

Supply Chain Resilience Optimization Integrating Transformers and Multi-Agent Reinforcement Learning

Zhaocheng Wang^{1*}

¹Yangzhou Polytechnic University, Yangzhou, Jiangsu, China

Abstract

INTRODUCTION: Manufacturing supply chains face increasing vulnerabilities from delayed risk identification, conflicting node-level decisions, and escalating recovery costs, particularly under complex disruption scenarios. Existing approaches often lack proactive, coordinated responses across autonomous supply chain nodes, limiting their ability to anticipate disruptions and synchronize mitigation actions effectively. To address these gaps, this paper proposes an integrated resilience optimization framework that combines predictive analytics with multi-agent coordination.

OBJECTIVES: The primary objective is to develop a supply chain resilience framework, termed T-MARL, that enables autonomous suppliers, manufacturers, warehousing nodes, and distributors to: (1) predict future disruption probabilities using temporal risk precursor data, and (2) execute coordinated, proactive replenishment, production, and distribution decisions under inventory and capacity constraints. The framework aims to improve resilience index, reduce total operating costs, and shorten recovery time compared to baseline strategies, while also examining the sensitivity of performance gains to prediction noise and communication latency.

METHODS: The T-MARL framework models each supply chain node as an independent agent. A lightweight Transformer captures temporal features from 6-dimensional risk precursor sequences over 12 consecutive periods to forecast disruption probabilities. These predictions feed into a value-decomposition multi-agent reinforcement learning mechanism with centralized training–decentralized execution, where a value-decomposition network orchestrates cross-node actions. Risk-constrained action gating ensures that proactive responses adhere to inventory and capacity limits. A digital twin simulation environment was built with fixed random seeds, covering three scenarios—single-source supply disruption, demand surge, and compound disruption. Five strategies were tested, with 30 runs per unit using common random number pairing.

RESULTS: On the trajectory-held-out test set, the Transformer achieved an AUC of 0.722 and an F1-score of 0.586, while Random Forest achieved a slightly higher AUC of 0.724. Under the compound disruption scenario, T-MARL attained a resilience index of 0.849—a 1.63 percentage-point gain over VDN-MARL—reduced total cost by 14.7%, and shortened average recovery time from 1.67 periods to zero. T-MARL also achieved lower operating costs while maintaining service levels comparable to rule-based strategies. However, the paper also found that prediction noise and communication latency diminish these performance gains.

CONCLUSION: This research contributes a reproducible digital twin analysis framework for disruption prediction, cross-node collaborative response, and rapid recovery in manufacturing supply chains. While T-MARL demonstrates significant improvements in resilience and cost-efficiency under ideal conditions, practical deployment must account for real-world imperfections such as noisy forecasts and delayed communications. The framework provides a scalable foundation for future work on adaptive, data-driven supply chain resilience.

Keywords: Supply chain resilience; Transformer; Multi-agent reinforcement learning; Disruption prediction; Digital twin; Adaptive recovery

Received on 25 August 2026, accepted on 20 September 2026, published on 05 October 2026

Copyright © 2026 Zhaocheng Wang, licensed to EAI. This is an open access article distributed under the terms of the CC BY-NC-SA 4.0, which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi: 10.4108/eetsis.14737

* Corresponding author. Email: 309678578@@qq.com

1. Introduction

While global manufacturing networks enhance resource allocation efficiency through specialization in division of labor, lean inventory practices, and cross-regional collaboration, they also couple risks across supply, production, warehousing, and distribution nodes. While global manufacturing networks improve resource-allocation efficiency through specialization, lean inventories, and cross-regional collaboration, they also propagate disturbances across supply, production, warehousing, and distribution nodes. A supplier shutdown, demand surge, or transportation delay can therefore generate ripple effects that cause inventory imbalance, degraded delivery performance, and higher recovery costs [1]. Supply-chain resilience extends beyond rapid post-disruption recovery and includes proactive preparation, shock absorption, dynamic recovery, and structural adaptation [2]. Conventional resilience measures such as safety stock, backup capacity, and multi-sourcing can improve buffering capability, but they also increase inventory, switching, and expediting costs. Maintaining resilience without excessive redundancy therefore remains a central operational challenge.

Advancements in digital twins and artificial intelligence lay down the technological groundwork towards shifting from reactive response to predictive decision making in supply chain resilience management. Digital supply chain twins enable dynamic mapping of network states, inject disruption scenarios, compare recovery strategies which provides a virtual space for risk stress test, low-cost training and validate strategy [3] whereas machine learning is also being used to identify suppliers' risks, forecast demands, control inventory deviations and allocate resources [4]. But current research tends to separate risk prediction and operational optimization. Predictive models focus on metrics like AUC, accuracy, and F1, while scheduling models assume complete observability of risk states leading to no auditable interface between predicting probabilities and particular replenishment, capacity, and distribution actions [5]. Thus, improving offline prediction does not guarantee better resilience, faster recovery times, or lower operational costs.

Manufacturing disruption formation usually involves multivariable, nonlinear, and intertemporal accumulation dynamics where factors including supply lead-time drift, decreasing fulfillment rates, transportation anomalies, fluctuating demands, and changes in equipment utilization are all potential early warning signs at various time scales. The use of self-attention mechanism modeling long-range dependency and variable interaction in parallel makes Transformer models a good choice to identify patterns connecting early weak signals to accumulation of near-term risks [6-7], but current time series research mainly emphasizes prediction error or ranking performance while neglecting subsequent impacts from false positives and false negatives on inventory, capacity and recovery strategies [8]. Also, with small-to-medium-scale datasets, Transformer does not always perform better than traditional models (e.g., Random Forest). Their practicality needs to be evaluated based on their marginal contribution to operational decisions

instead of being solely determined by one predictive metric [9].

Supply chains involve multiple agents with partial observability and coupled objectives: suppliers, manufacturers, warehousing nodes, and distributors all have their own local knowledge about inventory, backlogs, and capacity; optimizing independently at such nodes inevitably results in redundant stocking, order amplification, or canceling upstream/downstream actions[10]; multi-agent reinforcement learning (MARL), which learns cross-node coordination strategies under a "centralized training, decentralized execution" paradigm, has recently been used for decentralized inventory control, multi-echelon replenishment, and resource allocation in complex supply networks [11-12]. Although research has shown the potential of using multi-agent approaches to address information boundaries and dynamic uncertainties, existing works mostly consider standard demand fluctuations or average inventory costs without incorporating lead-time risk prediction, cross-level communication, action feasibility constraints, and post-disruption recovery as part of an integrated decision-making loop; furthermore, fully end-to-end black-box strategies could produce actions violating constraints over inventory, capacity, or management authority.

Transformer architectures were originally developed to replace recurrent sequence processing with self-attention, allowing dependencies between distant positions to be modeled directly and in parallel [13]. Subsequent studies have demonstrated their utility for multivariate and multi-horizon time-series representation because attention mechanisms can capture interactions among heterogeneous variables over different temporal scales [14–15]. For supply-chain disruption warning, this capability is particularly relevant because lead-time changes, fulfillment deterioration, transportation anomalies, demand variation, and capacity utilization may interact before an interruption becomes directly observable. However, the operational relevance of a Transformer should be assessed by how its probability output influences downstream decisions, rather than by prediction metrics alone.

MARL addresses a different but complementary problem. In decentralized supply networks, individual firms or echelons possess only partial information while their decisions jointly determine system-level inventory, service, and cost. CTDE methods allow global information and team rewards to be exploited during training while preserving decentralized execution. Value decomposition further represents the joint action value as a combination of local action values and provides a practical mechanism for cooperative learning under a shared reward [16]. Recent supply-chain applications have demonstrated the promise of decentralized MARL for inventory control, but prediction, constraint handling, and post-disruption recovery are still frequently treated as separate components. The present study focuses specifically on this interface between temporal risk prediction and coordinated constrained action.

To summarize, there exist four main drawbacks in current research [17]. Firstly, risk prediction and coordinated control are developed independently without any means to convert

probabilistic early warning into constrained, proactive actions; secondly, centralization optimization cannot handle information boundaries among enterprises while independent decision-making does not consider cross-level externalities concerning inventory and service behaviors [18]; thirdly, algorithms are evaluated mainly from aspects like prediction accuracy, cumulative returns, or average costs but seldom combine resilience indices, service levels, recovery times, and cost variations together; fourthly, how prediction noise, communication latency, and simulation-to-reality distribution shift affect the applicability limits of integrated strategies have not been sufficiently discussed [19]. Recent review papers show that research on artificial intelligence within supply chain resilience is still fragmented with an urgent need for clarifying the connection between predictive analytics and prescriptive decision-making [20].

Existing studies can be grouped into three related methodological streams. First, forecast-driven inventory-control methods introduce future demand or lead-time information into sequential replenishment decisions, but they generally optimize a single controller or do not explicitly address decentralized cross-echelon coordination. Second, MARL-based inventory studies coordinate multiple autonomous nodes under partial observability, yet most policies remain reactive to the current inventory and demand states rather than being triggered by an independently auditable disruption probability. Third, constrained-RL methods explicitly restrict unsafe or infeasible actions, but constraint handling is typically embedded within the learning algorithm rather than serving as an operational interface between probabilistic warning and supply-chain execution. Recent prediction-enhanced MARL studies have begun integrating forecasting and distributed decision making, but an explicit mechanism linking calibrated disruption probability, value-decomposed coordination, physical action feasibility, and post-disruption recovery remains insufficiently examined.

Accordingly, the novelty claimed in this study does not arise from the Transformer or VDN individually. Instead, it lies in the auditable coupling of four functions: disruption-probability estimation, decentralized value-decomposition coordination, risk-triggered feasibility gating, and recovery-oriented evaluation. This distinction is important because a more accurate predictor does not necessarily generate a better operational policy. We therefore evaluate predictor value not only through AUC and F1-score, but also by feeding calibrated Random Forest, MLP, and Transformer probabilities into the same VDN-gating backbone.

In response to these challenges in the above-mentioned issues, this paper proposes T-MARL: A framework of optimizing supply chain resilience by integrating a lightweight Transformer with value-decomposition multi-agent reinforcement learning.

This paper treats suppliers, manufacturers, warehousing nodes, and distributors as autonomous agents, where a Transformer extracts risk representations based on 6-dimensional precursor sequences over 12 consecutive periods and predicts disruption probabilities.

Under a centralized training and decentralized execution paradigm, it learns cross-level collaborative behavior using value decomposition and uses a risk-constrained action gating mechanism to convert the predicted probability to proactively adjust replenishments and capacities within inventory and capacity limits.

Using a digital twin with fixed random seeds, this paper evaluates three types of rule-based, independent, and collaborative learning methods (with common random number pairing), across different scenarios including single-source supply disruptions, demand surges, and compound disruptions, reporting metrics such as resilience, service level, cost, and recovery time. Through component ablation and sensitivity analysis about prediction noise and communication latency, this paper identifies the individual contribution or failure condition related to prediction, value decomposition, and neighborhood communication. The purpose of our work is not to extrapolate simulation results onto actual corporate operational performance but rather to offer a verifiable mechanism and experimental template encompassing "perception, prediction, coordination, execution, and feedback".

