between the distributor, supplier, and manufacturer are minimized.

In the second objective function, the amount of greenhouse gas emissions is minimized.

Constraints:

1-	$\sum_{k\in K}\sum_{i\in w}\sum_{d\in D} x_{ijkdt} = B_{jt}$	$\forall j \in c, t \in T$
2-	$B_{jt} \times M \geq de_{jt} - cp_j$	$\forall j \in c, t \in T$
3-	$de_{jt} \geq d_{jt}$	$\forall j \in c, t \in T$
4-	$de_{jt} \geq de_{j(t-1)} + d_{jt} - MB_{jt}$	$\forall j \in c, t \in T$
5-	$\sum_{i\in w} x_{ijkdt} = \sum_{i\in w} x_{jikdt}$	$\forall j \in c, t \in T, k \in K, d \in D$
6-	$\sum_{j\in c} x_{djkdt} = \sum_{j\in c} x_{jdkdt}$	$\forall t \in T, k \in K, d \in D$
7-	$LO_{kdt} = \sum_{j\in c}\sum_{i\in w} x_{ijkdt} \times de_{jt}$	$\forall t \in T, k \in K, d \in D$
8-	$LO_{kdt} \leq cap_k$	$\forall t \in T, k \in K, d \in D$
9-	$dc_{dt} = \sum_{k\in K} Lo_{kdt}$	$\forall t \in T, d \in D$
10	$dc_{dt} \leq cap_d$	$\forall t \in T, d \in D$
11-	$Et_{jt} \geq Et_{it} + st_{it} + \frac{dx_{ij}}{v_k} - M(1 - x_{jikdt})$	$\forall t \in T, k \in K, d \in D, i \in w, j \in c$
12-	$Et_{jt} \leq timea$	$\forall t \in T, j \in c$
13-	$hst_{jt} \geq Et_{jt} - timeb$	$\forall t \in T, j \in c$
14-	$L_{jt} \geq Lo_{kdt} - de_{jt} - M(1 - x_{jdkdt})$	$\forall t \in T, k \in K, d \in D, j \in c$
15-	$L_{jt} \geq L_{it} - de_{jt} - M(1 - x_{jikdt})$	$\forall t \in T, k \in K, d \in D, i \in w, j \in c$
16-	$x_{ijkdt} \leq A_{ijk}$	$\forall t \in T, k \in K, d \in D, i, j \in w$
17-	$\theta_{jt} \geq Lo_{kdt} \times dx_{dj} \times cst_k - M(1 - x_{jdkdt})$	
18-	$\theta_{jt} \geq L_{it} \times dx_{ij} \times cst_k - M(1 - x_{jikdt})$	
19-	$F_d \times M \geq x_{djkdt}$	$\forall t \in T, k \in K, d \in D, j \in c$
20-	$\sum_{f\in F} Y_{df} = de_{dt}$	$\forall t \in T, d \in D$
21-	$\sum_{d\in D} Y_{df} \leq Fcap_f$	$\forall t \in T, f \in F$
22-	$is_{ft} = in$	$\forall t = 1, f = F$
23-	$ie_{(f-1)} = is_{ft}$	
24-	$ie_{ft} = is_{ft} - \sum_{f\in F} Y_{df} + \sum_{s\in S} R_{rst} \times \alpha$	$\forall t \in T, f \in F$
25-	$ie_{ft} \geq 0$	$\forall t \in T, f \in F$
26-	$op_{pt} \geq \sum_{d\in D} Y_{df} - Fcas_f$	$\forall t \in T, f \in F, d \in D$
27-	$Es_{fst} \times M \geq R_{rst}$	$\forall t \in T, f \in F, s \in S$
28-	$\sum_{f\in F} R_{rst} \leq Scap_s$	$\forall t \in T, s \in S$

- Constraint one, if the customer demand level is met, a vehicle is assigned to service that location.
- Constraint two, determines which smart container the customer is projected to need to service during that time period.
- Constraints three and four, determine the number of products in the smart containers at each customer location.
- Constraint five, each vehicle that enters a location for service, the same vehicle leaves that location.
- Constraint six, in each time period, each vehicle that leaves a collection center, the same vehicle enters the same collection center.
- Constraint seven, determines the maximum amount of cargo that can be carried on a vehicle along the route.
- Constraint eight, determines the vehicle capacity constraint.
- Constraint nine, calculates the amount of demand that each distributor must send to customers.
- Constraint ten ensures that the distributors' capacity is met.
- Constraint eleven determines the time of delivery of the goods to the customers.
- Constraint twelve ensures that the strict time constraint is met.
- Constraint thirteen determines the amount of time constraint non-compliance.
- Constraints fourteen and fifteen determine the amount of load on the vehicle along the route.
- Constraint sixteen determines if a vehicle passes a route that is permitted or capable of passing that route.
- Constraints seventeen and eighteen calculate the additional cost due to the amount of load on the vehicle.
- Constraint nineteen determines which of the distributor centers is used.
- Constraint twenty determines the total demand of the distributors in each time period that is met by the manufacturer.
- Constraint twenty-one determines the capacity of the manufacturers in each time period.
- Constraint twenty-two determines the first period inventory.
- Constraint twenty-three, the inventory at the beginning of each period is equal to the inventory at the end of the previous period.
- Constraint twenty-four, determines the inventory at the end of each period.
- Constraint twenty-five, the inventory at the end of each period cannot be a negative number.
- Constraint twenty-six, determines the amount of production during overtime.
- Constraint twenty-seven, determines the ordering of suppliers.
- Constraint twenty-eight, determines the capacity of suppliers.

The presented mathematical model is a complex optimization problem in the field of supply chain management, which includes two objective functions and a set of constraints for demand management, distribution, production, and transportation. Further explanations are provided along with the complete equations.

1. **First objective function**

The first objective function minimizes the total system costs. These costs include the following:

Equation:

$$\min Z_1 = \sum_{d\in D} F_d \times Fix_d + \sum_{d\in D}\sum_{k\in K}\sum_{j\in C}\sum_{i\in C} Xijdk \times cost_k + \sum_{i\in W}\sum_{k\in K}\sum_{j\in W}\sum_{d\in D}\sum_{t\in T} Xijkdt \times ct_{ijk} +$$
$$\sum_{j\in C}\sum_{t\in T}\theta_{jt} + \sum_{f\in F}\sum_{s\in S}\sum_{t\in T} Efst \times RE_s + \sum_{f\in F}\sum_{t\in T} ie_{ft} \times iv_f + \sum_{f\in F}\sum_{d\in D}\sum_{t\in T} Yfdt \times TR_{df} +$$
$$\sum_{f\in F}\sum_{s\in S}\sum_{t\in T} Rfst \times tx_{fs}$$

Detailed Description:

1. Distributor Fixed Cost:

$$\sum_{d\in D} Fd \times Fix_d$$

   This section shows the fixed cost of using distributors.

2. Fixed cost of transportation:

$$\sum_{d\in D}\sum_{k\in K}\sum_{j\in C}\sum_{i\in C} Xijdk \times cost_k$$

   This section calculates the fixed cost of using vehicles.

3. Transportation cost between distributor and customers:

$$\sum_{i\in W}\sum_{k\in K}\sum_{j\in W}\sum_{d\in D}\sum_{t\in T} Xijkdt \times ct_{ijk}$$

   This section shows the cost of moving vehicles between different points.

4. Cost of additional cargo on the vehicle:

$$\sum_{j\in C}\sum_{t\in T}\theta_{jt}$$

   This section calculates the cost of additional cargo on the vehicle.

5. Purchase cost from supplier:

$$\sum_{f\in F}\sum_{s\in S}\sum_{t\in T} Efst \times RE_s$$

   This section shows the cost of ordering from suppliers.

6. Ending inventory cost:

$$\sum_{f\in F}\sum_{t\in T} ie_{ft} \times iv_f$$

   This section calculates the cost of holding inventory at the end of each period.

7. Ordering and moving costs between the distributor, supplier, and manufacturer:

$$\sum_{f\in F}\sum_{d\in D}\sum_{t\in T} Yfdt \times TR_{df} + \sum_{f\in F}\sum_{s\in S}\sum_{t\in T} Rfst \times tx_{fs}$$

This section includes costs related to shipping raw materials and products between different locations.

Second objective function

The second objective function minimizes greenhouse gas emissions.

