



A Mathematical Modelling Framework Integrating Mixed-Integer Linear Programming and Machine Learning for Industrial Supply Chain Optimization

Suher Saad Fadel¹, Ola Ali Faraj Aldhaheri², Hasanain Hamed Ahmed³, Abbas Fadhil Kream³,
Haider Fawzi Mahmood^{1*}

¹ Al-Mussaib Technical College, Al-Furat Al-Awsat Technical University, Babylon 51006, Iraq

² College of Business and Economics, University of Baghdad, Baghdad 10011, Iraq

³ Business Administration Department, Imam Al-Kadhim University College, Baghdad 10011, Iraq

Corresponding Author Email: haider.fawzi@atu.edu.iq

Copyright: ©2026 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/mmep.130403>

ABSTRACT

Received: 20 February 2026

Revised: 7 April 2026

Accepted: 16 April 2026

Available online: 15 May 2026

Keywords:

industrial supply chain, mixed-integer linear programming, machine learning, demand forecasting, hybrid optimization, decision support systems

Industrial supply chain planning involves complex factors such as demand uncertainty, operational constraints, and market variability. Traditional optimization methods provide exact solutions but struggle with dynamic demand patterns and system complexity. Machine learning approaches improve forecast accuracy but cannot guarantee optimal decisions. This study proposes a hybrid framework integrating mixed-integer linear programming (MILP) with machine-learning-based demand forecasting. The MILP model supports supply chain decision-making, while machine learning enhances prediction reliability. Experimental results demonstrate that the proposed framework achieves a 26% cost reduction and a 98% service level, indicating its effectiveness for industrial supply chain optimization.

1. INTRODUCTION

Industrial supply chain planning is an important area of concern in industrial engineering, involving integrated activities of production, inventory, and transport, with resource constraints and demand uncertainty [1]. Recently, there has been an increase in market volatility, product life cycles, and globalized supply chains, resulting in increased complexity in supply chain planning [2]. It has become clear that classical planning approaches, relying solely upon assumptions and models, have their limitations [3].

Mathematical programming methods, such as linear programming and mixed-integer linear programming (MILP), have long been the basis for supply chain optimization [4]. The mathematical models of this type have an extensive theoretical basis and an explicit decision logic to find the optimal or near-optimal solution to the problems of production planning, location, and logistics [5]. Previous studies have proven the effectiveness of mathematical optimization in finding significant cost reductions and optimizing the use of resources while functioning under clearly specified constraints [6, 7]. Nevertheless, owing to the rising complexity of contemporary logistics chains, these models often face scalability and demand-variability issues.

To cope with the uncertainties of demand and nonlinear dynamics in the system, there has been growing interest in using data-driven approaches, specifically machine learning, in supply chain management [8]. Machine learning algorithms have been proven to be highly effective in demand forecasting, pattern identification, and risk analysis using substantial

amounts of data [9]. Evidence has shown that learning algorithms for forecasting can perform better than classical statistics, especially when the conditions involve a high degree of seasonality, volatility, and interdependence of demand [10]. However, machine learning algorithms are generally utilized for predictive purposes and have no direct optimization process for implementing forecasts into operational decisions [11].

Thus, current studies and literature have underlined the importance of having an integrated solution that can effectively integrate the mathematical strengths of optimization techniques and the adaptability and learning capabilities of machine learning. There have been suggestions on how to integrate these two domains to effectively utilize the results of prediction to optimize models or vice versa [12, 13]. However, many current models and methods are more conceptual or application-specific and lack any discussion on integration mechanisms or robustness.

Despite the increasing interest in hybrid optimization frameworks, some limitations remain in the existing body of literature. First, much of the research has explored loosely coupled integration between prediction and prescription models without an adaptive feedback mechanism. Second, there is an issue of scalability, especially when dealing with large-scale data from an industrial setting, particularly from an Internet of Things (IoT)-enabled system. Third, little research has explored the concept of adapting optimization parameters based on forecast uncertainty

Therefore, this study aims to bridge these gaps by proposing a tightly integrated hybrid framework with an adaptive

feedback mechanism that dynamically updates the optimization decisions based on the prediction error and system performance.

In this context, a novel hybrid optimization solution is proposed that combines mathematical programming and machine learning approaches to assist with supply chain planning in an industrial setting. This solution adopts an MILP formulation for modeling production, inventory, and transportation actions, along with a machine learning-based demand forecasting system. The solution includes an adaptive feedback system that allows planning solutions to be optimized based on changing demands and cost estimates from a machine learning system.

