Proceedings of the 2025 Winter Simulation Conference
E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.

SUPPLY CHAIN OPTIMIZATION VIA GENERATIVE SIMULATION AND ITERATIVE DECISION POLICIES

Haoyue Bai[1], Haoyu Wang[2], Nanxu Gong[1], Xinyuan Wang[1], Wangyang Ying[1], Haifeng Chen[2], and Yanjie Fu[1]

[1]School of Computing and Augmented Intelligence, Arizona State University, AZ, USA
[2]Data Science and System Security, NEC Labs America, NJ, USA

ABSTRACT

High responsiveness and economic efficiency are critical objectives in supply chain transportation, both of which are influenced by strategic decisions on shipping mode. An integrated framework combining an efficient simulator with an intelligent decision-making algorithm can provide an observable, low-risk environment for transportation strategy design. An ideal simulation-decision framework must (1) generalize effectively across various settings, (2) reflect fine-grained transportation dynamics, (3) integrate historical experience with predictive insights, and (4) maintain tight integration between simulation feedback and policy refinement. We propose Sim-to-Dec framework to satisfy these requirements. Specifically, Sim-to-Dec consists of a generative simulation module, which leverages autoregressive modeling to simulate continuous state changes, reducing dependence on handcrafted domain-specific rules and enhancing robustness against data fluctuations; and a history–future dual-aware decision model, refined iteratively through end-to-end optimization with simulator interactions. Extensive experiments conducted on three real-world datasets demonstrate that Sim-to-Dec significantly improves timely delivery rates and profit.

1 INTRODUCTION

Efficient transportation plays a central role in supply chains across commerce, finance, and agriculture industries, directly supporting logistics execution, demand fulfillment, and resource coordination (Yu et al. 2017; Kristofik et al. 2012; Routroy and Behera 2017). A well-functioning supply chain transportation system must strike a balance between high responsiveness and economic efficiency, where achieving this balance largely depends on strategic transportation decisions (Saisridhar et al. 2024; Bi et al. 2022). Transportation decisions, such as selecting for each order between air, rail, or maritime shipping, involve inherent trade-offs between delivery speed and cost. Faster modes improve responsiveness but incur higher expenses, while slower options reduce costs at the risk of delay. Optimizing such decisions is essential for improving responsiveness and efficiency under dynamic supply chain conditions.

Simulation has long served as a powerful tool for evaluating supply chain transportation strategies before real-world deployment (Daroń 2022; Chen et al. 2024). A promising direction is to couple simulation with intelligent decision-making, enabling continuous optimization in a low-risk and adaptive environment. Several research directions are intrinsically relevant to the development of integrated simulation–decision frameworks. Traditional simulation techniques, such as discrete event simulation (Law et al. 2007) and Monte Carlo methods (Dimov 2008), provide detailed modeling of logistics processes but often require extensive expert knowledge and manual parameter tuning, which can limit their scalability across diverse operational settings. Reinforcement learning (RL) approaches (Sutton and Barto 1998) enable adaptive policy optimization through interaction with the environment. However, when applied to complex supply chain systems, RL methods may face practical challenges such as high computational costs and sensitivity to objective changes (Huang et al. 2022). They often treat the environment as a black box or assume fixed, coarse-grained dynamics, which limits their ability to model fine-grained state transitions or adapt to real-world disruptions such as demand fluctuations, policy shifts, and capacity constraints (Du et al. 2020). In addition, many existing methods loosely couple simulation and decision-making: traditional simulators are typically used as post-hoc evaluators (Zhou et al. 2018; Daroń 2022) without actively guiding the decision process.

To effectively support transportation strategy design, we argue that an ideal simulation–decision framework should satisfy four key criteria. (1) *Generalizability* — the framework should be data-driven and easily adaptable to various supply chain settings without relying on handcrafted, domain-specific modeling. (2) *Dynamic fidelity* — it should capture fine-grained transportation dynamics, including continuous and interdependent state transitions that evolve over time. (3) *Experience-forecast integration* — the decision model should leverage both historical operational data and future-oriented predictions to formulate informed strategies. (4) *Tight simulation–decision coupling* — the simulator and the decision model should interact iteratively, enabling the simulation to guide the decision-making rather than operating in isolation.