Equation:

$$\min Z_2 = \sum_{i\in W}\sum_{k\in K}\sum_{j\in W}\sum_{d\in D}\sum_{t\in T} Xijkdt \times Qijk$$

Detailed description:

- This objective function calculates the amount of greenhouse gas emissions caused by the movement of vehicles between different points.
- The parameter Q_{ijk} is the amount of greenhouse gas emissions per unit distance of movement between nodes i and j by vehicle k.

Constraints

Constraint 1: Vehicle Assignment to Customers

$$\forall j \in C,\ t \in T:\ \sum_{k\in K}\sum_{i\in W}\sum_{d\in D} Xijkdt = Bjt$$

- This constraint ensures that if customer j needs service in period t ($B_{jt} = 1$), at least one vehicle is assigned to it.

Constraint 2: Determining the projected smart containers

$$\forall j \in C,\ t \in T:\ Bjt \times M \geq de_{jt}cp_j$$

- This constraint determines which smart container needs to be emptied (its capacity is full).

Constraints 3 and 4: Smart Container Inventory

$$\forall j \in C,\ t \in T:\ de_{jt} \geq djt \quad \forall j \in C,\ t \in T:\ de_{jt} \geq de_j(t1) + djtM \times Bjt$$

- These constraints determine the number of products in the smart containers at any given time period.

Constraint 5: Vehicle Entry and Exit

$$\forall j \in C,\ t \in T,\ k \in K,\ d \in D:\ \sum_{i\in W} Xijkdt = \sum_{i\in W} Xjikdt$$

- This constraint ensures that every vehicle that arrives at a node also leaves that node.

Constraint 6: Return to the collection center

$$\forall t \in T,\ k \in K,\ d \in D:\ \sum_{j\in C} Xdjkdt = \sum_{j\in C} Xjdkdt$$

- This constraint ensures that vehicles return to their collection center after completing the route.

Constraint 7: Amount of cargo on the vehicle

$$\forall t \in T,\ k \in K,\ d \in D:\ LO_{kdt} = \sum_{j\in C}\sum_{i\in W} Xijkdt \times de_{jt}$$

- This constraint calculates the amount of cargo on the vehicle along the route.

Constraint 8: Respecting the vehicle capacity

$$\forall t \in T,\ k \in K,\ d \in D:\ LO_{kdt} \leq cap_k$$

- This constraint ensures that the vehicle load does not exceed its capacity.

Constraints 9 and 10: Distributor Demand

$$\forall t \in T,\ d \in D:\ dc_{dt} = \sum_{k\in K} LO_{kdt} \quad t \in T,\ d \in D:\ dc_{dt} \leq cap_d$$

- These constraints determine the demand that each distributor must meet and their capacity.

Constraint 11: Lead Time

$$\forall t \in T,\ k \in K,\ d \in D,\ i, j \in W:\ Et_{jt} \geq Et_{it} + st_{it} + vkdx_{ij}M \times (1Xjikdt)$$

- This constraint determines the time of delivery of goods to customers.

Constraints 12 and 13: Respecting the time constraint

$$\forall t \in T,\ j \in C:\ Et_{jt} \leq timea\, \forall t \in T,\ j \in C:\ hst_{jt} \geq Et_{jt}timeb$$

- These constraints ensure that the delivery time of the goods is within the time constraints.

Constraints 14 and 15: Load on the vehicle

$$\forall t \in T,\ k \in K,\ d \in D,\ j \in C:\ Ljt \geq LO_{kdt}de_{jt}M\times(1Xjdkdt)\forall t \in T,\ k \in K,\ d \in D,\ i,j \in W:\ Ljt \geq Litde_{jt}M\times(1Xjik$$

- These constraints determine the amount of load on the vehicle along the route.

Constraint 16: Permissible Vehicle Route

$$\forall t \in T, k \in K,\ d \in D,\ i,j \in W:\ Xijkdt \leq Aijk$$

- This restriction ensures that the vehicle only travels on authorized routes.

Restrictions 17 and 18: Excess Cargo Fee

$$\theta jt \geq LO_{kdt} \times dx_{dj} \times cst_k M \times (1Xjdkdt)\theta jt \geq Lit \times dx_{ij} \times cst_k M \times (1Xjikdt)$$

- These restrictions calculate the additional cost of additional load on the vehicle

Restriction 19: Use of a distributor

$$\forall t \in T,\ k \in K,\ d \in D,\ j \in C:\ Fd \times M \geq Xdjkdt$$

- This constraint determines which distributors are used.

Constraints 20 and 21: Distributor demand and producer capacity

$$\forall t \in T,\ d \in D:\ \sum_{f \in F} Yfdt = de_{dt} \quad \forall t \in T,\ f \in F:\ \sum_{d \in D} Yfdt \leq Fcap_f$$

Constraints 22 to 28 determine distributor demand and producer capacity.

3 Findings

3.1 Examining the complexity of the problem

In the present study, the complexity of the problem refers to the degree of difficulty in solving it in different dimensions. This complexity is determined by increasing the number of variables and input dimensions of the problem. The results show that as the dimensions of the problem increase, the time required to solve it increases exponentially. In other words, by increasing the number of variables or increasing the complexity of the inputs, the time required to solve the problem becomes significantly longer.

This increase in the solution time indicates the high complexity of the problem, so that in larger dimensions, its exact solution is not possible using common software such as GAMS. Therefore, in the present study, it can be concluded that the problem is highly complex due to these features and there is a need for optimization methods and alternative algorithms to solve it in larger dimensions. In this section, the complexity of the problem is first examined by analyzing its solution time. In order to perform this analysis, a set of examples with different problem dimensions were randomly selected and their solution time was measured using the GAMS software. The results show that as the problem dimensions increase, the time required to solve it increases exponentially. This rapid increase in the solution time in the GAMS software indicates the high complexity of the problem. After reaching a certain dimension, an exact solution of the problem will not be possible even using this software, because there are not enough computational resources to solve it.

Table 2: Dimensions of the examples presented

Example Number	Number of Manufacturers	Number of Distributors	Number of Suppliers	Number of Customer Locations
1	2	3	2	6
2	2	3	2	7
3	2	3	2	8
4	3	3	2	9
5	3	4	2	10
6	3	4	3	11
7	4	4	3	12
8	4	4	3	13
9	4	5	3	14
10	4	5	3	15

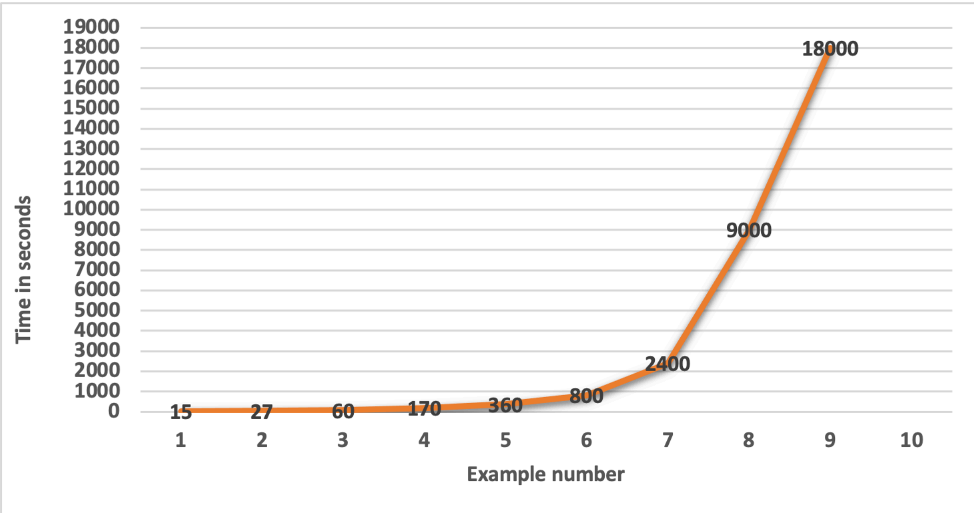

Figure 2: Study of the sharp increase in problem solving time with GAMS software

3.2 Multi-objective problem solving method using GAMS

Since the aforementioned model is multi-objective, the LP metric method has been used to solve the aforementioned multi-objective model in such a way that each of the objective functions is optimized each time and finally the optimal value of the objective functions is calculated by considering the constraints of the problem using the L-P metric method. In this method, deviations from the optimal value are calculated for each of the objective functions and the goal of optimization is to minimize the sum of these deviations.