2. Resilience Optimization Model

2.1. Four-Echelon Supply Chain and Basic Assumptions

This paper considers a single-product, four-echelon serial supply chain system comprising a supplier, a manufacturer, a warehousing node, and a distributor, designated as agents $i \in \{1, 2, 3, 4\}$. In each operating period, an agent first receives in-transit goods and fulfills end-customer demand and downstream orders, subsequently determining the intensity of replenishment, production, or distribution. Only local inventory, backlogs, risk buckets, and a neighborhood summary is observed for each agent; team rewards are shared during centralized training, but full value functions are never exchanged during decentralized execution phase. The transportation lead time is captured via queues; supply disruption is captured with reduced capacity factors; demand surge is captured with end-demand multipliers; compound disruptions add another period of transportation delay. Notation key is given in Table 1.

Table 1. Key symbols and definitions

Symbol	Definitions
i, t	Supply chain agent and discrete time step index
$S_{i,t}$	Local state of agent i at time t
$a_{i,t}$	Ordering, capacity, or distribution action
$p_t^{(h)}$	Disruption probability over a horizon of h periods as predicted by the Transformer
$I_{i,t}$	Available inventory at node i
$B_{i,t}$	Backlog at node i
Q_{tot}	Team action value derived via value decomposition

R_t	Team reward comprising service, inventory, backlog, and expediting costs
RI	Resilience index synthesized from service loss and backlog penalties

2.2. State Transitions, Disruption Processes, and Action Space

Baseline demand of 20 units per period plus a 13-period seasonal component and normal random disturbance. A demand surge multiplies demand by 1.55 within the disruption window; a single-source disruption reduces upstream capacity by 70% for 5 periods; and a compound disruption reduces capacity by 78% for 7 periods, while amplifying demand by a factor of 1.45 and adding a one-period transportation delay. Let $x_{i,t}$ be the actual shipment quantity from node i in the current period, and $y_{i-1,t-L_i}$ be the quantity arriving from upstream; inventory and backlogs are then updated according to Equation (1). The superscript “+” in brackets denotes the non-negative part.

$$\begin{aligned} I_{i,t+1} &= [I_{i,t} + y_{i-1,t-L_i} - x_{i,t}]^+, \\ B_{i,t+1} &= [B_{i,t} + d_{i,t} - x_{i,t}]^+ \end{aligned} \quad (1)$$

Transformer probabilities are not treated as direct substitutes for the environmental state. Instead, they enter a separate deployment-layer gating mechanism. This design separates the learned coordination policy from the auditable protection rule and permits the probability and inventory thresholds to be calibrated independently.

2.3. Objective Function and Evaluation Metrics

Instantaneous cost includes inventory holding costs, backlogs at both terminal and echelon level, expedited action costs, and penalty on fluctuation orders. Reinforcement learning reward takes a negative scaled version of cost given in Equation (2), which encourages agent not only accumulate too much inventory but also causes stockout due to too conservative strategy. Discount factor is taken as $\gamma = 0.96$.

$$R_t = -(0.16\Sigma_i I_{i,t} + 3.60B^{(c)}_t + 0.38\Sigma_i B_{i,t} + 1.35N^{\text{exp}}_t + 0.05\|q_t - q_{t-1}\|_1)/100 \quad (2)$$

The coefficients in Equation (2) are dimensionless simulation weights rather than empirically estimated monetary cost rates. They were selected to impose an explicit operational priority structure. The terminal customer backlog receives the largest penalty (3.60) because an unserved downstream order directly represents service failure. Expedited actions receive the second-largest penalty (1.35) to discourage excessive emergency interventions. Intermediate-echelon backlog (0.38) is penalized more strongly than ordinary inventory holding (0.16), reflecting the greater operational consequence of an unmet material requirement

relative to temporary excess stock. The order-variation term receives the smallest coefficient (0.05) and serves primarily as a regularizer against unnecessarily volatile replenishment decisions. The division by 100 rescales the team reward and does not change this relative priority structure. Consequently, the reported total cost values should be interpreted as simulation cost indices for within-study comparison, rather than direct monetary estimates. Enterprise deployment would require recalibration of these coefficients using organization-specific holding, shortage, expediting, and adjustment costs.

The Resilience Index (RI) is defined as the average service level over 52 periods minus a normalized penalty for the average terminal backlog, truncated to the range $[0, 1]$; its expression is provided below. The recovery time is measured from the end of the disruption window: recovery is considered complete when the service level remains at or above 0.95 for three consecutive periods and the terminal backlog does not exceed 18% of the baseline demand. Recovery time is measured from the end of the externally imposed disruption window. Recovery is declared when the service level is at least 0.95 for three consecutive periods and the terminal backlog is no greater than 18% of baseline demand. If both criteria are already satisfied at the end of the disruption window, the recorded recovery time is defined as zero periods. This value does not imply instantaneous physical restoration of the supply network. It means only that no additional post-disruption period is required to restore the predefined service and backlog criteria.

$$RI = \text{clip}(\text{mean}_t(\text{Service}_t) - 0.0018 \cdot \text{mean}_t(B^{(c)}_t), 0, 1) \quad (3)$$

2.4. MDP Formulation and Feasibility-Constrained Decision Process

The operational optimization problem is formulated as a finite-horizon partially observable Markov decision process with four decentralized agents. At period t , agent i observes the local state

$$\mathbf{s}_{i,t} = \left[\frac{I_{i,t}}{I_i^*}, \frac{B_{i,t}}{\bar{d}}, \frac{q_{i,t}^{\text{pipe}}}{\bar{d}}, r_t, \bar{I}_{N(i),t}, \bar{B}_{N(i),t} \right] \quad (4)$$

where $I_{i,t}$ and $B_{i,t}$ denote local inventory and backlog, I_i^* is the target inventory, $q_{i,t}^{\text{pipe}}$ is the quantity currently in the transportation pipeline, r_t is the discretized disruption-risk level obtained from the calibrated predictor, and $\bar{I}_{N(i),t}$ and $\bar{B}_{N(i),t}$ summarize the inventory and backlog conditions of directly connected neighboring nodes.

Each agent selects a discrete replenishment or capacity multiplier from

$$\mathcal{A}_i = \{0.78, 1.00, 1.30\} \quad (5)$$

The multiplier is applied to the baseline operating quantity and subsequently projected onto the node-specific feasible set

determined by non-negativity, available inventory, transportation feasibility, and capacity constraints. Inventory and backlog evolve according to

$$I_{i,t+1} = [I_{i,t} + y_{i-1,t-L_i} - x_{i,t}]^+ \quad (6)$$

$$B_{i,t+1} = [B_{i,t} + d_{i,t} - x_{i,t}]^+ \quad (7)$$

where $y_{i-1,t-L_i}$ is the upstream quantity arriving after lead time L_i , $x_{i,t}$ is the quantity fulfilled by node i , and $[\cdot]^+$ denotes $\max(\cdot, 0)$.

The period service level is calculated as

$$Service_t = \min \left[1, \frac{x_{4,t}}{d_t + B_{4,t} + \varepsilon} \right] \quad (8)$$

where node 4 is the customer-facing distributor and ε is a small numerical constant preventing division by zero.

The instantaneous team reward is

$$R_t = -(0.16 \sum_i I_{i,t} + 3.60 B_t^{(c)} + 0.38 \sum_i B_{i,t} + 1.35 N_t^{exp} + 0.05 \|q_t - q_{t-1}\|_1) / 100 \quad (9)$$

The resilience index is

$$RI = clip \left[\frac{1}{T} \sum_{t=1}^T Service_t - 0.0018 \frac{1}{T} \sum_{t=1}^T B_t^{(c)}, 0, 1 \right] \quad (10)$$

Under value decomposition,

$$Q_{tot}(\mathbf{s}_t, \mathbf{a}_t) = \sum_{i=1}^4 Q_i(\mathbf{s}_{i,t}, \mathbf{a}_{i,t}) \quad (11)$$

The tabular temporal-difference error is

$$\delta_t = R_t + \gamma \max_{a'} Q_{tot}(\mathbf{s}_{t+1}, a') - Q_{tot}(\mathbf{s}_t, \mathbf{a}_t) \quad (12)$$

and each local value table is updated using the common team TD error:

$$Q_i(\mathbf{s}_{i,t}, \mathbf{a}_{i,t}) \leftarrow Q_i(\mathbf{s}_{i,t}, \mathbf{a}_{i,t}) + \alpha \delta_t \quad (13)$$

This additive update is consistent with the VDN formulation, in which decentralized local action values are optimized against a shared team return.

3. Transformer-Based Multi-Agent Reinforcement Learning Integration Method

3.1. Overall Architecture

Six modules of our architecture include: multi-source precursor sequences; Transformer risk encoder; digital twin environment; value decomposition coordination backbone; risk constrained action gating mechanism; rolling feedback. A prediction module predicts with probability if there is a disruption in the next H periods; a decision module decides an action level at every node given its local inventory and backlog condition; and a gating module avoids adjacent echelon choosing low replenishment actions when it's risky. An illustration of information and decisions can be seen in Figure 1.

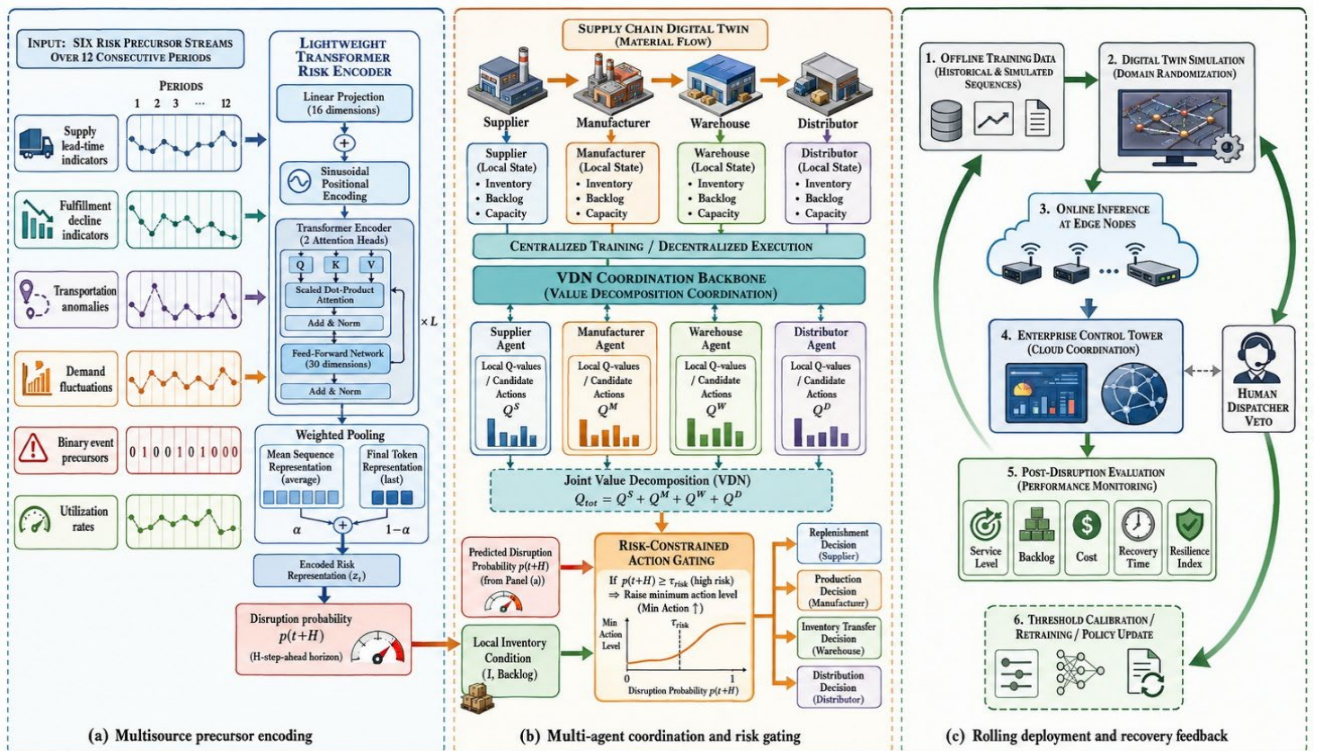


Figure 1. Transformer–multi-agent reinforcement learning fusion architecture

3.2. Lightweight Transformer-Based Disruption Predictor

Each prediction sample contains six precursor variables observed over a 12-period look-back window. The six variables are supply lead-time indicators, fulfillment-decline indicators, transportation anomalies, demand fluctuations, binary event precursors, and utilization rates [21]. A total of 120 independent latent-risk trajectories were generated, with each trajectory containing 74 operating periods. A 12-period sliding window with a stride of one period was then applied within each trajectory, producing 60 samples per trajectory and 7,200 samples in total. Windows were never allowed to cross trajectory boundaries.