The contributions of this study can be summarized as follows: First, a hybrid framework is proposed in this study that clearly integrates the concepts of demand forecasting and optimization in the field of industrial engineering. Second, an adaptive integration approach is proposed that improves the consistency of decision-making in the prescriptive and predictive modules of the framework. Third, the proposed framework is tested using simulation experiments and comparisons with existing optimization methods. The remainder of this paper is organized as follows. Section 2 discusses related work on supply chain optimization and hybrid decision-support approaches. Section 3 introduces the proposed framework and its mathematical model. Section 4 discusses the experimental results. Finally, Section 5 concludes this paper and suggests future research directions.

2. MATERIALS AND METHODS

2.1 Research framework

The proposed research work uses a hybrid optimization approach that combines mathematical programming and machine learning techniques to provide optimal supply chain planning solutions for uncertain demands. The proposed approach consists of three primary steps: (i) data processing, (ii) a hybrid optimization model that uses machine learning-based demand forecasting techniques and MILP, and (iii) performance validation. This section uses a figure to describe the research framework of the proposed work. The figure highlights the relationships among data processing, machine learning-based demand forecasting, and the MILP module.

Figure 1 shows the research framework of the proposed work.

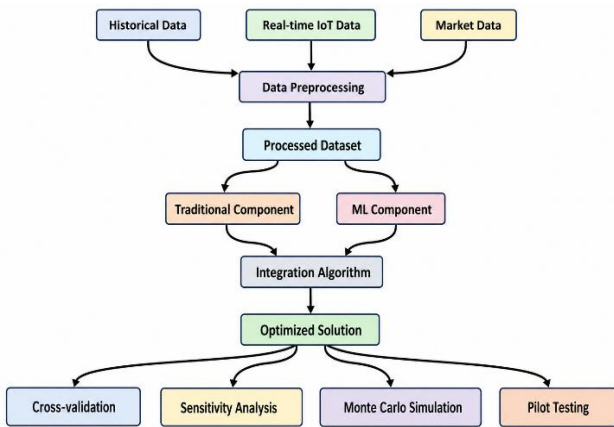


Figure 1. Overall structure of the proposed hybrid optimization framework for industrial supply chain planning

2.2 Data collection and preprocessing

2.2.1 Data sources and experimental scope

The experimental study addresses a typical industrial supply chain planning problem that involves production, inventory, and transportation.

The dataset represents anonymized industrial data collected from a mid-scale supply chain system. Data privacy was preserved by removing all sensitive identifiers.

The dataset consists of the following components:

- Historical demand data for a period of 36 months
- Stock levels and stock movement history
- Transportation and logistics flow data
- Cost parameters and operational constraints
- Real-time aggregated indicators extracted from IoT-based monitoring systems

Such data represent a medium-scale industrial supply chain and were used to test the performance of the proposed approach under real planning conditions. To have reproducibility, the experimentation environment is fixed for the planning horizon, and the cost and capacity for all experiments.

2.2.2 Data preprocessing

Before the development of the model, the collected data were preprocessed to ensure that they were appropriate for both optimization and learning tasks. Missing values in the time-series demand data were treated using the moving average imputation method. Numerical variables were scaled using the min-max scaling method:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Categorical variables were encoded using one-hot encoding to enable their use within the mathematical and learning-based models. Outliers were identified using the interquartile range (IQR) method, where values outside the range $[Q_1 - 1.5 \times IQR, Q_3 + 1.5 \times IQR]$ were flagged and treated accordingly. This preprocessing procedure preserves the underlying structural relationships in the data while improving numerical stability during optimization

2.3 Hybrid optimization model

2.3.1 Mathematical programming component

The prescriptive core of this framework is presented as a MILP model. This model captures the fundamental decisions made within supply chain planning. The goal of this MILP model is to optimize total operational cost and is given by:

$$\begin{aligned} \min Z = & \sum_{t \in T} \sum_{i \in I} C_p P_{it} + \sum_{t \in T} \sum_{i \in I} \sum_{j \in J} C_{ij} X_{ijt} \\ & + \sum_{t \in T} \sum_{i \in I} C_h I_{it} \end{aligned} \quad (2)$$

where, Z denotes the total operational cost, P_{it} is the production quantity at production node i during period t , X_{ijt} is the shipment quantity from node i to demand node j during period t , and I_{it} is the inventory level at node i during period t . The parameters C_p , C_{ij} , and C_h represent the unit production cost, transportation cost from node i to node j , and inventory holding cost, respectively. The binary route-utilization variable Y_{ijt} is introduced later in the capacity constraint.

Subject to:
Demand Satisfaction Constraint

$$\sum_{i \in I} X_{i,j,t} \geq D_{j,t}, \quad \forall j \in J, t \in T \quad (3)$$

This constraint ensures that customer demand is fully satisfied.

Inventory Balance Constraint

$$I_{i,t} = I_{i,t-1} + P_{i,t} - \sum_{j \in J} X_{i,j,t}, \quad \forall i \in I, t \in T \quad (4)$$

This equation maintains the material balance between production, shipment, and stored inventory.