The L-P metric method uses the combination of several different objective functions in a multi-objective model without the need to assign weights and solves the problem objectively and automatically and ensures that the results are free from personal bias and reliable [3]. Using this method in the present study allows the sustainable plant milk supply chain model to be optimized by considering different economic and environmental objectives. With the help of this method, the objectives are optimized independently of each other and then the final results are evaluated simultaneously in terms of the optimality of different objective functions. This process will lead to an optimal solution to the sustainable supply chain design problem.

3.3 Components of the Hybrid NSGA II Algorithm

The NSGA II algorithm in question has the following components [9]:

Initialization:

At the beginning of the problem-solving process, parameters related to the meta-heuristic algorithm are entered into the software. These parameters include things such as the mutation rate, the initial population size, and the number of algorithm iterations. These parameters play a very important role in the performance of the algorithm and model optimization.

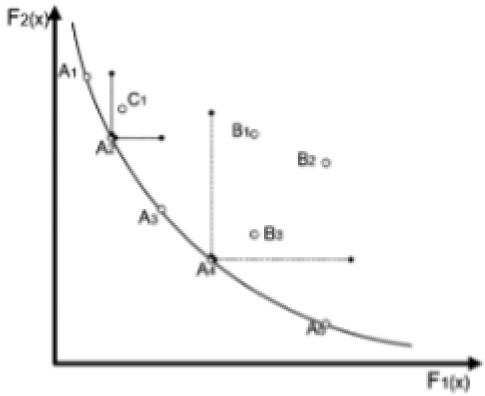

Figure 3: Comparison of optimal and non-optimal solutions

The mutation rate, which is commonly used in evolutionary and metaheuristic algorithms such as genetic algorithms, refers to the amount of change in the solution search process per generation. This parameter has a great impact on the algorithm's ability to explore the search space and can improve population diversity

The initial population size also determines how many initial solutions the algorithm starts searching with. The larger the population size, the greater the possibility of finding more optimal solutions in the search space, but on the other hand, the computational time also increases

Data entry

In the next stage of the research, data and parameters related to the sustainable supply chain design model for plant-based milk production were collected accurately and systematically and organized in an Excel file. This data includes various information such as costs, demands, capacities, resource constraints, and various environmental and economic parameters that are required for accurate modeling and optimization analysis. Then, this data will be transferred to the relevant software to obtain an optimal solution for sustainable supply chain design and management using the algorithms and problem-solving methods available in the software.

Generating the initial solution (chromosome structure)

In this part, the initial solution display has 4 parts. In the first part, the initial solution display is a chromosome with $n + k - 1$ (n is the number of demand centers or customers and k is the number of vehicles) houses. In the first chromosome, the numbers one to n represent customers and the numbers above n are the numbers of vehicles. Therefore, the number $n + i$ represents the i-th vehicle. In this case, the numbers before the number $n + i$ are the numbers of customers assigned to that vehicle and served in order. Also, the numbers after the last number greater than n, then those customers are assigned to the last vehicle.

If there is no number before the vehicle number, then that vehicle is not used. If there is no demand for that customer in any time period, then that customer number is not considered

In the second chromosome, there are $k+d-1$ cells (k is the number of vehicles and d is the number of distributors), then the numbers one to k represent the vehicle number and the number $k + i$ represents the ith distributor. In this part, the number before $k + i$ represents the vehicles that are assigned to the ith distributor.

In the third chromosome, there are $d + F - 1$ cells. (d is the number of distributors and d is the number of manufacturers), then the numbers one to k represent the vehicle number and the number $d + i$ represents the ith manufacturer. In this part, the number before $d+i$ represents the distributors that are assigned to the ith manufacturer.

Finally, for the last part of the chromosomes, the percentage of demand supplied by each supplier is determined for each producer. This representation of the initial answer is repeated for each time period.

Initial population generation

The first step in the genetic algorithm is the generation of the initial population. The way the initial population is created in this research is random and the initial population is created by generating random numbers. Each initial answer consists of two parts and to produce each answer, the previous part is created as an algorithm and both parts are placed in a row. This is done to the number of initial populations.

Fitness function

In this problem, the evaluation function calculates the value of both types of objective functions and examines the constraints of the problem. The first evaluation function calculates the maximum service time to the damaged locations and the second evaluation function calculates the system costs. If the desired solution does not meet the desired constraints, then the calculated objective function value is multiplied by a constant number in order to not select that solution. Determining the ranking of solutions based on non-dominated solutions of the crowding distance. The solutions in question are ranked according to how many solutions will definitely be better than that solution. Solutions will definitely be better than another solution if all the values of the objective functions are better than the desired solution. For example, solutions for which no better solution can be found are ranked first and solutions for which there is only one better solution are ranked second. The rest of the rankings are also divided on this basis. Therefore, the solutions are divided into different degrees according to the quality of the objective functions. The algorithm of this method is as follows: first, the initial population is randomly selected. Then, this population is assigned a separate rank to the n beam levels according to the objective functions and based on the concept of dominance, and an equal rank is assigned to the members within the level based on the concept of dominance. Now, for the members that are located in a beam level and have a rank, a distance number is assigned according to the population distance method, which expresses the distance rank (swarming distance).

The swarm sorting operation for the points of the last front that could not be completely joined to the new population is performed by sorting the swarming distance values in descending order and then selecting the points at the top of the sorted list. The swarming distance of point i, denoted by di, is a measure of the target space around point i that is not occupied by any other solution in the population. Here, there is a simple method for calculating the quantity di and it is done by estimating the space around the cube that is made by the nearest neighbors in the target space that are located at the vertices of the cube. This quantity is called the crowding distance. The method of calculating the crowding distance in the NSGAII algorithm is that after ranking the answers and dividing them into several non-dominated answer sets with a specific rank, in each of these sets it calculates the distance of the desired answer from its nearest neighboring answers. In fact, the crowding distance will be equal to:

$$d_i = cd_1(x) + cd_2(x)$$

Using this value, if two answers have the same rank in the selection stage of the genetic algorithm, the solution with the greater crowding distance will be more desirable. Parental selection After the fitness value is calculated for each chromosome in the population, the algorithm randomly selects two chromosomes and selects whichever one has the better fitness value as the parent. To select the more appropriate answer and the better fitness value between the two randomly selected members, the answer with the better rank and lower front is selected. If their fronts are equal to each other, the answer with the greater crowding distance will be selected. The answer with the lower front is selected as the parent. If two answers are on the same front, the answer with the greater crowding distance is selected. Elitism. In each series to produce the next population to produce the next N populations, M answers are selected by selecting the best answers from the previous population. The N-M value of the answer will also be done by the intersection operation.

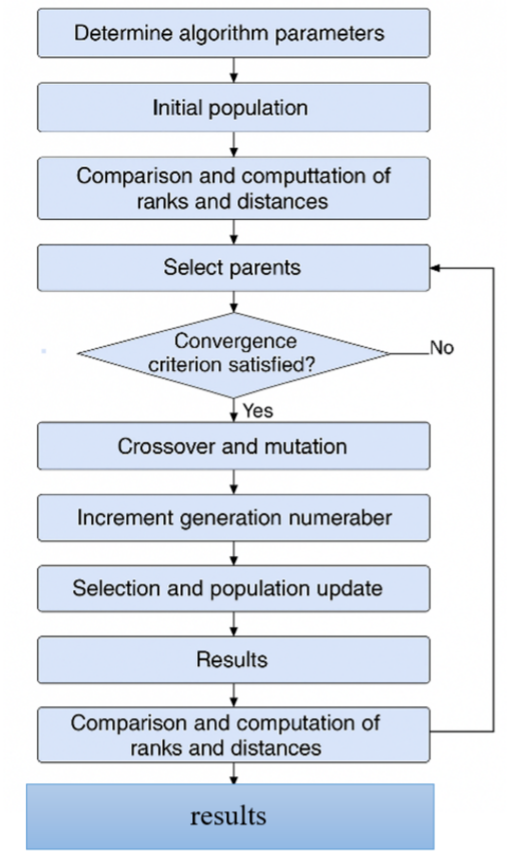

Figure 4: Designed NSGA II algorithm

In this study, in order to present the meta-heuristic algorithm, the initial solutions were obtained by the refrigeration simulation algorithm.