if a disruption onset occurred during periods $t+1$ to $t+3$, and $y_t=0$ otherwise. Thus, the main forecasting horizon was $\setminus(H=3\setminus)$ periods. In the multi-lead-time stress test, the look-back window remained fixed at 12 periods while the forecast horizon was varied from $H=1$ to $H=6$. Feature normalization parameters were estimated exclusively from the training trajectories and then applied unchanged to the validation and test sets.

$$A^h = \text{softmax}\left(\frac{Q^{hK^{ht}}}{\sqrt{d}}\right) V^h, \bar{h}_t = 0.45 \cdot \text{mean}(H_t) + 0.55 \cdot H_{t, \text{last}} \quad (14)$$

$$\hat{p}_t^{(H)} = \sigma(w_o^T \bar{h}_t + b_o) \quad (15)$$

The predictor was trained for 58 epochs using weighted binary cross-entropy, Adam updates, gradient clipping, and early stopping based on validation set AUC. The primary task label indicates whether a disruption occurs within the next three periods; the multi-lead-time stability experiment involved incrementally increasing label and observation uncertainty while maintaining the same risk score, in order to assess performance degradation across lead-time windows ranging from one to six periods [22]. As this extension constitutes a synthetic stress test rather than utilizing six independently collected sets of real-world labels, the results serve only to identify trends and should not be used to claim predictive accuracy across industries.

3.3. Multi-Agent Coordination via Value Decomposition

VDN expresses the joint action-value as the sum of the local action-values of individual agents, as shown in Equation (6). During training, a unified team reward is used to update the

four local Q-tables, whereas during execution, each agent selects actions based solely on its local state. Compared to independent Q-learning, value decomposition incorporates cross-level externalities—stemming from inventory, backlogs, and order fluctuations—into a single temporal-difference error; compared to fully centralized control, it does not require the continuous uploading of all states during execution [23]. This paper employs tabular VDN rather than deep networks to ensure that performance variations can be attributed to the prediction, communication, and value decomposition mechanisms, rather than to network scale.

$$Q_{\text{tot}}(s_t, a_t) = \sum_{i=1}^4 Q_i(s_{i,t}, a_{i,t}) \quad (16)$$

Training employed a curriculum-based initialization strategy: VDN-MARL was first trained for 1,400 episodes in a randomized-interruption environment to learn a backbone for inventory-backlog coordination that does not rely on predictions; subsequently, the Q-values were replicated, and T-MARL fine-tuning was performed for 750 episodes using a lower learning rate [24–25]. The exploration rate decayed according to a power law, with a lower bound of 0.04. To ensure a fair comparison with independent learning and communication ablation studies, IQL was trained for 1,150 episodes, and VDN without communication was trained for 1,300 episodes. Training returns are presented using a 45-episode moving average.

To distinguish algorithmic improvement from additional environment interaction, we introduced a matched-budget fairness experiment. All compared learning policies used identical initialization rules, training disruption streams, stopping criteria, and training seeds. For this experiment, the total interaction budget was fixed at 2,150 episodes, corresponding to the 1,400-episode coordination stage plus the 750-episode refinement stage used by T-MARL. Two additional controls were introduced. VDN-continued starts from the same 1,400-episode VDN backbone and continues training for another 750 episodes without predictive gating. T-MARL-frozen uses the 1,400-episode VDN backbone without updating its Q-tables during deployment; only the calibrated predictive gating mechanism is applied. The full T-MARL model uses both gating and the 750-episode prediction-augmented refinement stage. All three variants are evaluated on identical paired disruption trajectories.

3.4. Predictive Risk Gating and Safety Constraints

When the predicted probability $\hat{p}_t$ is below 0.68, T-MARL executes the action prescribed by the VDN backbone; when $\hat{p}_t$ enters the high-risk range and node inventory falls below 1.22 times the target inventory, the gating mechanism raises the minimum action level: the first three nodes must select at least the "high" replenishment tier, while customer-facing

distribution nodes must select at least the "baseline" tier [26]. Equation (17) presents the abstract formulation, where g_i represents the minimum actionable level under risk conditions, and Π_a denotes the projection onto the action set and capacity constraints. Rather than relying on hard-coded "disruption awareness" after the fact, the gating mechanism operates solely based on predicted probabilities; since false positives increase inventory costs and false negatives compromise recovery, the threshold must be calibrated within a validation environment.

Let

$$a_{i,t}^{VDN} = \arg \max_{a \in \mathcal{A}_i^{feas}(s_t)} Q_i(s_{i,t}, a) \quad (17)$$

denote the feasible action preferred by the coordination backbone. The validation-calibrated risk threshold and inventory threshold are denoted by $\tau_p = 0.68$ and $\tau_I = 1.22$, respectively. When

$$\hat{p}_t^{(H)} \geq \tau_p \text{ and } \frac{I_{i,t}}{I_i^*} < \tau_I \quad (18)$$

the gating lower bound is

$$g_i = \begin{cases} 1.30, & i = 1, 2, 3, \\ 1.00, & i = 4. \end{cases} \quad (19)$$

Otherwise, $g_i = 0.78$. The action executed is

$$\tilde{a}_{i,t} = \Pi_{\mathcal{A}_i^{feas}}[\max(a_{i,t}^{VDN}, g_i)] \quad (20)$$

where $\Pi_{\mathcal{A}_i^{feas}}$ projects the proposed action onto the discrete set satisfying the node capacity, inventory, and transportation constraints. Consequently, infeasible proposals cannot be executed. Across the three evaluation scenarios, the post-projection physical-constraint violation rate was therefore 0.00% (0 violations among 18,720 executed agent-period decisions).

Both gating parameters were calibrated exclusively on the validation environment and were fixed before test-scenario evaluation. Specifically, the disruption-probability threshold τ_p was searched from 0.50 to 0.80 and the inventory multiplier τ_I from 1.00 to 1.30 using increments of 0.02. Each candidate pair was evaluated on the same validation disruption trajectories using common random numbers. Parameter selection followed a lexicographic criterion: the candidate with the highest mean resilience index was preferred, and when two settings differed in resilience by less than 0.002, the setting with the lower total operating cost was selected. This procedure produced $\tau_p = 0.68$ and $\tau_I = 1.22$. Importantly, these parameters were calibrated once on a mixed validation set containing supply, demand, and compound disturbances and were subsequently held constant across all three test scenarios. They are therefore not scenario-specific parameters.

3.5 Train-Deploy-Recover Closed-Loop

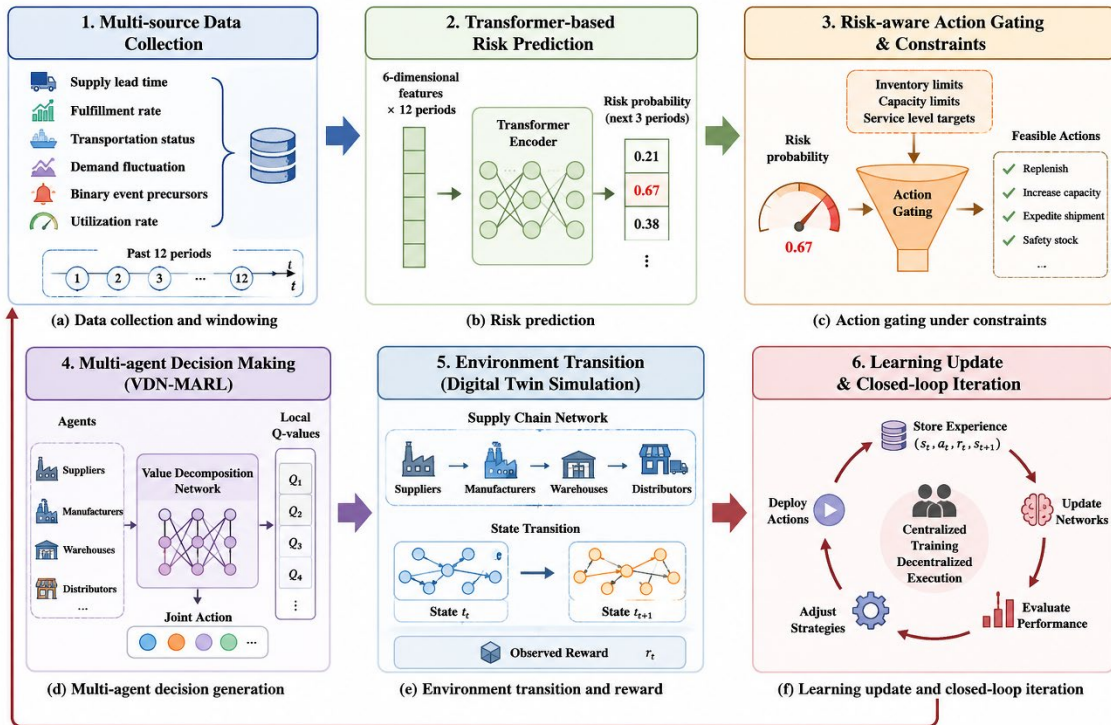


Figure 2. Closed-loop workflow for interrupt response and recovery

In the offline stage, this paper use historical/simulated sequences for training our predictor; in the digital twin, this paper generate policy experiences with domain randomization[27]; during the online stage, edge nodes perform feature aggregation and execute local actions whereas the cloud/enterprise control tower updates its risk encoder and coordination policies; in the post-disruption phase, it records service levels, backlogs, costs, and recovery times which triggers a re-training or threshold calibration. The human dispatchers have veto power on high impact actions to take into account constraints of compliance, quality, and supplier relations that cannot be captured by the model. Figure 2 shows the rolling closed-loop.

4. Experimental Design

4.1. Synthetic Data, Sample Partitioning, and Reproducibility

All computational experiments were conducted in a unified Python-based simulation environment on the same local workstation to ensure consistency across predictor training, multi-agent policy learning, ablation experiments, and final evaluation. The Transformer predictor, digital-twin supply chain simulator, and multi-agent decision modules were executed under identical software and numerical settings. Fixed random seeds were used for data generation, model initialization, disruption generation, and evaluation. For comparative experiments, all strategies were exposed to the same disruption trajectories and paired random-number streams. Data preprocessing, model training, statistical analysis, and visualization were performed within the same experimental environment. Because the proposed framework employs a lightweight Transformer and tabular value-decomposition learning rather than large-scale deep reinforcement learning networks, no distributed computing or specialized high-performance computing platform was required. This unified configuration minimizes hardware- and environment-induced variation and improves the reproducibility and fairness of the comparative experiments.