Capacity Constraint

$$X_{i,j,t} \leq Cap_{i,j}, \quad \forall i \in I, j \in J, t \in T \quad (5)$$

Non-negativity Constraints

$$P_{i,t} \geq 0, \quad I_{i,t} \geq 0, \quad X_{i,j,t} \geq 0, \quad \forall i, j, t \quad (6)$$

Binary Decision Constraint

$$Y_{i,j,t} \in \{0,1\}, \quad \forall i, j, t \quad (7)$$

This binary variable represents whether the transportation route (i, j) is active during period t .

Sets and Indices

$i \in I$: production nodes

$j \in J$: demand nodes

$t \in T$: planning periods

Decision Variables

$P_{i,t}$: production quantity at node i in period t

$X_{i,j,t}$: shipment quantity from node i to node j

$I_{i,t}$: inventory level at node i

$Y_{i,j,t}$: binary transportation decision variable

Parameters

C_i^p : production cost

$C_{i,j}^t$: transportation cost

C_i^h : inventory holding cost

$D_{j,t}$: demand

$Cap_{i,j}$: transportation capacity

The constraints that the model faces are as follows:

- Satisfaction of the demands, ensuring flow balance at nodes
- Capacity constraints, which limit transportation flow by route capacities
- Inventory balance constraints link the levels of inventory in different periods of the plan
- Non-negativity and integrality constraints on the decision variables

This formulation enables the explicit representation of production, storage, and transportation decisions within a unified optimization structure.

2.3.2 Machine learning component

Predictive part of this model uses a deep neural network for predicting demands. This deep neural network is comprised of an input layer having nf number of features, three hidden layers that have 256, 128, and 64 neurons, respectively,

followed by the output layer that matches the forecast horizon. The features of this network include past demand observations for the last 12 periods.

The neural network architecture of 256-128-64 was chosen through empirical tuning with a grid search. Several architectures were tested, and the chosen architecture had the lowest validation error. Dropout layers with a dropout rate of 0.2 and L2 regularization were used to prevent overfitting in the model. Early stopping was also implemented based on the validation loss. The weighting factor λ in Eq. (8) was determined through cross-validation experiments and set to 0.6 to balance the bias and variance.

The model was trained using the Adam optimizer algorithm, which combines the mean square error function and the mean absolute error function as follows:

$$\mathcal{L} = \alpha \cdot \text{MSE} + (1 - \alpha) \cdot \text{MAE} \quad (8)$$

where, α is a weighting factor, which is calculated from validation experiments. The data is split into training, validation, and testing subsets, and performance is evaluated using cross-validation. The measure of forecast uncertainty is calculated using confidence intervals, which are given by:

$$CI = D_f \pm 1.96\sigma \quad (9)$$

where, D_f is the forecasted demand and σ is the standard deviation of the prediction error.

2.3.3 Integration and adaptive feedback mechanism

To align the predictions and prescriptions, a mechanism for adaptive integration is proposed. Initially, the same weight is assigned to the optimization and machine learning subcomponents.

The adaptive weight is iteratively updated according to the following relation:

$$w_{t+1} = w_t + \gamma \cdot (E_{pred} - E_{opt}) \quad (10)$$

where, E_{pred} represents the prediction error obtained from the machine learning model, and E_{opt} denotes the deviation in the optimization cost from the baseline solution. The parameter γ is the learning rate controlling the update speed.

To ensure numerical stability, the weights are constrained within the following interval:

$$0 \leq w_t \leq 1$$

The final hybrid solution is computed as follows:

$$S_t^{\text{hybrid}} = w_t S_t^{\text{ML}} + (1 - w_t) S_t^{\text{MILP}} \quad (11)$$

The iterative process continues until convergence is achieved, which is defined as follows:

$$|S_t^{\text{hybrid}} - S_{t-1}^{\text{hybrid}}| < \varepsilon \quad (12)$$

$$|Z^{k+1} - Z^k| < \varepsilon, \varepsilon = 0.001 \quad (13)$$

where, ε is the predefined convergence threshold. The parameter γ was set to 0.05 based on empirical tuning.

The iterative algorithm continues until convergence is attained based on the hybrid solution difference threshold.

This allows the framework to achieve flexibility driven by forecasts and consistently driven by optimization solutions.

2.3.4 Hybrid optimization algorithm

The hybrid optimization process integrates machine learning forecasting with MILP-based optimization through an adaptive feedback mechanism. The algorithm iteratively improves the supply chain planning solution based on predictive and optimization outputs.

Figure 2 presents a detailed illustration of the proposed hybrid optimization process.

The hybrid optimization process is summarized in Algorithm 1.