3.4 Determining the algorithm parameters by Taguchi

Since the output of the problems is strongly dependent on the parameters of the proposed algorithms, we use the Taguchi method to adjust their parameters. The advantage of the Taguchi method over other experimental design methods, in addition to the cost, is to obtain the optimal levels of parameters in less time. One of the most important steps of this method is to select an orthogonal array that estimates the effects of factors on the average response and changes. In this study, the most appropriate design is identified as three-level experiments, and considering the standard Taguchi orthogonal arrays, the L_9 array has been selected as the appropriate experimental design for adjusting the parameters of the proposed algorithms. In order to adjust the optimal parameters, Taguchi considers a statistical measure of performance called the S.N ratio, which includes the average and changes, and the higher the level, the more desirable it is. The response variable considered is the ratio of two standard indices MID to Diversity for multi-objective algorithms. Since this response variable is of the "less is better" type, its corresponding S.N. ratio is considered as the following relationship. The proposed meta-heuristic algorithms are run for each Taguchi experiment and then the S.N. ratios are calculated by Minitab 14.1 software [14].

$$\frac{S}{N}\ ratio = -10\log\left(\frac{sumy^2}{n}\right)$$

According to the results obtained, the mutation rate value of 0.05 and the initial population size of 100 are the best values for the mutation rate.

Figure 5: Result of parameter adjustment

3.5 Examining the efficiency of the algorithm in small dimensions

In the previous section, it was proven that the problem presented is complex and that a hybrid algorithm is needed to solve the problem in large dimensions. At this stage, the efficiency of the algorithm in small dimensions is examined.

In order to examine the correctness, accuracy and efficiency of the algorithm, five numerical examples in small dimensions were randomly generated and solved by the hybrid algorithm. The answers obtained from the GAMS software are presented in the table below and then compared.

Considering the small difference between the average answers of the hybrid algorithm compared to the answers obtained from the GAMS software, it is proven that the hybrid algorithm presented in small dimensions has a suitable efficiency. The value of this deviation is examined in the table below and the maximum deviation is 2.88 percent, which is an acceptable value.

Table 3: Examining the efficiency of the hybrid algorithm in small dimensions

Example number	Number of sets					Objective function: Cost			Objective function: Greenhouse gas emissions		
	Distributors	Manufacturers	Customers	Suppliers	Time period	Answer: Gams	Hybrid algorithm	Deviation	Answer: Gams	Hybrid algorithm	Deviation
1	3	2	5	2	3	190.5	191	0.26%	40.7	41	0.74%
2	4	2	6	2	3	216.4	218.9	1.16%	45.6	46.1	1.10%
3	5	3	8	2	4	289.7	295.6	2.04%	59.8	61	2.01%
4	5	3	10	3	4	325.6	333.6	2.46%	63.4	64.8	2.21%
5	6	4	12	3	4	375.4	386.2	2.88%	68.6	70.4	2.62%

As can be seen in the table above, the relative deviations in both objective functions (greenhouse gas emissions and cost) compared to the solutions obtained from GAMS software are within an appropriate range. This indicates that the proposed hybrid algorithm is able to provide acceptable performance in small dimensions compared to known methods.

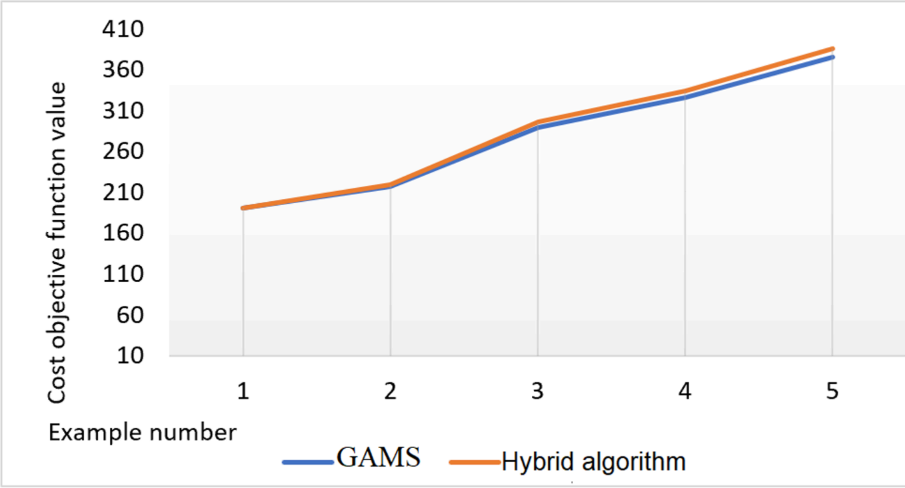

Figure 6: Comparison of the value of the Gamze solution with the hybrid algorithm for the cost objective function

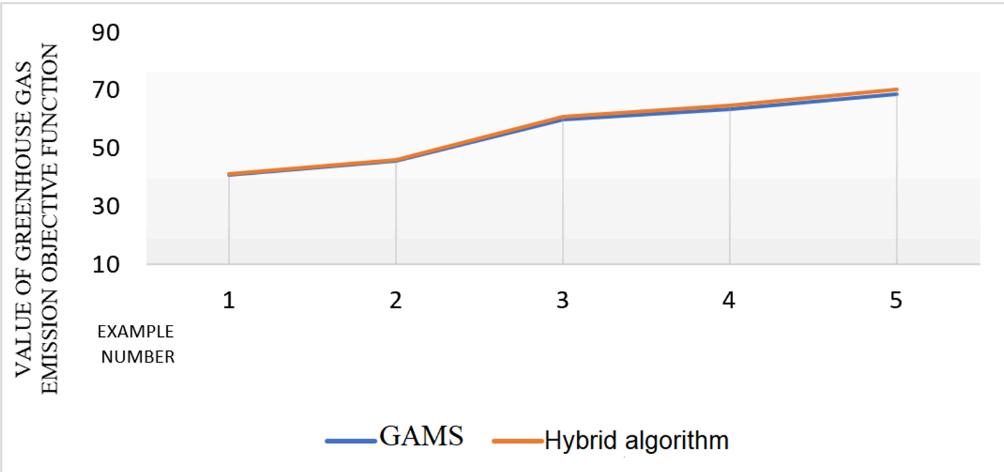

Figure 7: Comparison of GAMS solution value with hybrid algorithm for greenhouse gas emission objective function

As can be seen in Table 2, the objective function values for both hybrid algorithm and GAMS software are almost close to each other, which indicates the good performance of the hybrid algorithm. This close agreement of the results, especially at small problem sizes, proves that the hybrid algorithm has been able to provide answers effectively and acceptably that are similar to the exact answers of GAMS software. Small deviations in the objective function values, especially around 2.88%, indicate the good performance of the algorithm in solving small-scale problems.

In the next step, the comparison of the solution time of the examples presented using GAMS software and the hybrid algorithm is made in the graph. In this graph, the solution time of the hybrid algorithm is shown to be significantly lower than the solution time with GAMS software. This time difference means that the hybrid algorithm has been able to reach similar answers faster. This feature is a significant advantage, especially in larger problems where solution time is of great importance.

This difference in solution time shows the advantage of the hybrid algorithm in solving problems faster and also proves that this algorithm can be used as an effective and efficient method for solving complex problems in larger dimensions. Therefore, considering that computation time can play an important role in practical problems, the use of the hybrid algorithm can be considered as a suitable and fast choice compared to classical methods such as GAMS.