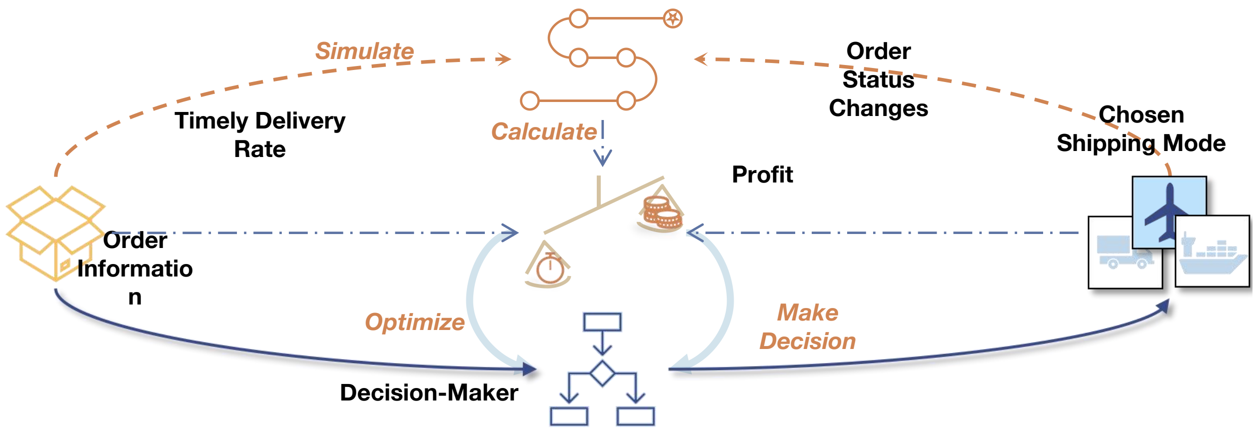


Figure 1: Proposed Framework

To meet these criteria and support adaptive, efficient transportation strategy design, we propose ***Sim***ulation-to-***Dec***ision (Sim-to-Dec), a novel framework that unifies generative simulation with end-to-end decision optimization. As illustrated in Figure 1, the framework operates in three stages. First, the decision model selects an appropriate shipping mode for each order based on its attributes. Then, the simulator predicts the order's status evolution conditioned on the order details and the chosen mode. Finally, the system evaluates the strategy using key metrics such as profit and on-time delivery rate based on the simulated states. Sim-to-Dec consists of two tightly integrated components: (1) *Generative simulation for scalable and data-driven modeling.* The simulator learns transportation dynamics directly from historical data, eliminating the need for manual rule engineering. It is implemented as an autoregressive model that sequentially predicts order states across transportation stages, capturing both short-term fluctuations and long-term trends. This enables fine-grained simulation of continuous and interdependent logistics behaviors, supporting generalization across diverse supply chain settings. (2) *End-to-end decision-making with independent yet coupled learning.* The decision model is parameterized independently from the simulator, allowing it to flexibly adapt to evolving objectives and constraints. It interacts iteratively with the simulator in a low-risk virtual environment, continuously refining transportation strategies through simulated feedback. To support experience-forecast integration, the model combines two complementary perspectives: historical execution data is used to estimate expected outcomes of shipping modes, while a value network predicts future rewards. These sources jointly guide policy optimization, enabling decisions that are both historically grounded and forward-looking. By integrating these components, Sim-to-Dec forms a closed feedback loop: the generative simulator infers dynamic logistics conditions, while the decision model actively learns and adapts strategies. This architecture directly addresses the four key design goals outlined earlier—generalizability, dynamic fidelity, experience-forecast integration, and tight simulation–decision coupling—making Sim-to-Dec well-suited for the complexity and uncertainty of real-world supply chain transportation.

Key Contributions: This work makes the following contributions to supply chain transportation optimization: (1) **Framework:** We propose a novel simulation-decision framework that integrates generative simulation with iterative decision-making, providing a flexible and adaptive approach for optimizing transportation strategies in the supply chain. (2) **Decision Optimization:** Our method combines insights from historical data with future reward estimation to guide decision-making, achieving a balance between leveraging experience and adapting to evolving conditions. (3) **Empirical Validation:** We validate the proposed framework through extensive experiments on three real-world supply chain datasets and a live transportation system, demonstrating significant improvements in transportation performance, responsiveness, and decision robustness. The code is available at https://github.com/HaoyueBai98/Sim-to-Decision.

This paper is organized as follows. Section 2 reviews related work on simulation and decision-making in supply chain optimization. Section 3 provides background on shipping mode management in supply chains and formally defines the problem. Section 4 introduces our Sim-to-Dec framework, detailing its two core components: a generative simulator and a decision-making model. Section 5 presents our experimental setup and results, including comparisons with baselines and ablation studies. We conclude with a discussion of future directions.

2 RELATED WORK

Simulation-based methods are widely used to evaluate supply chain transportation strategies under uncertainty. Traditional techniques include discrete event simulation (DES) (Law et al. 2007), system dynamics (SD) (Forrester 1997), and agent-based modeling (ABM), each capturing different aspects of system characteristics and behaviors. DES is particularly suitable for modeling operational-level disruptions, SD provides insights into system-wide feedback-driven behaviors, and ABM effectively models decentralized agent interactions. Hybrid approaches that combine these paradigms have also been explored to leverage their complementary modeling strengths (Brailsford et al. 2019). Probabilistic methods such as Monte Carlo simulation (Dimov 2008) and Markov-based modeling (Hosseini et al. 2019) are also commonly employed, particularly for quantifying variability and modeling stochastic transitions or cascading failures (Dixit et al. 2020). These traditional simulations offer strong interpretability and transparency, but typically require extensive expert knowledge and rule-based parameterization, which significantly limits their adaptability to dynamic and evolving operational conditions.