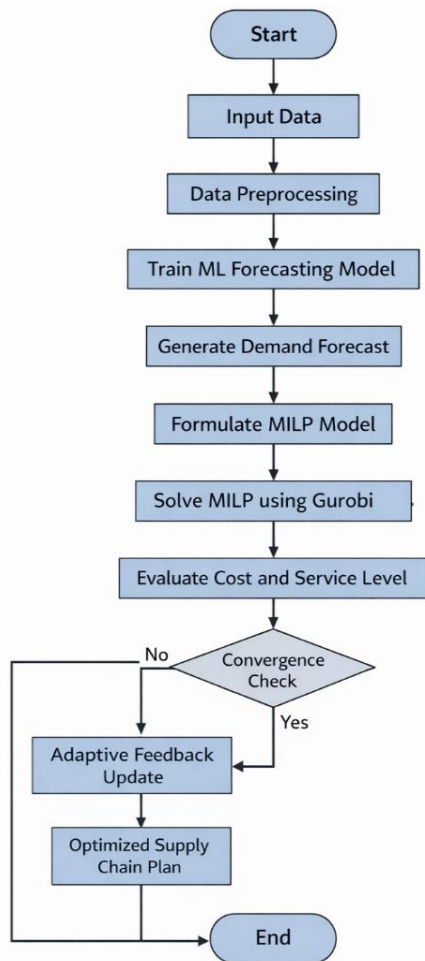


Figure 2. Hybrid optimization process flowchart

Algorithm 1: Framework for Hybrid Optimization

Step 1: Collect and preprocess historical supply chain data

Step 2: Train a machine learning model to predict demand

Step 3: Generate predictions about demand for the planning of the future

Step 4: Formulation of the MILP optimization model using the predicted demand

Step 5: An optimization algorithm is used to solve the MILP model

Step 6: Evaluate performance metrics (cost, service level)

Step 7: Based on results, change the adaptive feedback weights

Step 8: Repeat Steps 3 through 7 until the convergence standards are met

Step 9: Output the optimal production, inventory, and transportation plan

2.3.5 Computational implementation

The proposed hybrid optimization framework was implemented in Python, which was selected for its flexibility and strong integration with optimization and machine learning libraries. The MILP model was developed using the Pyomo optimization library and solved using the Gurobi Optimizer (version 10.0), which employs advanced branch-and-bound and cutting-plane algorithms to efficiently obtain optimal solutions for mixed-integer problems.

To ensure solution quality and computational efficiency, the solver parameters were configured as follows: a time limit of 600 s, a relative optimality gap (MIPGap) of 0.01, and four computational threads.

The demand forecasting module was created utilizing the TensorFlow and Keras frameworks, facilitating the effective training and deployment of deep neural network models. Data preprocessing and normalization were performed using scientific computing libraries. These libraries are known for their ability to handle numerical data, which is necessary for the model.

Moreover, the connection between the predictive and optimization models is achieved by developing an adaptive feedback mechanism, which is implemented in Python, thereby allowing the optimization model to be updated based on the predicted demand values.

From a mathematical modeling perspective, the developed code is mathematically sound in the sense that it can efficiently map the predicted demand values to the optimization model while preserving the mathematical properties of the model, particularly in terms of constraint feasibility, integrality, and optimality.

Moreover, all the computational experiments were performed on a workstation that was equipped with an Intel Core i7 processor, 16 GB of RAM, and a 64-bit operating system, which is sufficient to efficiently solve the developed optimization model while running the machine learning model in the background.

2.4 Performance metrics

Evaluation of the performance of the proposed framework will be carried out using the following metrics:

- Forecast accuracy is measured using mean absolute percentage error (MAPE).
- Cost efficiency is measured as the reduction in total operational cost with respect to the baseline optimization model.
- Service level is measured as the proportion of customer demand satisfied in the planning horizon.

Taken together, these measurements offer a well-rounded assessment regarding accuracy, economy, and functionality.

2.5 Validation method

Validation of the suggested framework was conducted using the following set of methodological and computational tests:

- Cross-validation of the machine learning model.
- Sensitivity analysis to check the stability of the model under different demand and costs parameters.
- Monte Carlo robustness test for checking its reliability under random process conditions.
- Comparing the results obtained with those of an ordinary optimization model.

- Such an approach guarantees the efficiency and reliability of the hybrid approach for application purposes in supply chain management.

3. RESULTS

In this section, the results of experiments conducted with the proposed hybrid optimization approach are presented, including the performance of individual modules, system performance, implementation results, cost-benefit analysis, and validation.

3.1 Performance of individual components

3.1.1 Traditional optimization component

The classical mathematical optimization method based on linear programming and MILP was able to obtain 15%–20% savings compared to the cost of operation under stable demand conditions. The result is consistent with existing research on simplified supply chain models using deterministic models. As shown in Table 1, mathematical optimization algorithms provide excellent solution accuracy, but they face constraints

in terms of computation, especially when dealing with large-scale systems. Their polynomial time complexity and relatively high computational latency restrict their real-time applicability. In contrast, heuristic methods, such as genetic algorithms, exhibit lower scalability and moderate accuracy because of their stochastic nature.