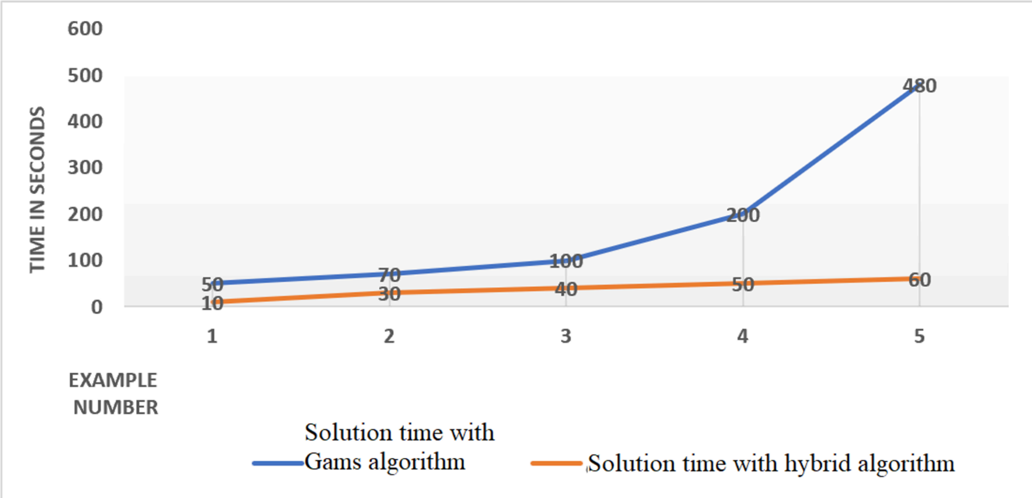

Figure 8: Comparison of solution time with the hybrid algorithm and the Gams algorithm

3.6 Evaluation of the algorithm's performance in large dimensions

Here, several criteria are used to evaluate the algorithm's performance, including: the difference between the best and worst solutions, the average deviation from the best solution, and the convergence of the algorithm. The differences observed in this study are very small, indicating the appropriate performance of the algorithm in finding solutions close to the best possible solution. In particular, the average deviation from the best solution for the cost objective function is reported to be 3.02% with a variance of 0.391 and for the greenhouse gas emission objective function to be 1.8% with a variance of 0.537. These values indicate the performance of the algorithm in finding appropriate solutions and reducing the dispersion in the results. As a result, the results obtained from the hybrid algorithm indicate its very high efficiency in solving multi-objective problems, and this algorithm has been able to optimize two different objectives (cost and greenhouse gas emissions) simultaneously and approach the optimal solutions, so that the deviation from the best solutions has been very low compared to other algorithms.

Table 4: Evaluation of the algorithm's efficiency for the cost objective function

Cost objective function														
variance	**worst deviation Average and worst deviation**	**Best answer**	**Average answer**	**Worst answer**	**Best answer**	**Execution**								Example number
						8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	
0.002	0.05%	0.19%	52.23	52.3	52.2	52.2	52.3	52.2	52.2	52.2	52.3	52.2	52.2	**1**
0.008	0.20%	0.43%	69.74	69.9	69.6	69.7	69.9	69.8	69.7	69.7	69.6	69.8	69.7	**2**
0.225	0.55%	1.40%	86.18	86.9	85.7	85.9	86.9	85.7	86.5	86.7	86.2	85.8	85.7	**3**
0.414	0.61%	1.75%	86.43	87.4	86.9	87.4	85.9	86.3	86.9	87.4	85.9	86.3	85.9	**4**
391.0	0.63%	3.02%	279.65	286.3	277.9	278.4	281.3	278.2	286.3	278.1	278.6	278.2	278.4	**5**

Table 5: Algorithm efficiency evaluation for greenhouse gas emission objective function

Greenhouse gas emissions objective function														
variance	**worst deviation Average and worst deviation**	**Best answer**	**Average answer**	**Worst answer**	**Best answer**	**Execution**								Example number
						8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	
0.006	0.17%	0.34%	58.30	58.4	58.2	52.2	52.3	52.2	52.3	52.3	52.4	52.4	52.3	**1**
0.014	0.22%	0.60%	67.25	67.5	67.1	67.1	67.2	67.3	67.2	67.2	67.5	67.3	67.2	**2**
0.071	0.44%	0.84%	83.56	83.9	83.2	83.2	83.4	83.8	83.9	83.9	83.5	83.4	83.4	**3**
0.376	0.54%	1.80%	95.11	96.3	94.6	94.7	95.3	95.7	94.9	96.3	94.8	94.6	94.6	**4**
0.537	0.61%	1.81%	105.44	106.7	104.8	104.8	104.8	106.7	106.1	105.9	105.5	104.9	104.8	**5**

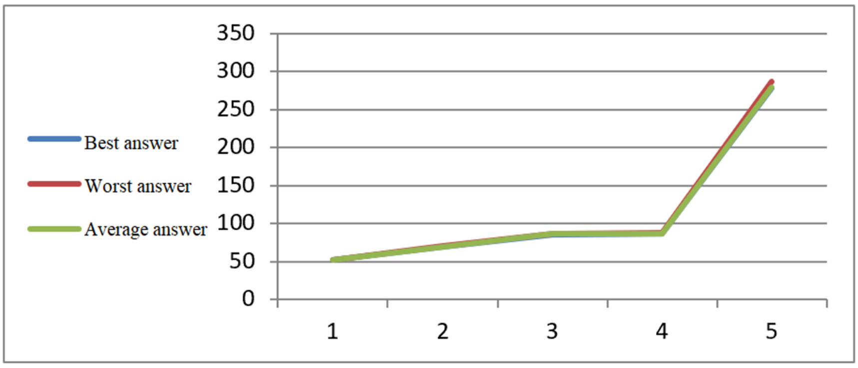

Figure 9: Checking the efficiency of the algorithm for the cost objective function

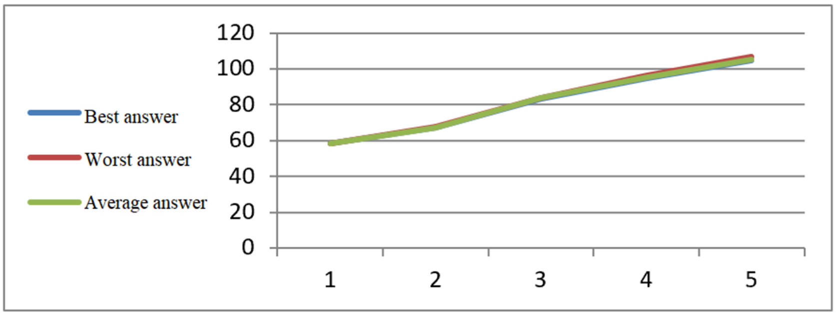

Figure 10: Algorithm efficiency evaluation for greenhouse gas emission objective function

3.7 Investigating the performance of the hybrid algorithm in large dimensions

In this part of the research, the aim is to investigate the performance of the hybrid algorithm in comparison with the NSGA-II algorithm in large dimensions. To perform this comparison, several random examples in large dimensions with different data have been prepared. In order to evaluate the results more accurately and validate them, the problem has been repeated 30 times. The reason for repeating this process is that running the problem a large number of times normalizes the results and reduces the effect of randomness.

After solving the problem 30 times with the hybrid algorithm, the best solution obtained was compared with the results obtained from the NSGA-II algorithm. The comparison of the results of these two algorithms is shown in Table 5, Figures 11 and 12. The results show that the difference between the solutions obtained from both algorithms was very small. The maximum difference observed between the solutions of the hybrid algorithm and the NSGA-II algorithm was about 2.98 percent.

These small difference values indicate the high efficiency of the hybrid algorithm in solving large-dimensional problems. In other words, the hybrid algorithm has been able to perform the necessary optimizations in large dimensions and provide solutions similar to the NSGA-II algorithm. Therefore, this comparison proves that the hybrid algorithm has a good performance in more complex conditions and larger dimensions and can approach the optimal solutions well.

In the following, a number of other examples in different dimensions will be presented to more accurately evaluate the performance of both algorithms in larger dimensions, in which the results for both algorithms will be compared and analyzed.