For prediction experiments, the 120 independently generated trajectories were partitioned at the trajectory level before sliding-window extraction was used for model fitting and evaluation. Eighty trajectories were assigned to training, 20 to validation, and 20 to testing, yielding 4,800, 1,200, and 1,200 sequence samples, respectively. All windows originating from the same latent-risk trajectory remained in the same partition. This prevents temporally adjacent or overlapping windows from appearing simultaneously in training and evaluation sets. The partition was fixed for all prediction models, and identical trajectory IDs were used in Logistic Regression, Random Forest, MLP, and Transformer comparisons.

Operational Experiment: this paper used episodes of length 52; in the training phase, disruption onset (periods 12–31), duration (3–8 periods) severity, demand surge, and transportation delays were randomized. Three different

scenarios were fixed in the evaluation phase, and there were 30 paired replications per scenario $\times$ strategy combination.

Advantages: Transparent, reproducible, ablation friendly; Limitations: Does not include factors such as actual supplier heterogeneity, contractual constraints, quality related rework, carbon emissions, shared capacity between multiple products [28] Our Excel file contains 450 main experimental records, 150 ablation records, all statistical test results. See Table 2 for detailed parameters.

Table 2. Main parameters of the simulation experiment

Parameter	Symbol	Setting
Supply chain structure	Supplier—	4 agents, single product
	Manufacturer—	
	Warehousing—	
	Distributor	
Operational cycle	T	52 periods
Average demand	$\bar{d}$	20 units/period; includes seasonal and random fluctuations
Training episodes	N_{train}	IQL: 1,150; VDN: 1,400; VDN (no communication): 1,300; T-MARL fine-tuned for 750 episodes
Evaluation repetitions	N_{eval}	30 repeated runs per scenario-algorithm pair
Discount factor	gamma	0.96
Action set	A	{0.78, 1.00, 1.30} $\times$ baseline replenishment quantity

4.2. Comparison Strategies and Ablation Settings

Table 3. Comparison methods and key characteristics

Method	Category	Key Characteristics
Base-stock	Static baseline inventory rules	Passive replenishment based solely on inventory position
Reactive	Event-triggered rules	Increasing replenishment or capacity in response to observed stockouts or backlogs
IQL	Independent Q-learning	Agents share team rewards but do not perform value decomposition or communication
VDN-MARL	Value-decomposition multi-agent learning	Uses current state and neighborhood information; excludes predictive representations
T-MARL	Proposed method	Transformer-based risk representation + value decomposition + neighborhood communication

This paper compares our approach with three types of methods: rule-based, independent learning, and cooperative learning. Our Base-stock method chooses actions according to inventory position; Reactive boosts replenishment and capacity when it observes disruptions/backlogs; IQL shares team rewards while updating independently; VDN-MARL

uses value decomposition and neighborhood information without look-ahead prediction; and T-MARL introduces a Transformer-based risk gating into the standard VDN backbone [29]. These comparisons allow us to separate out the incremental effects from learning, cooperation, and prediction. This paper performs ablation experiments removing the Transformer, value decomposition, and agent communication as presented in Table 3.

To isolate the operational contribution of the risk encoder, an additional matched predictor-to-control experiment was conducted. Logistic regression, Random Forest, MLP, and Transformer predictors were trained on exactly the same trajectory-level partitions. Their probability outputs were calibrated on the validation partition before deployment. During operational evaluation, the coordination backbone, action space, gating logic, disruption trajectories, random-number streams, and evaluation seeds were kept identical; only the calibrated predictor output supplied to the gating mechanism was changed. Consequently, differences in resilience, cost, and recovery can be attributed to the quality and temporal characteristics of the risk representation rather than to different downstream controllers.

4.3. Evaluation Metrics and Statistical Methods

Prediction layer: ROC-AUC, F1-score, precision, recall, and accuracy. Operations layer: Resilience index, average service level, total cost, recovery time, peak backlog, and inventory fluctuation [30]. Main tables show mean and standard deviation of the resilience index over 30 repetitions. Error bars in our figures correspond to 95% confidence intervals. This paper uses a one-sided Wilcoxon signed-rank test paired by repetition ID for statistical comparison testing an alternative hypothesis that the resilience index of T-MARL is greater than that of the baseline. With 12 intra-scenario comparisons This paper relies mostly on robust evidence ($p < 0.001$) when interpreting significance with both p-values and mean differences shown; if T-MARL has a lower mean it does not equate to equivalence given a "large p-value".

Cost-service analysis computes average over the three scenarios type and repetitions. A strategy is a Pareto improvement when it decreases costs but does not compromise service levels (or improves service levels

substantially whilst increasing cost by an acceptable amount). The analysis highlights rule-based strategies which use high inventory levels in order to maintain resilience and avoids reinforcement learning agents decreasing costs at the expense of customer service [31]. In sensitivity experiments this paper set the standard deviation of prediction noise to: 0, 0.08, 0.16, 0.24; and information delay to: 0 – 3 periods and run paired repetitions for every combination.

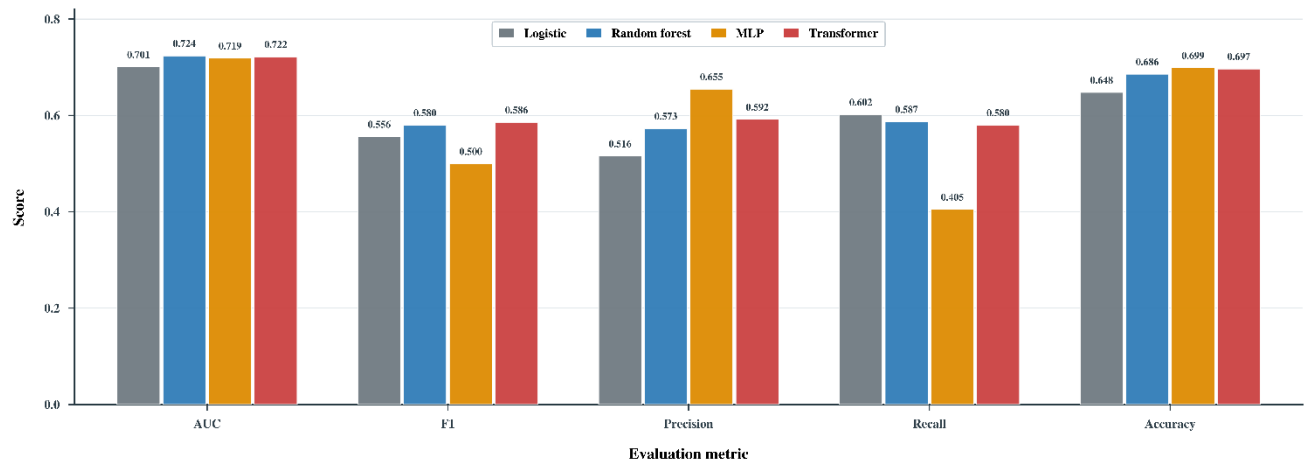
5. Experimental Results

5.1. Interruption Prediction Performance

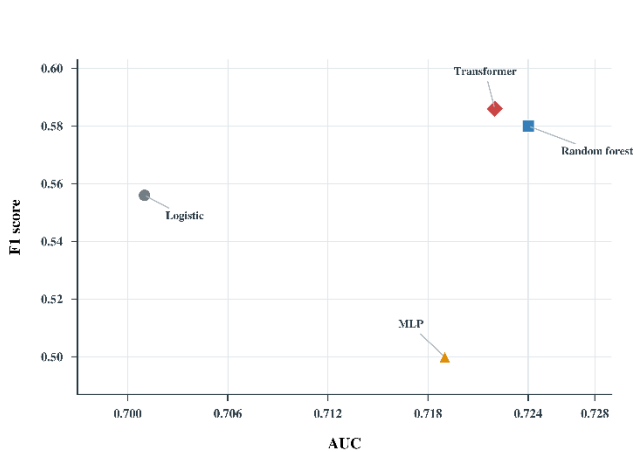
Table 4. Performance of the interruption prediction model on the independent test set

Model	AUC	F1	Precision	Recall	Accuracy
Logistic	0.701	0.556	0.516	0.602	0.648
Random Forest	0.724	0.580	0.573	0.587	0.686
MLP	0.719	0.500	0.655	0.405	0.699
Transformer	0.722	0.586	0.592	0.580	0.697

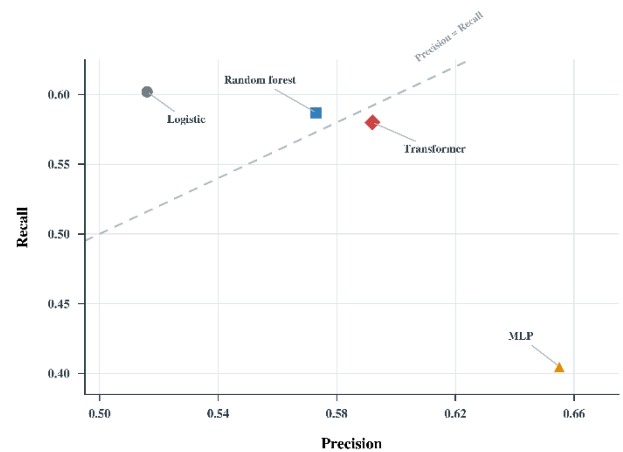
The Random Forest achieved the highest AUC of 0.724, followed by the Transformer (0.722), MLP (0.719), and Logistic Regression (0.701), as shown in Table 4. Although the Transformer did not achieve the highest AUC, its ranking performance was close to that of the Random Forest. As shown in Figure 3, the Transformer achieved the highest F1-score of 0.586, with a precision of 0.592 and a recall of 0.580, indicating a comparatively balanced trade-off between precision and recall. The MLP achieved the highest accuracy of 0.699 and the highest precision of 0.655, but its recall was only 0.405, indicating a relatively conservative prediction tendency. Logistic Regression achieved the highest recall of 0.602 but exhibited lower AUC and precision. For service-interruption warning, false negatives may reduce the available lead time for proactive intervention. Therefore, the Transformer was selected as the risk encoder because it provides a favorable balance between ranking ability, F1 performance, and sequence-processing capability, rather than relying solely on the highest AUC value.



(a) Overall classification performance



(b) AUC-F1 performance space



(c) Precision-recall trade-off

Figure 3. Comparison of comprehensive performance of interruption prediction models

Figure 4 illustrates the stress test results across various lead times. The AUC values remained at 0.714, 0.724, and 0.714 for lead times of 1 to 3 periods, dropping to 0.693 at period 4 and 0.676 at period 6; the F1 score fluctuated between 0.564 and 0.581. The non-monotonic variation observed at short lead times stems from limited test samples and random label perturbations; thus, it should not be

interpreted as implying that period 2 is necessarily superior to period 1. While ranking capability generally declines as the lead time extends, threshold performance remains at a moderate level. Consequently, practical deployment should employ a multi-threshold strategy: utilizing long lead times for flexible preparation and short lead times for resource commitment.