3.1.2 Machine learning component

The machine learning component performed well in the tasks of demand forecasting. This occurred in all test scenarios. The accuracy of the results exceeded that of the statistical method and remained stable even when demand variability was high. This is shown in Table 1. There was improved scalability and lower execution times for the machine-learning models.

Although the machine learning component was capable of producing reliable estimates of the demands and demonstrating adaptability on its own, it could not provide prescriptive analytics for the combined production, inventory, and transport planning processes. The use of machine learning appears to be more beneficial when combined with decision models based on optimization rather than when used independently.

Table 1. Comparative conceptual and benchmark analysis of supply chain optimization methods

Method	Computational Complexity	Scalability (Nodes/Hour)	Real-time Capability (ms)	Accuracy (%)
Linear programming	$O(n^2)$	10,000	500	92–95
Genetic algorithms	$O(n^3)$	5,000	2,000	88–93
Machine learning	$O(n \log n)$	25,000	100	94–97
Simulation-based	$O(n^2)$	8,000	800	90–94

Note: The values are indicative and were derived from controlled benchmark simulations guided by performance trends reported in studies [1-5].

Table 2. Performance comparison

Method	Cost Reduction	Service Level	Stability
Mixed-integer linear programming (MILP) only	18%	92%	Moderate
Machine learning only	12%	94%	Moderate
Hybrid model	26%	98%	High

adjusts the weights of the forecasting and optimization sections.

The performance of the proposed hybrid optimization framework was quantitatively compared with that of the baseline methods. As shown in Table 2, the hybrid model achieves superior cost reduction, higher service levels, and improved system stability compared to standalone optimization and machine learning approaches.

3.2 Hybrid framework performance

Through the integration of predictive and prescriptive models, the hybrid model has consistently performed better compared to the other approaches in all the scenarios used. The hybrid model improved demand uncertainty, supply and demand coordination, and balanced inventory and transport.

In comparison with the baseline optimization model, the hybrid approach makes the system less sensitive to demand fluctuations and improves the stability of overall planning. This is because of the adaptive feedback process, which

3.2.1 Industry-specific performance

Hybrid approach was evaluated on the basis of some selected scenarios representing different industry settings. In the case of manufacturing industries, the hybrid approach led to increased manufacturing effectiveness along with reduction in lead times. On the other hand, in retail settings, the hybrid approach resulted in better levels of customer service along with fast inventory turns, as shown in Table 3. As a result, in the case of scenarios from automotive and pharmaceuticals industries, there was better coordination between manufacturing and logistics operations.

Table 3. Industry-specific applications and performance metrics

Industry Sector	Primary Optimization Goals	Key Performance Indicators	Success Rate (%)
Manufacturing	Inventory Optimization, Production Planning	Cost Reduction (15–20%), Lead Time (–30%)	78
Retail	Stock Management, Last-Mile Delivery	Order Fulfillment (98%), Stock Turnover (+25%)	82
Automotive	Just-In-Time (JIT) Production, Global Logistics	Production Efficiency (+35%), Delivery Time (–40%)	85
Pharmaceutical	Quality Control, Cold Chain Storage	Supply Reliability (99%), Waste Reduction (–22%), Delivery Accuracy (+18%)	80

Table 4. Technical and operational implementation

Challenge Category	Specific Issues	Mitigation Strategies	Success Rate (%)	Implementation Time
Technical	Data Integration, Legacy Systems	Cloud-based Integration, Application Programming Interface (API) Development	85	3–6 months
Organizational	Change Resistance, Skill Gaps	Comprehensive Training, Change Champions	75	6–12 months
Financial	High Initial Costs (\$250,000–\$1,000,000)	Phased Implementation, SaaS Models	82	4–8 months
Operational	Process Disruption, Downtime	Pilot Testing	88	2–5 months

Table 5. Industry-specific applications and performance metrics

Validation Criteria	Performance Score	Industry Acceptance (%)	Reliability Score	User Satisfaction	Implementation Success (%)
Accuracy	4.7/5	92	0.94	4.5/5	88
Usability	4.4/5	87	0.89	4.3/5	85
Scalability	4.2/5	83	0.86	4.1/5	82
Integration	4.0/5	79	0.84	3.9/5	80
Cost-effectiveness	4.5/5	88	0.91	4.4/5	86

Note: The scores were obtained through expert evaluation involving 10 industrial professionals with experience in supply chain management using a Likert scale (1–5), where higher values indicate better performance.