Table 6: Evaluation of the efficiency of the hybrid algorithm compared to NSGAII (Multi-objective Genetic Algorithm)

Example number	Number of sets				Objective function: Cost			Objective function: Greenhouse gas emissions		
	Customers	Distributors	Manufacturers	Suppliers	Hybrid algorithm	NSGII algorithm	Deviation	Hybrid algorithm	NSGII algorithm	Deviation
1	15	4	4	3	823.2	826.6	0.41%	202.3	202.4	0.05%
2	18	5	4	3	965.5	970.3	0.50%	226.4	227.1	0.31%
3	20	5	5	3	1100.3	1107.7	0.67%	246.6	249.2	1.05%
4	25	6	5	4	1320.1	1330	0.75%	269.7	276.2	.41%
5	30	6	6	4	1563.2	1579	1.01%	292.7	300	2.49%

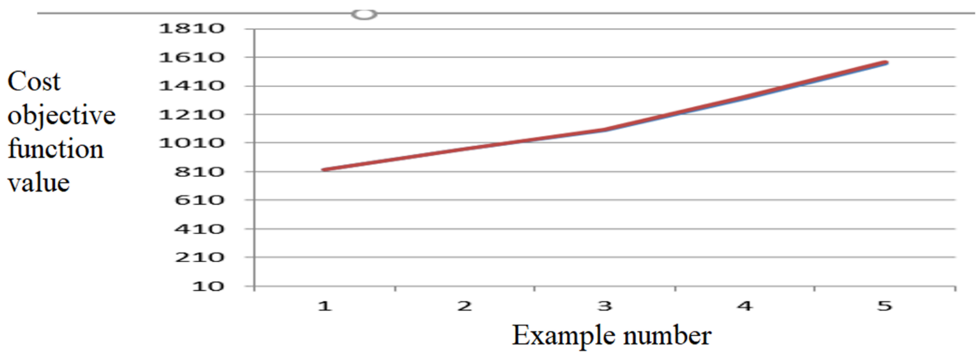

Figure 11: Evaluation of the efficiency of the hybrid algorithm compared to the NSGAII algorithm (multi-objective genetic algorithm) in the cost objective function.

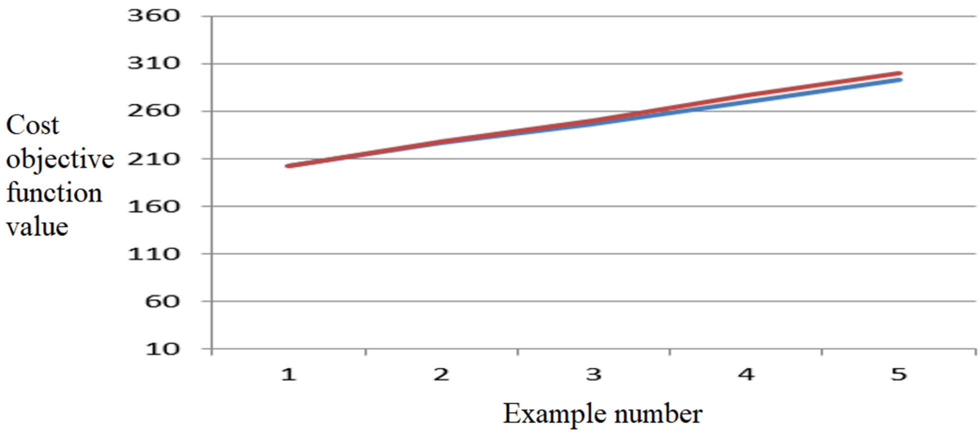

Figure 12: Evaluation of the efficiency of the hybrid algorithm compared to the NSGAII algorithm in the greenhouse gas emission objective function.

3.8 Comparison of problem solving time with the presented methods

In this section, a numerical example is presented in small and medium dimensions, which is solved using three different methods including the NSGA-II algorithm, the hybrid algorithm, and the GAMS software. The purpose of this comparison is to examine the problem solving time in different dimensions and evaluate the efficiency of each of these methods. The table below shows the problem solving time by these three methods. As can be seen in this table, the solution time in the algorithms for different dimensions has significant differences. In small dimensions (15 and 25), the solution time of the algorithms is almost similar, so that in these dimensions, the GAMS algorithm has performed slightly faster than other methods. However, in medium dimensions (45 and 78), the hybrid algorithm and NSGA-II show similar solution times, while the solution times in the GAMS software are significantly higher. This shows that as the problem dimensions increase, GAMS software gains significant solution time due to higher computational complexity. At larger dimensions (142), the solution time in GAMS software increases sharply and reaches 7201 time units, while the NSGA-II and hybrid algorithms have much lower solution times. This clearly shows that metaheuristic algorithms (hybrid and NSGA-II) perform better in solving large and complex problems. In other words, in large-dimensional problems, the use of metaheuristic algorithms significantly reduces the solution time and prevents the loss of efficiency of conventional software such as GAMS. The results obtained from this comparison clearly show the greater efficiency of hybrid algorithms and NSGA-II in solving complex and multi-objective problems compared to GAMS. This comparison proves the importance of using metaheuristic methods in solving complex problems in large dimensions, especially in multi-objective and combinatorial optimizations.

Table 7: Comparison of solution time using the presented method

Hybrid	NSGA II	GAMS
15	10	8
25	14	42
45	22	321
78	39	1865
142	58	7201

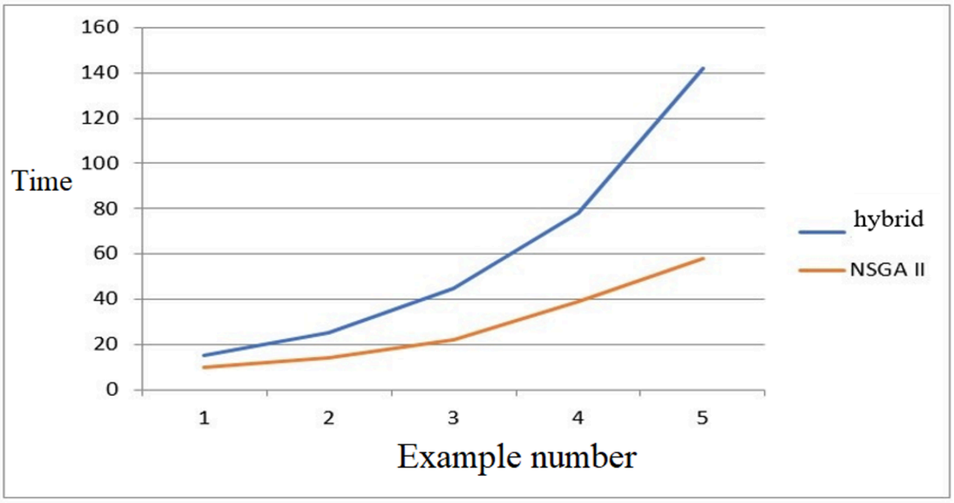

Figure 13: Comparison of solution time by NSGAII and hybrid

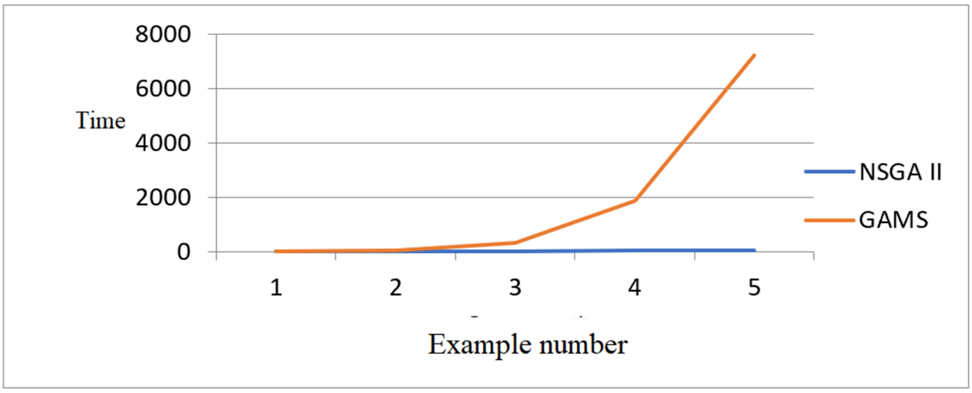

Figure 14: Comparison of solution time by NSGAII and GAMS

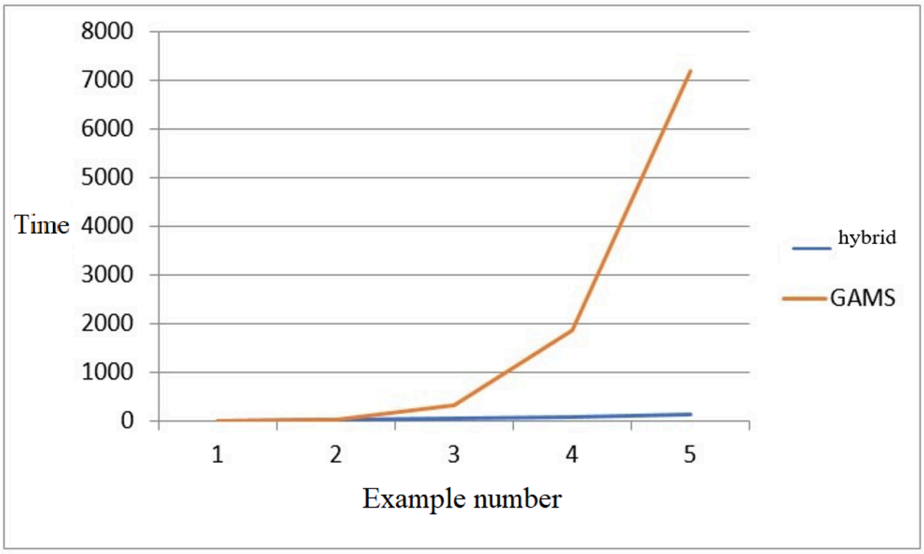

Figure 15: Comparison of solution time by hybrid and GAMS

3.9 Investigating the efficiency of the algorithm in small dimensions

In the previous section, it was proven that the problem presented is complex and that a hybrid algorithm is needed to solve the problem in large dimensions. At this stage, the efficiency of the algorithm in small dimensions is investigated. In order to investigate the correctness, accuracy and efficiency of the algorithm, thirty numerical examples in small dimensions have been randomly generated and investigated.