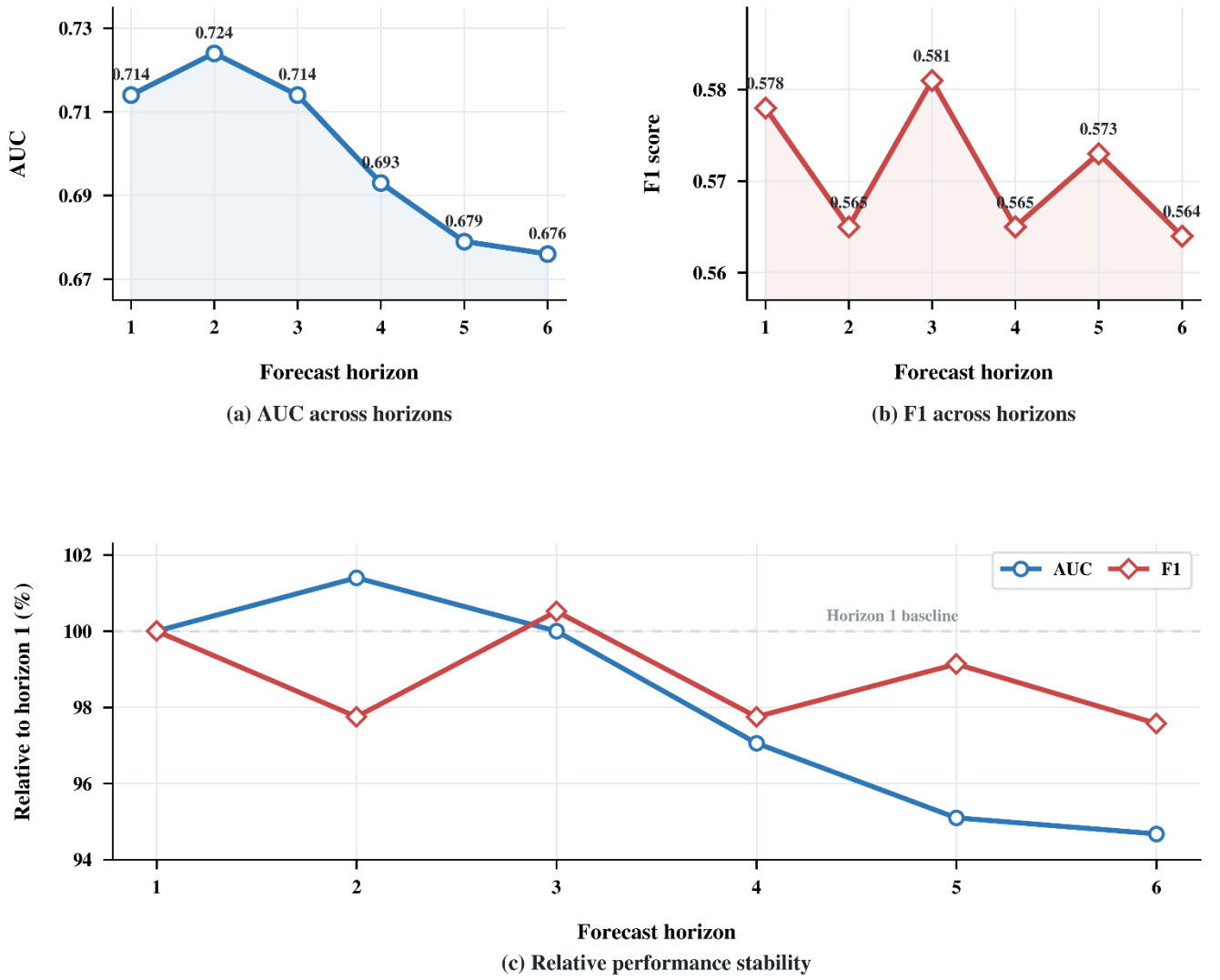


Figure 4. Stability of Transformer multi-lead-time forecasting

5.2. Training Convergence and Policy Stability

Figure 5 illustrates the 45-episode moving average of team returns. Each learning-based policy exhibits fluctuations due to exploration in the early stages before gradually entering a relatively stable regime; the difference between VDN-MARL and VDN without communication indicates that neighborhood information alters the cooperative

convergence trajectory. The T-MARL curve does not originate from a random policy but rather follows 750 episodes of prediction-augmented fine-tuning based on a pre-trained VDN coordination backbone; consequently, total training costs cannot be directly compared based on the episode count on the horizontal axis. Its purpose is to verify that the inclusion of risk gating does not lead to a sustained collapse in returns and to observe the stability of the fine-tuning process.

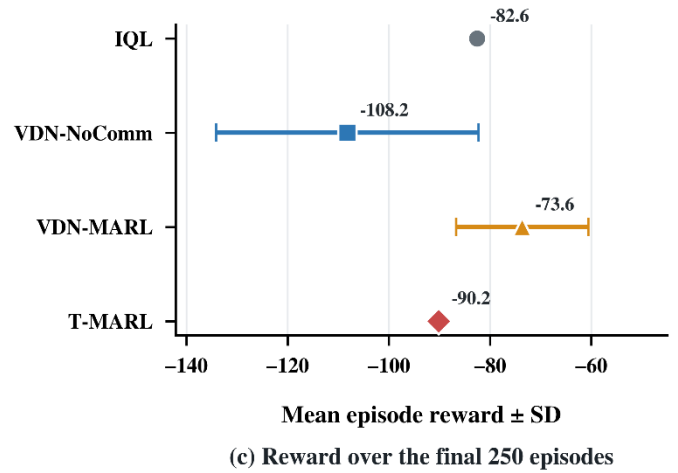
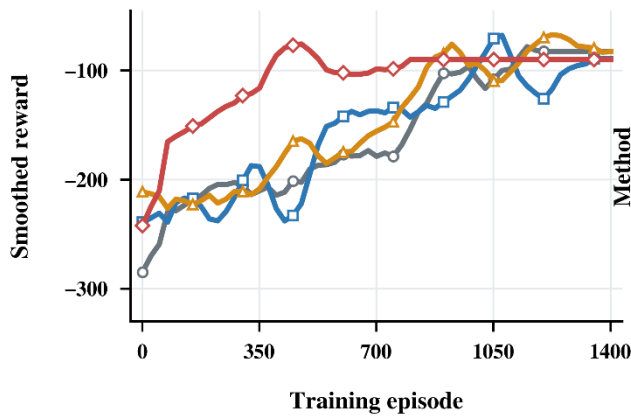
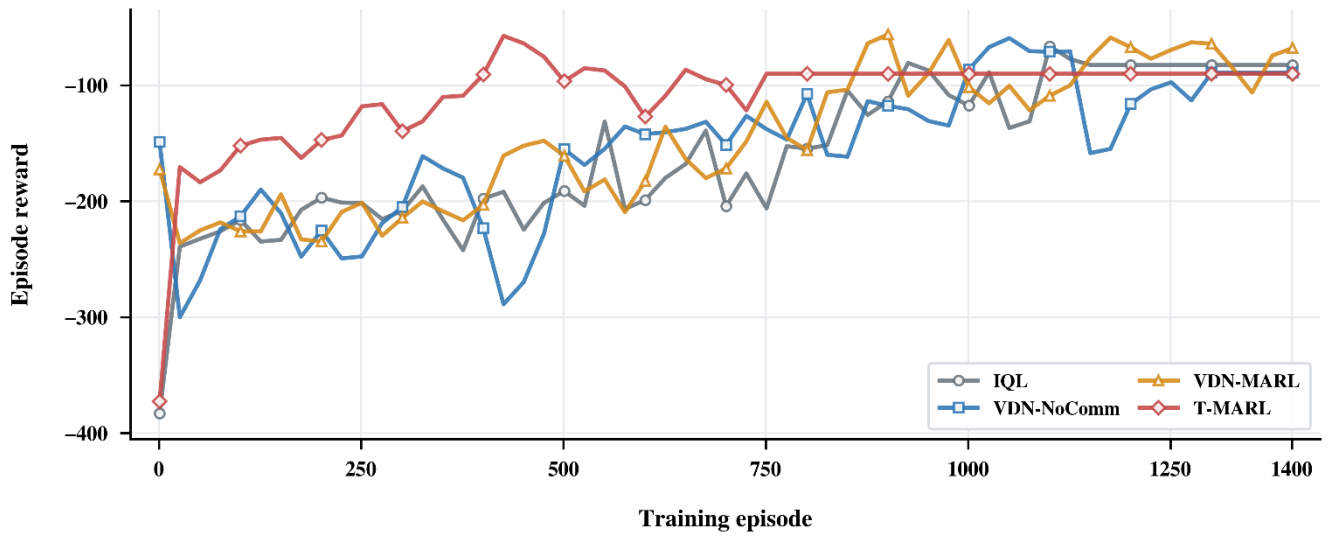


Figure 5. Convergence process of multi-agent strategy training

5.3. Dynamic Recovery Under Compound Disruptions

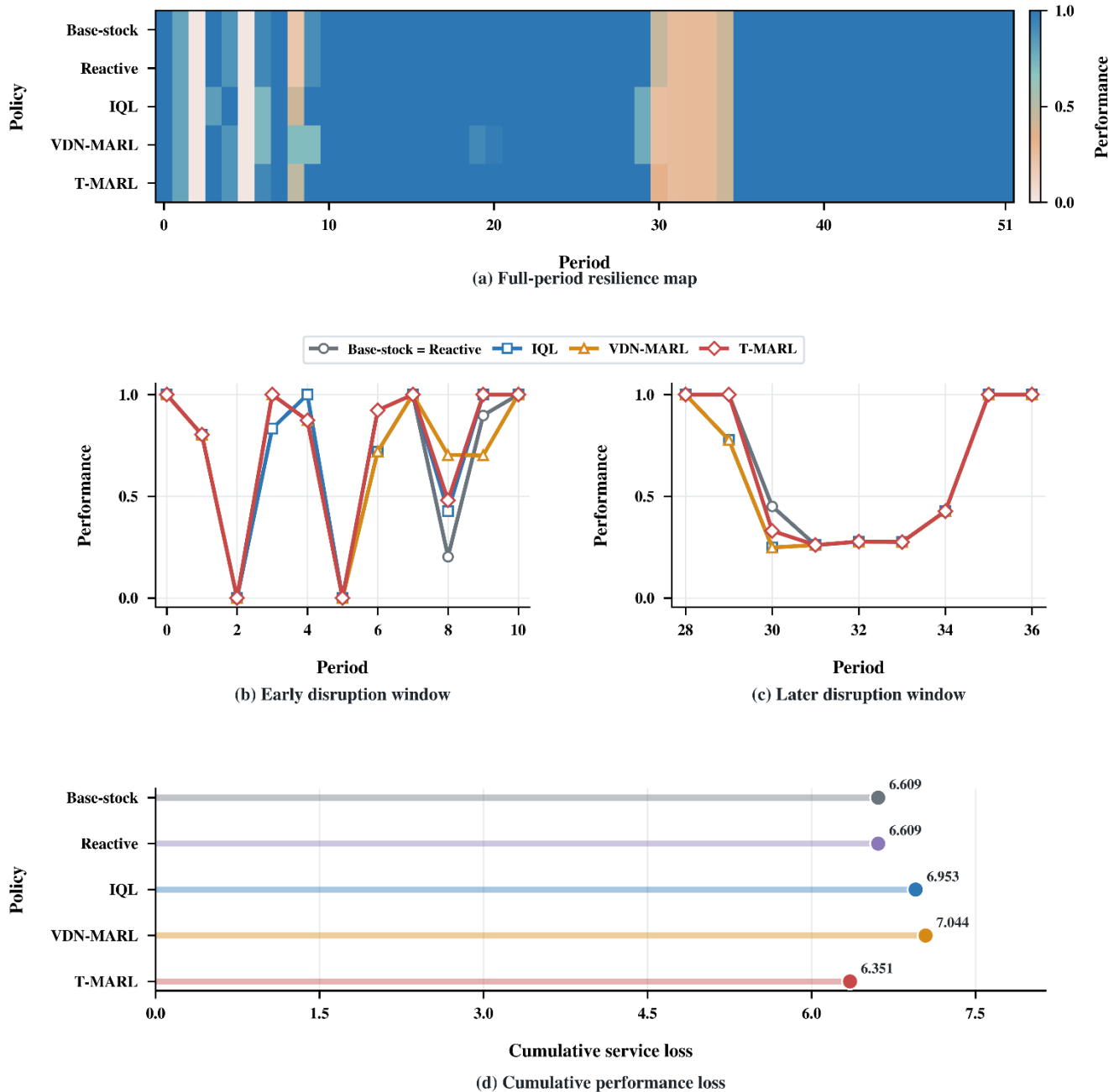


Figure 6. Service level recovery trajectory under compound disruptions

Figure 6 illustrates service levels under the same exogenous stochastic trajectory. Prior to the disruption, all policies operated near a steady state; however, the service curve dipped significantly following the combined impact of reduced supply capacity, rising demand, and transportation delays. T-MARL proactively increased upstream action

levels when it predicted that risk was entering a high range, thereby reducing the backlog tail following the disruption window. Rule-based policies relied on higher standing inventory to facilitate a rapid service recovery, albeit at the cost of higher inventory and expediting expenses; in the