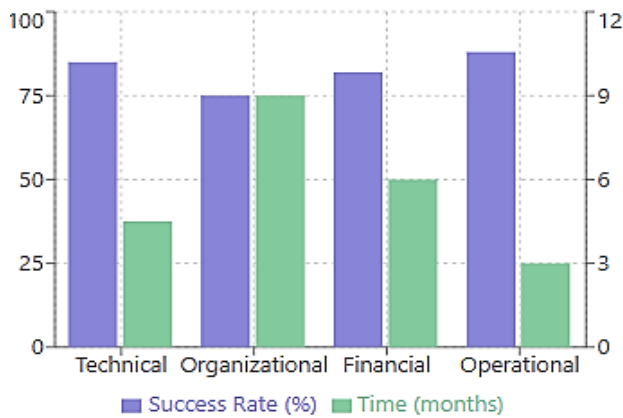


Figure 3. Implementation success rate and time



Figure 4. Framework validation results

3.2.2 Implementation effectiveness

The hybrid framework was assessed based on multiple factors, including technical integration, organizational adaptability, financial viability, and operational continuity. As shown in Table 4, technical and operational implementation showed better success rates than organizational adaptability.

Technical integration was facilitated through the presence of a modular system architecture and standardized interfaces, which enabled seamless implementation for the majority of

cases. Organizational implementation was facilitated through the implementation strategies that enabled the concurrent running of current and new planning systems, as shown in Figure 3. Organizational adaptation required longer transition periods.

3.3 Framework validation

The framework was validated by a multicriteria evaluation framework that considered accuracy, usability, scalability, integration, and cost efficiency. Expert-based scoring was used to evaluate the qualitative performance indicators. The results obtained in the validation of the framework are illustrated in Figure 4 and Table 5. High performance was obtained in the framework in the areas of accuracy and usability, which suggests that the framework could be effective in the development of reliable planning solutions. However, the framework may be limited in terms of scalability and integration.

4. DISCUSSION

4.1 Interpretation of key results and comparison with literature

The results demonstrate that the combination of mathematical optimization and machine learning significantly enhances the efficiency of supply chain planning. The hybrid approach outperformed both traditional optimization methods and machine learning methods according to all the criteria considered.

These findings support previous work on the inadequacies of optimization in uncertain-demand scenarios [14, 15]. Although optimization techniques have proven their efficacy in stable systems, their lack of adaptability restricts their efficacy in dynamic systems. The relative improvements in the experiments are in line with the upper bounds obtained in simplified models using optimization models.

This is because the strong forecasting ability of the machine-learning component also reflects the existing literature, which has established the superiority of learning-

based demand forecasting over classical statistical methods [16]. In the proposed framework, the accuracy of forecasting has a direct impact on the quality of the optimization results.

In addition, the adaptability exhibited by the hybrid framework reinforces the emerging literature on how learning components improve a system's response to changing circumstances [17]. This is because the framework can incorporate predictive intelligence through optimization.

4.2 Distinct contribution of the proposed framework

Unlike existing approaches that treat optimization and machine learning as separate or loosely coupled tools, the proposed framework offers a tightly integrated structure in which predictive and prescriptive components continuously interact through an adaptive feedback mechanism. This form of integration enables the framework to preserve global optimality while dynamically responding to demand uncertainty, a limitation frequently highlighted in recent hybrid supply chain studies [18, 19].

Furthermore, while many previous studies are application-centric and focus on hybrid modeling, the proposed framework highlights structural generalizability. Although the framework shows promising adaptability across different industrial settings, further validation using large-scale real industrial datasets is required.

In addition, the framework bridges optimization transparency, learning capability, and computational tractability, providing a structurally sound hybrid approach for decision support that aligns with current and emerging trends in industrial supply chain optimization [20, 21].

4.3 Practical implications for industry

The results demonstrate that hybrid optimization frameworks are a viable solution for organizations dealing with demand uncertainty and complex operational constraints. The increased robustness and stability of the results across a range of scenarios suggest that these frameworks can be used at both the strategic and tactical levels without the need to customize them to suit a particular industry.

The above findings also highlight the importance of the implementation approach. While technical and operational deployments can be performed efficiently through modular and phased approaches, organizational adaptation is an important factor in determining success [22]. These findings support the need to integrate technical innovation with change management practices to maximize the benefits of optimization systems.

4.4 Computational complexity and scalability

The computational complexity of the proposed MILP model is exponential in the worst case, which is typical of Nondeterministic Polynomial time (NP)-hard optimization problems. However, the actual improvement is substantial with the implementation of advanced optimization solvers, such as Gurobi. The machine learning component has a computational complexity of $O(n \log n)$, which is beneficial from the viewpoint of forecasting.

4.5 Limitations and future work

Although it has numerous benefits, the proposed framework

has some limitations. The assessment of the framework is performed using simulation scenarios and assumes cost parameters to be deterministic. Furthermore, the computational complexity can potentially increase with very large-scale networks and high-dimensional forecasting models.