Table 8: Values of the indices of the two algorithms in the sample of solved problems

DM		**DIM**		**Q**		**N**		**Issue**
Hybrid	**GAMS**	**Hybrid**	**GAMS**	**Hybrid**	**GAMS**	**Hybrid**	**GAMS**	**Number**
0.983	0.897	1.066	0.851	0.858	0.715	8	4	1
1030.2	1.187	1.028	0.88	0.575	0.998	5	8	2
1.424	1.426	0.744	0.626	1.025	0.548	8	6	3
1.43	1.006	0.779	0.852	0.846	0.727	7	8	4
1.346	1.28	1.164	0.615	0.875	0.698	11	9	5
1.195	1.207	1.019	0.27	1.075	0.498	12	10	6
1.13	1.05	0.787	0.965	0.942	0.631	4	5	7
1.254	1.053	0.668	0.854	0.869	0.529	5	7	8
1.178	1.439	1.068	0.685	1.044	0.529	6	6	9
1.034	0.87	0.83	0.674	1.164	0.409	10	7	10
1.167	1.08	0.883	0.694	0.838	0.736	9	10	11
1.41	1.013	0.898	0.591	0.869	0.704	10	11	12
1.365	0.873	0.724	0.516	0.942	0.631	9	9	13
1.452	1.058	0.711	0.889	0.825	0.848	7	6	14
1.178	1.439	0.771	0.74	0.858	0.715	10	8	15
1.034	0.87	0.963	0.769	0.989	0.584	10	7	16
1.167	1.08	0.808	0.683	0.704	0.869	6	10	17
1.178	1.439	0.975	0.819	0.675	0.898	7	6	18
1.334	0.546	0.895	0.524	0.942	0.631	5	6	19
1.161	0.997	0.912	0.842	0.689	0.683	6	9	20
1.237	1.068	0.831	0.866	1.075	0.869	9	12	21
0.902	1.037	0.87	0.65	1.161	1.012	11	14	22
1.243	0.91	0.986	0.9	0.737	0.836	10	15	23
1.271	0.938	0.925	0.92	0.918	0.655	11	12	24
0.915	1324.2	1.02	0.765	1.008	0.565	7	8	25
1.415	1236.2	0.995	0.898	1.002	0.571	9	15	26
1.445	1.138	0.715	0.689	0.846	0.727	11	16	27
1.135	0.948	0.811	0.955	0.775	0.798	11	17	28

To compare the performance of two hybrid algorithms and the LP metric method in GAMS software, a pairwise comparison test is used in the evaluation indicators. In the pairwise comparison test, the assumption of equality of the means of two algorithms is tested against the assumption of inequality of their means. Given that the requirement for performing this test is the normality of the data, the Kolmogorov-Smirnov test is used first. In this test, the normality of the data is also tested against their non-normality.

SPSS software was used to perform both of the above tests. The results of the Kolmogorov-Smirnov test for the hybrid algorithm indicators are reported in Table 8 and for the GAMS indicators in Table 9. According to the tables, the last two rows show the z-test statistic and the P-VALUE value of the test (Sigs), respectively. According to this test, if the Sig value is greater than 5% in both tables, the normality of the data is confirmed. Therefore, considering the value of Sigs in both tables, the normality of the data is not rejected.

Table 9: K-S normality test for the combined algorithm indices

	N	Q	DIM	DM
N	30	30	30	30
normal parameters Mean	7.56	0.66	0.695	1.113
Std. Deviation2	1.88	0.164	0.351	0.178
Kolmogorov-Smirnov Z	0.691	0.535	0.77	0.752
Asymp. Sig. (2-tailed)	0.899	0.935	0.74	0.729

Table 11 also shows the results of the pairwise comparison test of the indicators. Rows 1 to 4 show the results of the pairwise comparison of the number of Pareto solutions, the quality of the solutions, the distance from the ideal,

Table 10: K-S normality test for GAMS software indices

	N	Q	DIM	DM
N	30	30	30	30
normal parameters Mean	7.097	0.466	0.59	0.917
Std. Deviation2	1.211 6	0.185	0.279	0.2887
Kolmogorov-Smirnov Z	1.321	0.585	0.821	0.745
Asymp. Sig. (2-tailed)	0.15	0.905	0.585	0.56

and the diversity of the solutions, respectively. The last column shows the P-VALUE value of the test. If the SIG value is less than 5%, the equality of the means of the two algorithms is rejected. Considering this point and the results of the table, it can be said that:

The performance of the two algorithms in the “number of Pareto solutions” does not have a significant difference.

The assumption of equality of the “quality of solutions” of the two algorithms is rejected, in other words, the quality of the GAMS solutions is better than the combined algorithm

The assumption of equality of the “distance from the ideal” for the two algorithms is rejected, in other words, the GAMS solutions have a smaller distance from the ideal than the combined solutions.

The assumption of equality of “diversity” of the solutions of the two algorithms is rejected, in other words, the diversity of the combined solutions is better than GAMS.

Table 11: Pairwise comparison test for the indices of the two algorithms

	Paired Differences						
			95% Confidence Interval of the Difference				
	.Std Deviation	**Std. Error Mean**	**Lower**	**Upper**	**t**	**df**	**.Sig (2-tailed)**
Pair 1 NSGA_N - GAMS_N	2.425	0.423	-0.8	0.932	3.85	29	0.004
Pair 2 NSGA_Q - GAMS_Q	0.280	0.055	0.069	0.293	3.283	29	0.003
Pair 3 NSGA_DIM - GAMS_DIM	1.790	0.034	0.09	0.228	12.4	29	0.000
Pair 4 NSGA_DM - GAMS_DM	2.560	0.052	0.066	0.280	4.215	29	0.003

Examining the complexity of the problem

The complexity of the problem is defined as the time to solve it in different dimensions. As the number of variables (n) and the input dimensions of the problem (m) increase, the solution time increases exponentially. This is shown using the following relationship:
$T(n, m) = O(nm)$
Or in some cases:
$T(n, m) = O(2n)$
This exponential increase in the solution time indicates the high complexity of the problem. In large dimensions, even powerful software such as GAMS is not able to solve the problem accurately.

Table of dimensions of examples

To examine the complexity of the problem, examples with different dimensions have been selected. The following table shows the dimensions of these 11 examples:

Table 12: Pairwise comparison test for the indices of two algorithms

Example number	Number of manufacturers	Number of distributors	Number of suppliers	Number of customer locations
1	2	3	2	6
2	2	3	2	7
3	2	3	2	8
4	3	3	2	9
5	3	4	2	10

Comparison of GAMS and Hybrid Algorithm Results The following table shows the differences between GAMS and Hybrid Algorithm results :

Table 13: Differences between GAMS and Hybrid Algorithm Results

Example number	Cost objective Cost objective function (GAMS)	Cost objective function Cost objective function (combined algorithm)	Deviation (%)
1	190.5	191	0.26
2	216.4	218.9	1.16
3	289.7	295.6	2.04

Maximum deviation: 2.88%

Therefore, the hybrid algorithm performs well in small dimensions.