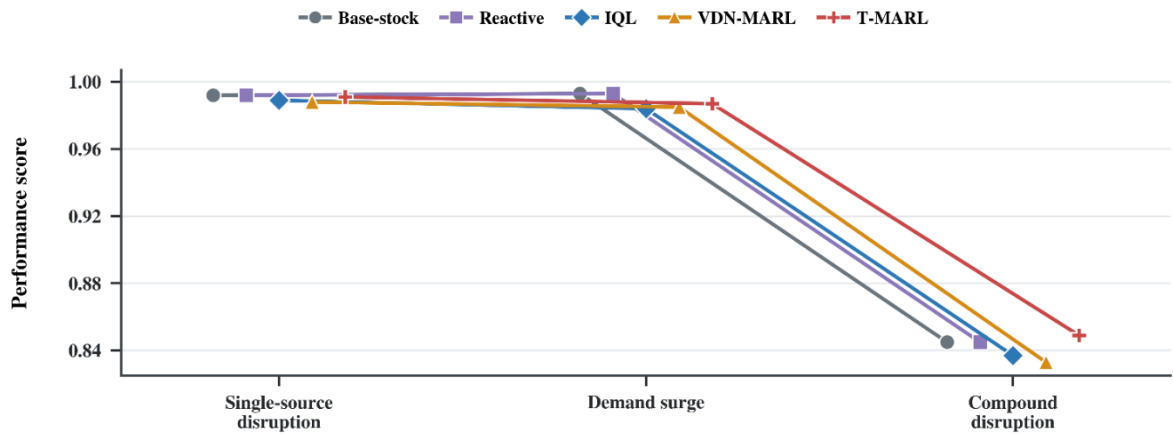
absence of probabilistic foresight, VDN-MARL and IQL relied more heavily on realized changes in inventory levels.

5.4. Key Results: Resilience, Cost, and Recovery Time

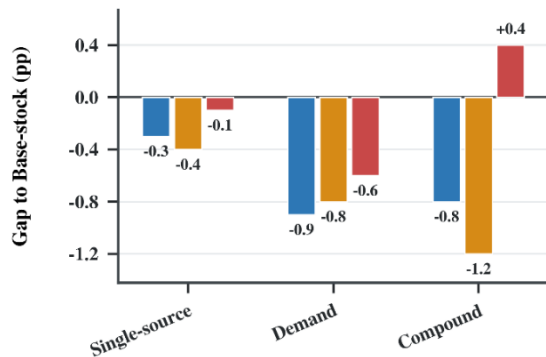
Table 5 and Figure 7 demonstrate that different disruption types correspond to different optimal frontiers. Under single-source supply disruptions, the resilience indices for the Base-stock and Reactive policies are both 0.992, while that for T-MARL is 0.991; under demand surges, the indices are 0.993 for Reactive and 0.987 for T-MARL. In other words, when facing simple shocks, high-inventory policies achieve a resilience advantage of approximately 0.06–0.61 percentage points at the cost of significantly higher expenses, meaning T-MARL does not yield the absolute highest resilience. Compared to the learning-based baselines, T-MARL improves resilience by 0.25 and 0.35 percentage points over IQL and VDN-MARL, respectively, under single-source disruptions, and by 0.29 and 0.16 percentage points under demand surges; one-sided paired tests confirm these positive differences ($p \leq 0.0026$).

Table 5. Main results of supply chain operations under three types of disruption scenarios

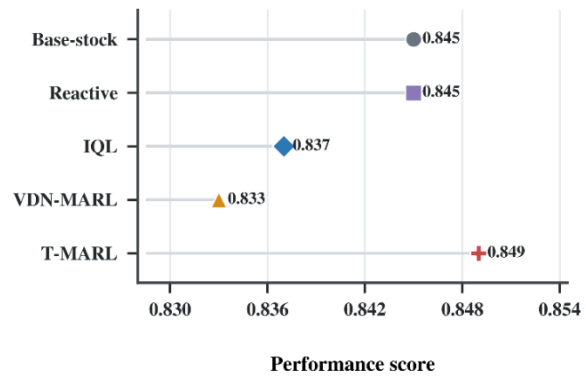
Scenario	Algorithm	Mean Resilience Index	Resilience Index Standard Deviation	Service Level	Total Cost	Recovery Time
Single-source supply disruption	Base-stock	0.992	0.005	0.993	9,983.5	0.00
	Reactive	0.992	0.005	0.993	10,303.0	0.00
	IQL	0.989	0.004	0.990	2,149.9	0.00
	VDN-MARL	0.988	0.005	0.989	2,212.2	0.00
	T-MARL	0.991	0.005	0.992	1,563.5	0.00
Surge in demand	Base-stock	0.993	0.005	0.993	8,240.7	0.00
	Reactive	0.993	0.005	0.993	8,969.1	0.00
	IQL	0.984	0.005	0.985	1,515.7	0.33
	VDN-MARL	0.985	0.006	0.987	1,340.4	1.03
	T-MARL	0.987	0.007	0.988	1,291.7	2.07
Compound disruption	Base-stock	0.845	0.030	0.877	34,501.2	0.80
	Reactive	0.845	0.030	0.877	34,595.5	0.80
	IQL	0.837	0.030	0.873	26,570.8	0.90
	VDN-MARL	0.833	0.033	0.872	31,413.3	1.67
	T-MARL	0.849	0.029	0.881	26,791.0	0.00



(a) Scenario-wise performance comparison



(b) Gap relative to Base-stock

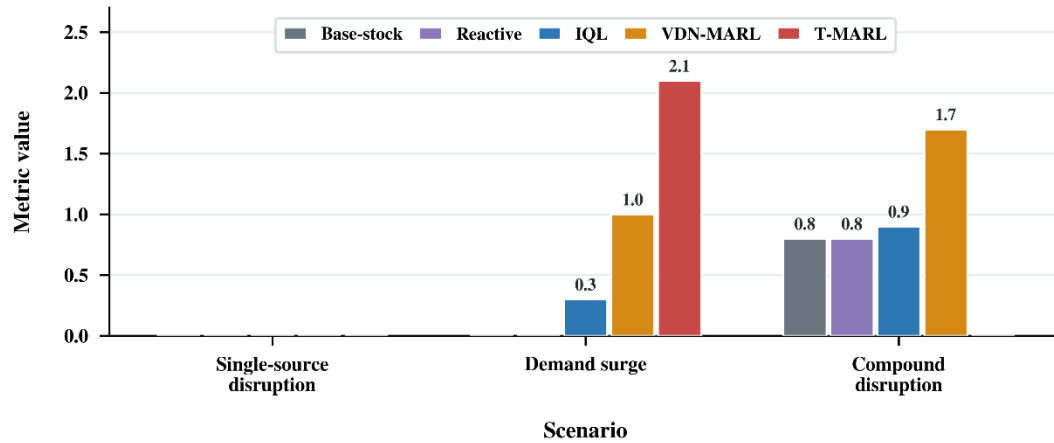


(c) Compound disruption comparison

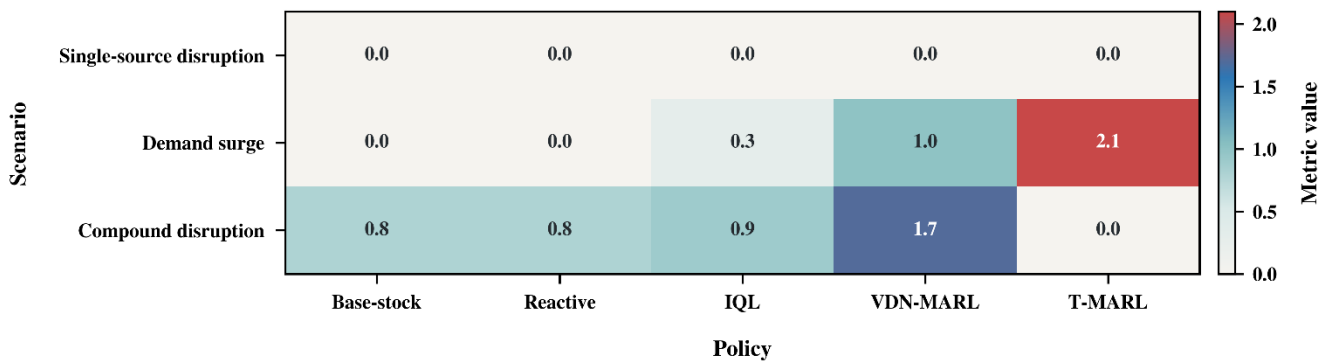
Figure 7. Supply chain resilience index under different disruption scenarios

Compound disruptions represent the scenario where the integrated framework demonstrates the greatest advantage. The T-MARL resilience index stands at 0.849, surpassing the 0.845 of the Base-stock and Reactive policies and the 0.837 of IQL, while representing a 1.63 percentage point improvement over VDN-MARL (0.833). One-sided pairwise comparisons across all compound scenarios yield p-values of less than 0.001. As shown in Figure 8, the mean post-

disruption recovery times are 0.00 periods for T-MARL, 0.80 periods for Base-stock and Reactive, 0.90 periods for IQL, and 1.67 periods for VDN-MARL. For T-MARL, a value of 0.00 indicates that the predefined service and backlog criteria had already been satisfied when the externally imposed disruption window ended. It should therefore be interpreted as no additional post-disruption recovery delay, rather than instantaneous physical recovery of all supply-chain states.



(a) Scenario-wise metric comparison



(b) Annotated response map

Figure 8. Post-interruption recovery time for different strategies

The Cost results change our view on the performance of rule-based strategies: The T-MARL cost (single-source disruption) is 1,563.5 which is less than that of Base-stock (9,983.5), and Reactive (10,303.0); For demand surges, the T-MARL cost is 1,291.7 which is again significantly less than both rule-based strategies. In case of compound disruption, the T-MARL cost is 26,791.0 – this is a decrease

of 14.7% when compared with VDN-MARL and 22.3% as compared to Base-stock but it increases by only 0.8%, when compared to IQL. So clearly T-MARL outperforms VDN-MARL; Compared to IQL we achieve an increase in resilience by 1.18 percentage points along with quicker recovery while there is a negligible increase in cost.