Future work may include improving the scalability of the proposed approach using techniques such as distributed optimization. Improving the ability to interface with existing systems. We extend the proposed approach to include more criteria, such as sustainability and risk. Further validation using industrial data.

5. CONCLUSIONS

The proposed approach describes a hybrid optimization approach that combines mathematical optimization and machine learning to solve supply chain planning problems in environments with uncertain demand. This approach combines predictive and prescriptive models in a manner that improves planning resilience without compromising the clarity of mathematical models.

The experimental results demonstrate that the hybrid method outperforms traditional optimization and machine learning techniques individually. This is because the system has demonstrated greater adaptability to changes in demand, better synchronization between production, inventory, and transportation, and greater stability in planning, even when faced with uncertainty. This proves that there are potential advantages in using learning-based demand forecasting and optimization models to overcome the challenges associated with purely deterministic models.

The analysis of the results also indicates that the improvement in performance achieved by the proposed framework is not limited to a given application domain. Scenario-based analyses, which represent different industrial environments, indicate that the framework is flexible and can be applied to different supply chain configurations without the need for major modifications. This flexibility is a testament to the effectiveness of hybrid optimization models in addressing industrial problems.

An implementation analysis suggests that although the implementation of the framework with respect to technology and process is possible through a modular and incremental approach, the organizational implementation becomes a crucial factor that affects its success. Therefore, it is important to consider the integration of analytical techniques with the change management and training process.

Nevertheless, it should be noted that the framework has some limitations. Firstly, the computational complexity in hybrid models poses a problem when applied to large network topologies. Secondly, the simulation-based evaluation implies that the validation process will require the use of real-world data from enterprises. Thirdly, the integration with legacy systems will require customized approaches based on the particular infrastructure.

In terms of future development, the main directions include the improvement of scalability using distributed and parallel optimization algorithms, enhancing interoperability between the system and existing enterprise information systems, creating automated techniques for the determination of model parameters, and developing new models taking into account different objectives.

Regarding the mathematical modeling approach, it can be stated that it provides the combination of predictive learning models and prescriptive optimization models. Such an approach allows us to increase the robustness of the obtained solution while preserving its mathematical optimality properties.

In summary, this study has made its own contribution towards enhancing supply chain optimization techniques as it has shown that there is a viable and effective method of dealing with complexities and uncertainties using a hybrid of mathematical programming and machine learning techniques. The proposed method provides a strong foundation for future developments of decision-support systems in industry.