Investigation of the efficiency of the algorithm in large dimensions

Evaluation indicators

Number of Pareto points: The higher the number of Pareto points, the more desirable it is.

Distance from the ideal point: The shorter the distance, the better.

Diversity: The higher the diversity, the better.

Comparison results: The hybrid algorithm performs better than GAMS in large dimensions.

Investigation of the efficiency of the algorithm in small dimensions

Results of statistical tests

K-S test: Normality of data confirmed.

Paired comparison test: GAMS and hybrid algorithm perform differently in some indicators.

Problem solving time comparison

Table 14: Solution time comparison

Example number	Solution time (GAMS)	Solution time (NSGA-II)	Solution time (hybrid)
1	8	10	15
2	42	14	25
3	321	22	45

In large dimensions, the GAMS solution time increases significantly.

Metaheuristic algorithms (NSGA-II and hybrid) have shorter solution times.

Conclusion

In this study, a framework for a sustainable supply chain in plant-based milk production based on the Internet of Things has been designed. The aim of this study is to design a system that uses the Internet of Things (IoT) to help improve and optimize the distribution, production, and inventory management processes in the plant-based milk supply chain. In this system, in order to prevent product spoilage and improve the distribution network, the inventory level in distribution centers is monitored using the Internet of Things technology. When the inventory level reaches a certain value, the ordering process is performed intelligently using the Internet of Things. In addition to ordering, this process includes intelligent routing in the distribution and logistics network, which is also carried out based on information received from Internet of Things devices, production planning, and the supply of raw materials and packaging. To solve the complex problems of this supply chain, GAMS software and the hybrid NSGA-II algorithm (multi-objective genetic algorithm) were used. This model proposes an IoT-based supply chain network that is able to retrieve data from IoT devices in real time and adjust them as inputs for optimization and decision-making in various processes of the system. In addition, the proposed model pays special attention to sustainability aspects, and since sustainability criteria are specifically considered as objective functions in this research, this system will help reduce negative environmental impacts, improve social conditions, and increase economic profitability.

Given that supply chain management includes planning, designing, executing, controlling, and monitoring supply chain activities to cover objectives such as value creation, net creation of a competitive base, helping global logistics, coordinating supply with demand, and measuring performance, therefore, the need to have a comprehensive and holistic view at the supply chain levels is felt.

Also, given the growing demand for plant-based milks, the production process of these products should be designed to be similar in composition and structure to cow's milk. These products are usually produced from plant seeds and vegetable oils, which can be processed in various ways such as soaking, grinding, and homogenization. The production processes of these alternatives should be such that they create a colloidal structure similar to cow's milk in order to have the desired quality characteristics such as creamy texture and desired taste. In this regard, the composition of proteins, polysaccharides, sugars, and minerals in plant-based milks, similar to cow's milk, is of great importance in order to improve the taste and nutritional properties.

Finally, this research has designed a sustainable supply chain for plant-based milk production that can optimize internal processes using the Internet of Things and simultaneously pay attention to economic, social, and environmental sustainability aspects. This model can help improve productivity, reduce costs and negative environmental impacts and serve as a suitable framework for plant-based milk companies worldwide. The analysis also showed that the solution times of the algorithms differ significantly for different dimensions. At small dimensions (15 and 25), the solution times of the algorithms are almost similar, so that at these dimensions, the GAMS algorithm performed slightly faster than the other methods. However, at medium dimensions (45 and 78), the hybrid algorithm and NSGA-II show similar solution times, while the solution times in the GAMS software are significantly higher. This indicates that as the problem dimensions increase, the GAMS software gains significant solution times due to its higher computational complexity. In larger dimensions (142), the solution time in GAMS software increases sharply and reaches 7201 time units, while the NSGA-II and hybrid algorithms have much shorter solution times. This clearly shows that metaheuristic algorithms (hybrid and NSGA-II) have better performance in solving large and complex problems. In other words, in large-dimensional problems, the use of metaheuristic algorithms significantly reduces the solution time and prevents the loss of efficiency of common software such as GAMS. The results obtained from this comparison clearly show the greater efficiency of hybrid algorithms and NSGA-II in solving complex and multi-objective problems compared to GAMS software. This comparison proves the importance of using metaheuristic methods in solving complex problems in large dimensions, especially in multi-objective and hybrid optimizations.

References

[1] A. M. Aamer, M. A. Al-Awlaqi, I. Affia, S. Arumsari and N. Mandahawi, *The internet of things in the food supply chain: adoption challenges*, Benchmarking: An International Journal **28** (2021), no. 8, 2521-2541.

[2] N. Adamashvili, N. Zhizhilashvili and C. Tricase, *The integration of the internet of things, artificial intelligence, and blockchain technology for advancing the wine supply chain*, Computers **13** (2024), no. 3, 72-84.

[3] A. S. Alluhaidan, M. S. Alluhaidan and S. Basheer, *Internet of Things Based Intelligent Transportation of Food Products During COVID*, Wirel. Pers. Commun **34** (2021), no. 27, 19-31, doi: 10.1007/s11277-021-08777-6.

[4] L. Bibby and B. Dehe, *Defining and assessing industry 4.0 maturity levels–case of the defence sector*, Prod. Plan. Control **29** (2018), no. 4, 1030-1043.

[5] L. Christian, S. Martin and T. Schäffer, *Industry 4.0 and Lean Production—A matching relationship? An analysis of selected Industry 4.0 models*, In Proceedings of the 2017 Federated Conference on Computer Science and Information Systems (FedCSIS), Prague, Czech Republic; IEEE: New York, NY, USA, 2017, 989-993.

[6] Z. Kamar, A. Cherrafi, I. Bouhaddou, A. C. Benabdellah and R. Raut, *A holonic architecture for the supply chain performance in industry 4.0 context*, Int. J. Logist. Res. Appl **12** (2021), no. 6, 1-28.

[7] D. Kumar, R. K. Singh, R. Mishra and T. U. Daim, *Roadmap for integrating blockchain with Internet of Things (IoT) for sustainable and secured operations in logistics and supply chains: Decision making framework with case illustration*, Technological Forecasting and Social Change **196** (2023), no. 11, 155-166.

[8] S. Mantravadi and J. Singh Srai, *How Important are Digital Technologies for Urban Food Security? A Framework for Supply Chain Integration using IoT*, Procedia Computer Science **217** (2023), no. 6, 1678-1687, https://doi.org/10.1016/j.procs.2022.12.368.

[9] H. Maulana, S. L. B. Ginting, P. Aryan, M. R. Fadillah, R. N. Kamal, *Utilization of internet of things on food supply chains in food industry*, International Journal of Informatics, Information System and Computer Engineering (INJIISCOM) **2** (2021), no. 1,103-112.

[10] B. Mosallanezhad, F. Gholian-Jouybari, L. E. Cárdenas-Barrón, M. Hajiaghaei-Keshteli, *The Iot-enabled sustainable reverse supply chain for COVID-19 Pandemic Wastes (CPW)*, Engineering Applications of Artificial Intelligence **120** (2023), no. 4, 105903, https://doi.org/10.1016/j.engappai.2023.105903.

[11] M. C. Schmidt, J. W. Veile, J. M. Müller, K. I. Voigt, *Industry 4.0 implementation in the supply chain: A review on the evolution of buyer-supplier relationships*, Int. J. Prod. Res. **16** (2022), no. 10, 1-8.

[12] A. Tang, E. Tutu Tchao, A. Selasi Agbemenu, E. Keelson, G. Selorm Klogo and J. John Kponyo, *Assessing blockchain and Iot technologies for agricultural food supply chains in Africa: A feasibility analysis*, Heliyon **10** (2024), no. 15, 70-89, https://doi.org/10.1016/j.heliyon.2024.e34584.

[13] R. Tavakkoli-Moghaddam, J. Ghahremani-Nahr, P. Samadi Parviznejad, H. Nozari and E. Najafi, *Application of internet of things in the food supply chain: a literature review*, Journal of applied research on industrial engineering **9** (2022), no. 4, 475-492.

[14] I. Vlachos, R. M. Pascazzi, M. Ntotis, K. Spanaki, S. Despoudi, P. Repoussis, *Smart and flexible manufacturing systems using Autonomous Guided Vehicles (AGVs) and the Internet of Things (IoT)*, International Journal of Production Research **62** (2024), no. 15, 5574-5595.