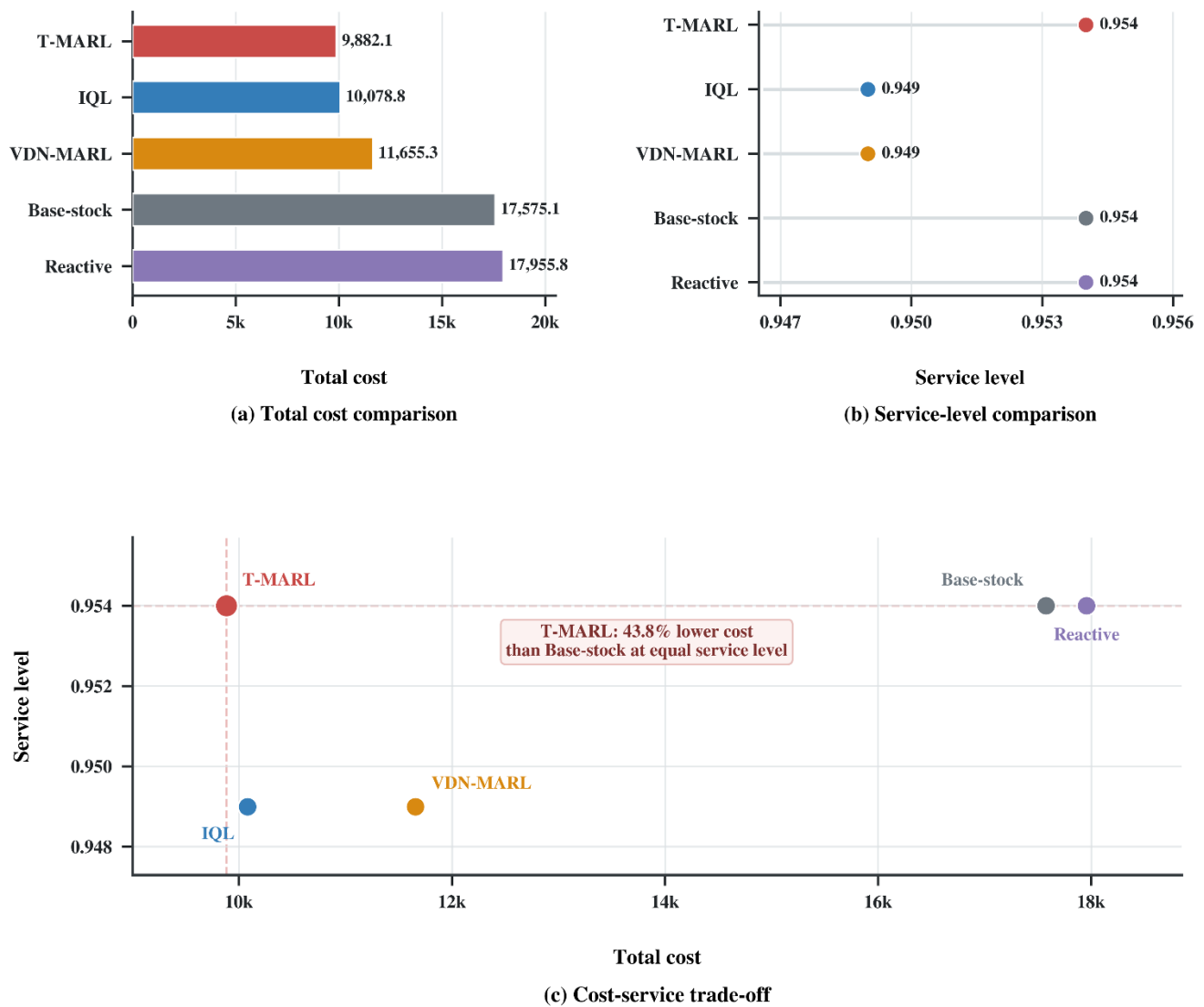


Figure 9. Cost-Service Level Pareto Relationship

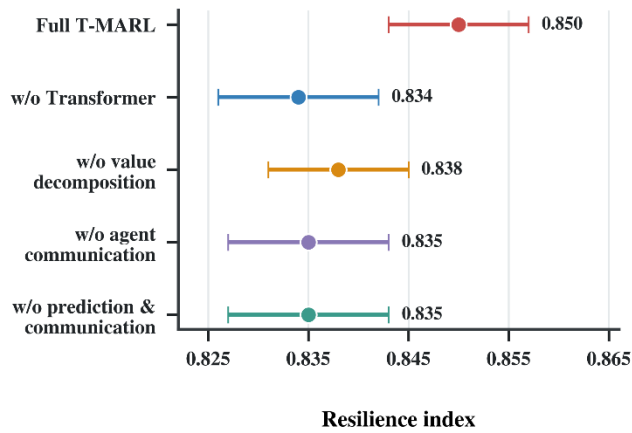
Figure 9 summarizes the three scenarios. T-MARL achieves an average service level of 0.954 and an average cost of 9,882.1, the lowest among the five strategies; IQL has a cost of 10,078.8 but a service level of only 0.949; VDN-MARL has a cost of 11,655.3 and a service level of 0.949; the two rule-based strategies achieve service levels of approximately 0.954 but incur costs of 17,575.1 and 17,955.8, respectively. Based on unrounded data, the service levels of Base-stock and Reactive are slightly higher than that of T-MARL—by approximately 0.04 to 0.05 percentage points—meaning T-MARL does not strictly dominate the rule-based strategies; more accurately, it saves approximately 43.8%–45.0% in costs while maintaining nearly identical average service levels, placing it in a more attractive trade-off zone.

5.5. Ablation, Robustness, and Hypothesis Testing

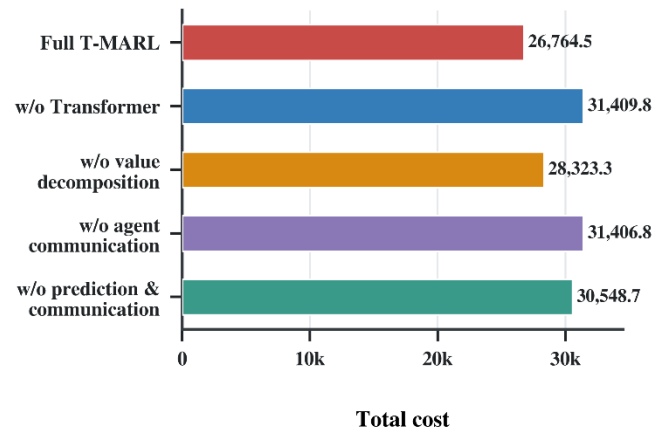
In Figure 10 this paper analyzes the effect on individual parts when subjected to compound disruptions. this paper finds that with the full T-MARL model, our resilience index is at 0.850 while if the Transformer is removed then it drops down to 0.834 (total cost rises from 26,764.5 to 31,409.8); if communication is removed, then our resilience becomes 0.835 (cost: 31,406.8) or if value decomposition is removed, then our resilience becomes 0.838 (cost: 28,323.3). Our full model performs better than all three ablations' models by about 1.56, 1.47 and 1.15 percentage points respectively showing how prediction, neighborhood information and team value attribution contribute independently. Interestingly, if this paper removes both prediction and communication there is no additional significant drop as

opposed to just removing communication suggesting they do not have purely additive effects and implying some form of

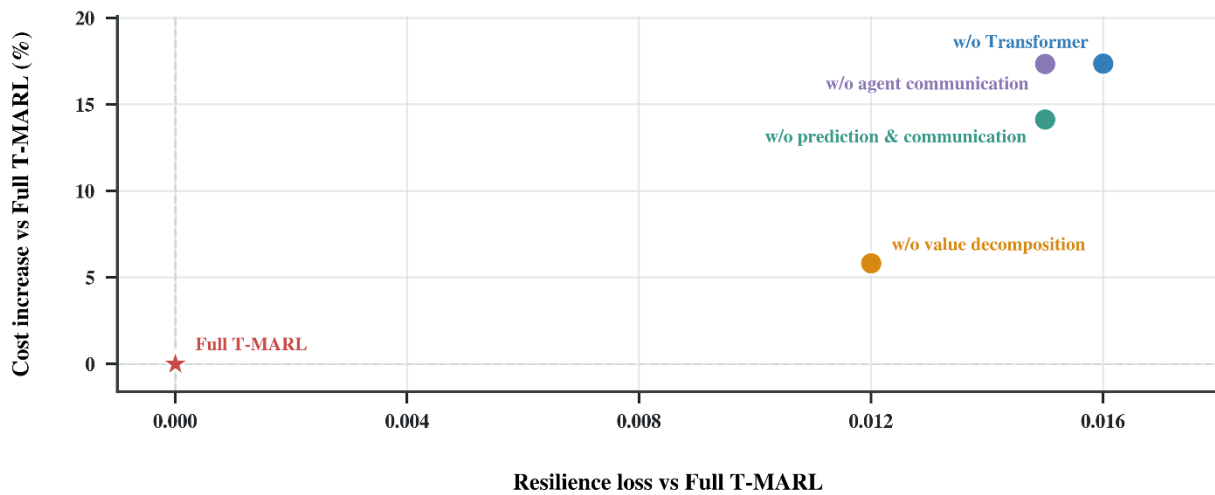
substitutability for the gating mechanism and the collaborative backbone.



(a) Resilience with 95% CI



(b) Total cost comparison



(c) Joint ablation penalty

Figure 10. T-MARL ablation study

Figure 11 supports H3. The resilience index is 0.847 in the absence of prediction noise and latency, but it drops to 0.786 when communication latency increases to three periods. When prediction noise increases up to 0.24 with real-time communication, this paper see resilience decrease down to 0.799; when combining noise of 0.24 and a three-period latency, it decreases further to only 0.726—down by 12.09 percentage points from ideal conditions. This shows how

improved prediction does not remove the necessity for data governance and robust communication architecture: erroneous signals cause unnecessary accumulation in inventory, and delays in information result in coordination using outdated states. Continuous monitoring is necessary in deployment strategies regarding input drift, message timestamps, missing values, and network isolation.