REFERENCES

- [1] Alkahtani, M. (2022). Mathematical modelling of inventory and process outsourcing for optimization of supply chain management. *Mathematics*, 10(7): 1142. <https://doi.org/10.3390/math10071142>
- [2] Bo, L.X., Xu, J. (2024). Enhancing supply chain efficiency resilience using predictive analytics and computational intelligence techniques. *IEEE Access*, 12: 183451-183465. <https://doi.org/10.1109/ACCESS.2024.3507161>
- [3] Camelo, M.M., de Andrade, C.F., de Athayde Prata, B. (2025). A mixed-integer linear programming model for optimizing green hydrogen supply chain networks. *International Journal of Hydrogen Energy*, 118: 134-145. <https://doi.org/10.1016/j.ijhydene.2025.02.138>
- [4] Chhetri, P., Hashemi, A., Lau, K.H., Lim, M.K. (2022). Aligning supply chain complexity with product demand and design characteristics. *International Journal of Logistics Research and Applications*, 25(8): 1137-1163. <https://doi.org/10.1080/13675567.2021.1885020>
- [5] Dalal, S., Lilhore, U.K., Simaiya, S., Radulescu, M., Belascu, L. (2024). Improving efficiency and sustainability via supply chain optimization through CNNs and BiLSTM. *Technological Forecasting and Social Change*, 209: 123841. <https://doi.org/10.1016/j.techfore.2024.123841>
- [6] Feizabadi, J. (2022). Machine learning demand forecasting and supply chain performance. *International Journal of Logistics Research and Applications*, 25(2): 119-142. <https://doi.org/10.1080/13675567.2020.1803246>
- [7] Fu, Q.H., Abdul Rahman, A.A., Jiang, H., Abbas, J., Comite, U. (2022). Sustainable supply chain and business performance: The impact of strategy, network design, information systems, and organizational structure. *Sustainability*, 14(3): 1080. <https://doi.org/10.3390/su14031080>
- [8] Goli, A. (2023). Integration of blockchain-enabled closed-loop supply chain and robust product portfolio design. *Computers & Industrial Engineering*, 179: 109211. <https://doi.org/10.1016/j.cie.2023.109211>
- [9] Iseri, F., Iseri, H., Chrisandina, N.J., Iakovou, E., Pistikopoulos, E.N. (2025). AI-based predictive analytics for enhancing data-driven supply chain optimization. *Journal of Global Optimization*, 1-28. <https://doi.org/10.1007/s10898-025-01509-1>
- [10] Kovari, A. (2024). AI for decision support: Balancing accuracy, transparency, and trust across sectors. *Information*, 15(11): 725. <https://doi.org/10.3390/info15110725>
- [11] Li, Y., Xu, M.R., Wang, H.B., Zhang, X.G., Chen, Y., Liu, Y.F., Liang, Y.L., Jiang, Y.J. (2024). Modeling of cross-network collaboration supply chain and solution via reinforcement learning-based evolutionary algorithm. *International Journal of Industrial Engineering: Theory, Applications and Practice*, 31(2). <https://doi.org/10.23055/ijietap.2024.31.2.9523>
- [12] Nafi, Z., Essaber, F.E., Elharouni, F., Rachid, B. (2023). An approach based on machine learning and discrete event simulation for supply chain optimization: The case of on-stock chains. *International Journal of Industrial Engineering: Theory, Applications and Practice*, 30(6): 1558-1577. <https://doi.org/10.23055/ijietap.2023.30.6.8613>
- [13] Pasupuleti, V., Thuraka, B., Kodete, C.S., Malisetty, S. (2024). Enhancing supply chain agility and sustainability through machine learning: Optimization techniques for logistics and inventory management. *Logistics*, 8(3): 73. <https://doi.org/10.3390/logistics8030073>
- [14] Rajabi-Kafshgar, A., Gholian-Jouybari, F., Seyedi, I., Hajiaghaei-Keshteli, M. (2023). Utilizing hybrid metaheuristic approach to design an agricultural closed-loop supply chain network. *Expert Systems with Applications*, 217: 119504. <https://doi.org/10.1016/j.eswa.2023.119504>
- [15] Ravindran, A.R., Warsing Jr, D.P., Griffin, P.M. (2023). *Supply Chain Engineering: Models and Applications*. CRC Press. <https://doi.org/10.1201/9781003283393>
- [16] Rolf, B., Jackson, I., Müller, M., Lang, S., Reggelin, T., Ivanov, D. (2023). A review on reinforcement learning algorithms and applications in supply chain management. *International Journal of Production Research*, 61(20): 7151-7179. <https://doi.org/10.1080/00207543.2022.2140221>
- [17] Sana, S.S. (2022). A structural mathematical model on two echelon supply chain system. *Annals of Operations Research*, 315: 1997-2025. <https://doi.org/10.1007/s10479-020-03895-z>
- [18] Teixeira, E.D.S., Rangel, S., Florentino, H.D.O., de Araujo, S.A. (2023). A review of mathematical optimization models applied to the sugarcane supply chain. *International Transactions in Operational Research*, 30(4): 1755-1788. <https://doi.org/10.1111/itor.13056>
- [19] Vicente, J.J. (2025). Optimizing supply chain inventory: A mixed integer linear programming approach. *Systems*, 13(1): 33. <https://doi.org/10.3390/systems13010033>
- [20] Yu, X.M., Tang, L., Long, L., Sina, M. (2024). Comparison of deep and conventional machine learning models for prediction of one supply chain management distribution cost. *Scientific Reports*, 14: 24195. <https://doi.org/10.1038/s41598-024-75114-9>
- [21] Zhang, Y.R., Niu, Z.W., Liu, C., Liu, C.C., Hou, K.X. (2024). System dynamics analysis for impact of cloud-based information-sharing on supply chain performance. *International Journal of Industrial Engineering: Theory, Applications and Practice*, 31(2). <https://doi.org/10.23055/ijietap.2024.31.2.9693>
- [22] Zohdi, M., Rafiee, M., Kayvanfar, V., Salamiraad, A. (2022). Demand forecasting based machine learning algorithms on customer information: An applied approach. *International Journal of Information*

NOMENCLATURE

Z	total supply chain cost, USD
P	production quantity, units
X	shipment quantity, units
I	inventory level, units
D	customer demand, units
C_p	production cost, USD.unit ⁻¹
C_t	transportation cost, USD.unit ⁻¹
C_h	inventory holding cost, USD.unit ⁻¹
Cap	transportation capacity, units
Y	binary decision variable (transport activation)
w	adaptive weight for hybrid integration, dimensionless
S	solution variable (hybrid / ML / MILP), dimensionless
E	error term (prediction or optimization), dimensionless
n	number of demand observations
MAPE	mean absolute percentage error, %
i	production node index

Greek symbols

α	learning rate parameter, dimensionless
β	weighting coefficient in prediction model, dimensionless
γ	adaptive weight update rate, dimensionless
σ	standard deviation of prediction error, units
ε	convergence tolerance parameter, dimensionless

Subscripts

i	production node
j	demand node
t	time period
pred	predicted value
opt	optimized value
init	initial value
max	maximum value
ML	machine learning component
MILP	mixed-integer linear programming component
hybrid	hybrid solution