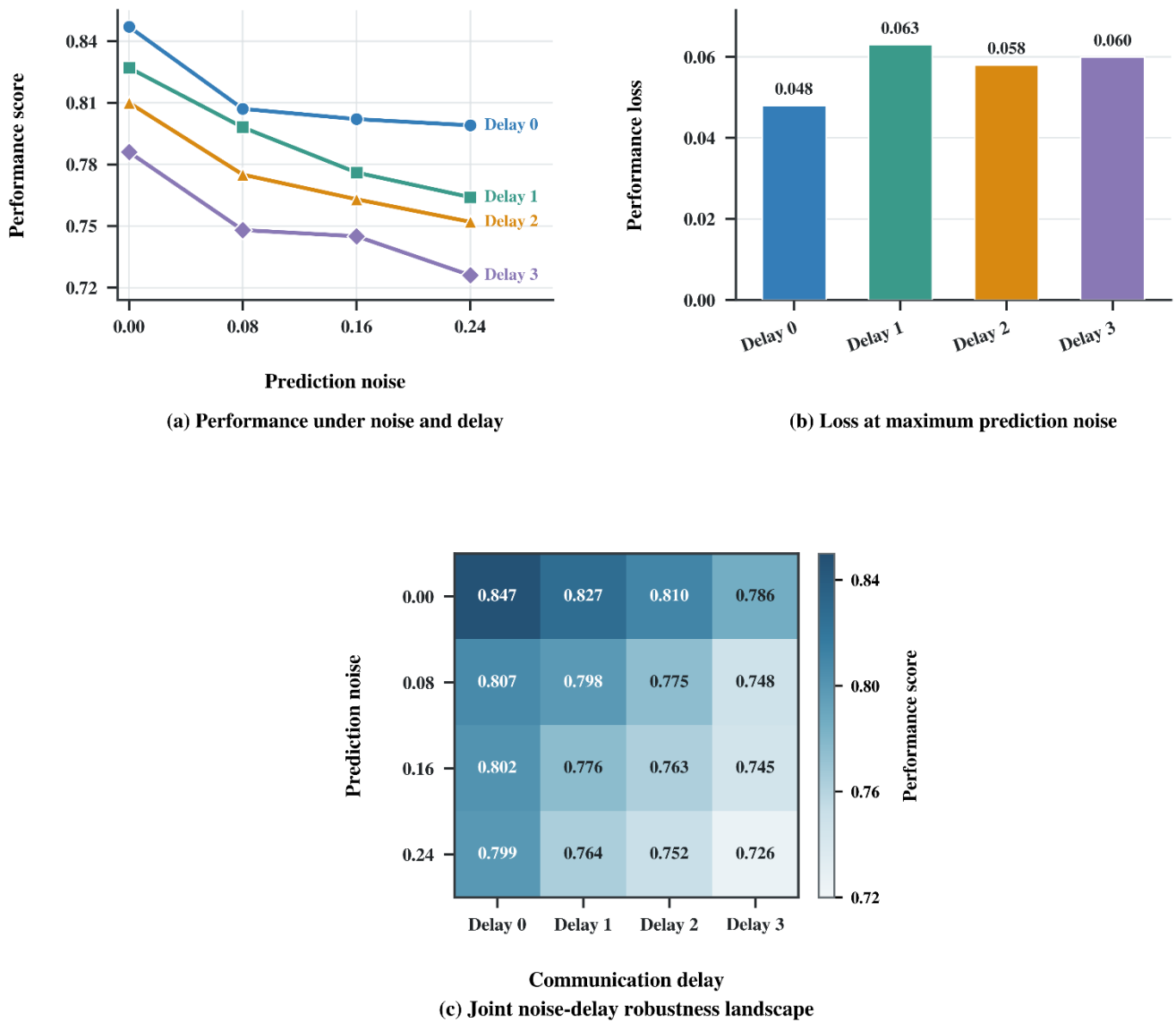


Figure 11. Robustness sensitivity to prediction error and communication delay

H1 is supported by comparison between our hybrid interruption mechanism versus a standard cooperative backbone: predictive gating improves resilience and reduces costs. H2 is partially supported; combining value decomposition with communication performs better than IQL or its ablations but the gain is less pronounced for simple demand shocks. H3 is supported: predictive noise and information latency lead to persistent performance degradation. High inventory rules show slightly greater resilience under simple scenarios indicating that the hybrid approach benefits most from shocks with identifiable precursors where multiple nodes need coordination and cost constraints play an important role.

6. Conclusions

In summary, the proposed T-MARL framework integrates disruption prediction, value-decomposition coordination, feasibility-constrained action gating, and recovery feedback into a closed operational loop. Its principal contribution is not the use of an individual prediction or reinforcement-learning algorithm. Rather, it provides an explicit interface through which a calibrated disruption probability can modify decentralized supply-chain actions while preserving physical feasibility.

a. In terms of disruption prediction, the Transformer achieved an AUC of 0.722, slightly lower than the Random Forest value of 0.724, while obtaining the highest F1-score of 0.586 among the four evaluated predictors. Its precision and recall were 0.592 and 0.580, respectively, demonstrating a comparatively balanced classification performance. These results indicate that the selection of a manufacturing

disruption predictor should not rely solely on a single ranking metric such as AUC; the cost of missed detections, risk lead time, precision–recall balance, and compatibility between predictive outputs and downstream scheduling actions should also be considered.

b. Second for resilience optimization: The best benefits were achieved under compound disruptions using T-MARL compared to VDN-MARL without a prediction module (increased resilience index by 1.63 percentage points), decreased total costs by 14.7%, and shortened the average recovery time from 1.67 periods down to zero. Compared to high-inventory rule-based strategies, though having slightly lower resilience against some simple shocks, T-MARL can significantly reduce inventory, expediting, and fluctuation costs nearly at an identical average service level, which is more rational on cost-service trade-off.

c. Ablation and sensitivity analysis presented here shows that each component (Transformer prediction, value decomposition and neighborhood communication) contributes independent towards system performance; prediction and coordination are not additive but rather work in concert with risk gating and team value attribution for enhancing the recovery process. Conversely, prediction noise and communication latency will always degrade resilience highlighting the need for high-quality data, timely messages, and reliable communication as a prerequisite to effectively deploy this framework.

d. The results of this paper are based upon parameterized simulations on a single product, four tier supply chain which is not yet applicable to actual corporate performance: we plan to use event logs from procurement, logistics, equipment, and suppliers for out-of-sample validation and extending to multi-product, multi-supplier, and shared-capacity networks with constrained reinforcement learning, graph neural networks, and distributionally robust optimization. For practical implementation, our paper must include mechanisms for human override, threshold calibration, and policy auditing. The T-MARL provides us with a methodology (interpretable, reproducible, and clear applicability boundaries), for AI-powered proactive early warning, collaborative mitigation, and rapid recovery in supply chains.

Acknowledgments

This study was funded by the Visiting and Research Program for Teachers of Colleges in Jiangsu Province (grant number 2025GRFX058).

References

- [1] Tukamuhabwa B R, Stevenson M, Busby J, Zorzini M. Supply chain resilience: Definition, review and theoretical foundations for further study. *International Journal of Production Research*, 2015, 53(18): 5592-5623.
- [2] Hosseini S, Ivanov D, Dolgui A. Review of quantitative methods for supply chain resilience analysis. *Transportation Research Part E: Logistics and Transportation Review*, 2019, 125: 285-307.
- [3] Dolgui A, Ivanov D, Sokolov B. Ripple effect in the supply chain: An analysis and recent literature. *International Journal of Production Research*, 2018, 56(1-2): 414-430.
- [4] Ivanov D, Dolgui A, Blackhurst J V, Choi T M. Toward supply chain viability theory: From lessons learned through COVID-19 pandemic to viable ecosystems. *International Journal of Production Research*, 2023, 61(8): 2402-2415.
- [5] Ivanov D, Dolgui A. A digital supply chain twin for managing the disruption risks and resilience in the era of Industry 4.0. *Production Planning & Control*, 2021, 32(9): 775-788.
- [6] Guo D, Mantravadi S. The role of digital twins in lean supply chain management: Review and research directions. *International Journal of Production Research*, 2025, 63(5): 1851-1872.
- [7] Kassa A, Kitaw D, Stache U, Beshah B, Degefu G. Artificial intelligence techniques for enhancing supply chain resilience: A systematic literature review, holistic framework, and future research. *Computers & Industrial Engineering*, 2023, 186: 109714.
- [8] Culot G, Podrecca M, Nassimbeni G. Artificial intelligence in supply chain management: A systematic literature review of empirical studies and research directions. *Computers in Industry*, 2024, 162: 104132.
- [9] Jauhar S K, Jani S M, Kamble S S, et al. How to use no-code artificial intelligence to predict and minimize the inventory distortions for resilient supply chains. *International Journal of Production Research*, 2024, 62(15): 5510-5534.
- [10] Chen Z, Ma M, Li T, Wang H, Li C. Long sequence time-series forecasting with deep learning: A survey. *Information Fusion*, 2023, 97: 101819.
- [11] Ahmed S, Nielsen I E, Tripathi A, et al. Transformers in time-series analysis: A tutorial. *Circuits, Systems, and Signal Processing*, 2023, 42(12): 7433-7466.
- [12] Mousa M, van de Berg D, Kotecha N, del Rio-Chanona E A, Mowbray M. An analysis of multi-agent reinforcement learning for decentralized inventory control systems. *Computers & Chemical Engineering*, 2024, 188: 108783.
- [13] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. *Advances in neural information processing systems*, 2017, 30.
- [14] Zerveas G, Jayaraman S, Patel D, et al. A transformer-based framework for multivariate time series representation learning[C]//*Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*. 2021: 2114-2124.
- [15] Lim B, Arık S Ö, Loeff N, et al. Temporal fusion transformers for interpretable multi-horizon time series forecasting[J]. *International journal of forecasting*, 2021, 37(4): 1748-1764.
- [16] Sunehag P, Lever G, Gruslys A, et al. Value-decomposition networks for cooperative multi-agent learning[J]. *arXiv preprint arXiv:1706.05296*, 2017.
- [17] Stranieri F, Fadda E, Stella F. Combining deep reinforcement learning and multi-stage stochastic programming to address the supply chain inventory management problem. *International Journal of Production Economics*, 2024, 268: 109099.
- [18] Kotecha N, del Rio Chanona A. Leveraging graph neural networks and multi-agent reinforcement learning for inventory control in supply chains. *Computers & Chemical Engineering*, 2025, 199: 109111.
- [19] Hu J, Xia L, Huang T, Wu H. A multi-agent deep reinforcement learning approach for multi-echelon inventory optimization and its application to the beer game. *Transportation Research Part E: Logistics and Transportation Review*, 2025, 203: 104367.

- [20] Smyth C, Dennehy D, Fosso Wamba S, Scott M, Harfouche A. Artificial intelligence and prescriptive analytics for supply chain resilience: A systematic literature review and research agenda. *International Journal of Production Research*, 2024, 62(23): 8537-8561.
- [21] Lowe R, Wu Y I, Tamar A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[J]. *Advances in neural information processing systems*, 2017, 30.
- [22] Oroojlooyjadid A, Nazari M R, Snyder L V, et al. A deep q-network for the beer game: Deep reinforcement learning for inventory optimization[J]. *Manufacturing & Service Operations Management*, 2022, 24(1): 285-304.
- [23] Boute R N, Gijsbrechts J, Van Jaarsveld W, et al. Deep reinforcement learning for inventory control: A roadmap[J]. *European journal of operational research*, 2022, 298(2): 401-412.
- [24] Bengio Y, Louradour J, Collobert R, et al. Curriculum learning[C]//*Proceedings of the 26th annual international conference on machine learning*. 2009: 41-48.
- [25] Gijsbrechts J, Boute R N, Van Mieghem J A, et al. Can deep reinforcement learning improve inventory management[J]. *Performance on dual sourcing, lost sales and multi-echelon problems*, 2019.
- [26] Achiam J, Held D, Tamar A, et al. Constrained policy optimization[C]//*International conference on machine learning*. Pmlr, 2017: 22-31.
- [27] Tobin J, Fong R, Ray A, et al. Domain randomization for transferring deep neural networks from simulation to the real world[C]//*2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017: 23-30.
- [28] Yang Y, Wang M, Wang J, et al. Multi-agent deep reinforcement learning for integrated demand forecasting and inventory optimization in sensor-enabled retail supply chains[J]. *Sensors*, 2025, 25(8): 2428.
- [29] Dehaybe H, Catanzaro D, Chevalier P. Deep reinforcement learning for inventory optimization with non-stationary uncertain demand. *European Journal of Operational Research*, 2024, 314(2): 433-445.
- [30] Wang J, Swartz C L E, Huang K. Risk-averse supply chain management via robust reinforcement learning. *Computers & Chemical Engineering*, 2025, 192: 108912.
- [31] Dolgui A, Ivanov D, Simchi-Levi D. Stress tests for supply chains: Towards resilience and viability. *International Journal of Production Research*, 2025, 63(9): 3254-3258.