On the decision-making side, classical optimization methods (e.g., linear programming (Dantzig 2002; Churchman et al. 1957)) have long been used to derive cost-effective policies. More recently, reinforcement learning (RL) has emerged as a promising alternative (Rolf et al. 2023), allowing agents to learn adaptive strategies through environmental interaction. However, RL approaches are often sample inefficient, computationally expensive, and assume black-box environments with coarse dynamics (Adobor 2020; Huang et al. 2022; Du et al. 2020), which significantly undermines their applicability in complex, fine-grained supply chains. To address these limitations, recent research has explored integrating simulation and decision-making into unified frameworks. For example, Correa-Martinez and Seck (Correa-Martinez and Seck 2023; Bai et al. 2025) and others have proposed simulation-driven policy optimization or digital twin architectures (Barykin et al. 2020), enabling continuous improvement of decision strategies under uncertainty. However, many of these frameworks still treat simulation as a passive evaluator rather than an active participant in iterative decision refinement. The coupling between simulation and policy learning remains loose (Zhou et al. 2018; Daroń 2022), and simulation fidelity is often limited to simplified dynamics or offline scenario replay, which reduces generalizability. Our work contributes to this growing literature by proposing a tightly integrated simulation–decision framework that directly overcomes these limitations.

3 BACKGROUND AND PROBLEM STATEMENT

Challenges of Shipping Mode Management & Order Representation. In supply chain transportation, selecting an appropriate shipping mode is a fundamental decision that directly affects both responsiveness and cost efficiency. Expedited modes such as air freight improve delivery speed but incur higher costs, while slower options like maritime or rail transport are more economical but risk significant delays. Thus, one of the key challenges in shipping mode management is to strategically balance timely delivery rates and profits through mode selection. To support this decision process, each order in the supply chain is represented by three distinct groups of attributes: (1) Order information related attributes, denoted by F^I, are a set of attributes observed and collected at the time of order placement. The order information is inherent attributes that are not affected by decisions and will not change over time, denoted by $F^I = \{f_1^I, f_2^I, \ldots, f_{|F^I|}^I\}$, where $|F^I|$ is the number of order information related attributes, for instance, the origin, destination, product type, quantity, and required delivery time of an order. (2) Shipping modes, denoted by (F^D), are a set of candidate shipping modes (in our experiments, we has four shipping modes) for an order, denoted by $F^D = \{f_1^D, f_2^D, \ldots, f_{|F^D|}^D\}$. Each $f_d^D \in F^D$ is a specific shipping mode chosen from a predefined set of options, such as air, maritime, or ground transport. (3) Order states related attributes, denoted by F^E, are a set of attributes describing the changes or evolutions of an order after a shipping mode is selected since an order has dynamics and its state changes. We define three order state attributes: i) a binary variable indicating whether the order has a risk of delay, denoted by f_{risk}^E, for instance, high risk or low risk. ii) a categorical variable indicating the number of days required for delivery, f_{time}^E, for instance, 1, 2, 3, or 4 days. iii) a binary variable indicating whether the order is ultimately delivered on time, denoted by f_{status}^E.

Simulation–Decision Integration for Shipping Strategy Optimization. The goal of shipping mode management is to jointly optimize two key metrics: the timely delivery rate (T^{Timely}) and profit (T^{Profit}), which are often in conflict. Direct deployment of optimization strategies in real-world systems is infeasible due to operational risks and constraints. To address this, we propose Sim-to-Dec, a closed-loop simulation–decision framework that enables risk-free evaluation and refinement of transportation strategies using historical data.

Sim-to-Dec consists of two tightly coupled components: a generative simulator and a policy-based decision maker. The simulator, denoted as $\mathscr{S}$, learns fine-grained transportation dynamics from historical data and simulates the impact of shipping decisions. The decision maker, $\mathscr{M}$, learns a policy to select optimal shipping modes for individual orders, based on their information features F_n^I. Specifically, for the n-th order:

$$\hat{f}_{d,n}^D = \mathscr{M}(F_n^I; \Phi), \tag{1}$$

where Φ is the parameter set of the decision model. The selected shipping mode $\hat{f}_{d,n}^D$ is then passed into the simulator to generate the predicted order state $\hat{F}_n^E$:

$$\hat{F}_n^E = \mathscr{S}(F_n^I, \hat{f}_{d,n}^D; \Theta), \tag{2}$$

where Θ denotes the simulator's parameters. The simulation provides feedback on expected outcomes, which is used to iteratively refine the decision policy. The objective is to maximize both T^{Timely} and T^{Profit} over a batch of N orders. After preprocessing, all necessary attributes—including order details, candidate modes, and outcomes—are available from real datasets to support both training and evaluation.

4 THE SIM-TO-DEC APPROACH

Sim-to-Dec consists of two integrated components: a *generative simulator* and a *decision-maker*. The generative simulator models the dynamics of a supply chain system by learning from historical order data and generating how the supply chain system changes when the shipping modes of orders are chosen. The decision-maker interacts with the generative simulator to iteratively optimize shipping mode selection, ensuring a balance between timely delivery and cost efficiency. Through continuous simulation feedback, the decision-maker dynamically refines decision policies to adapt to changing supply chain conditions.

4.1 Generative Simulator via Deep Generative AI

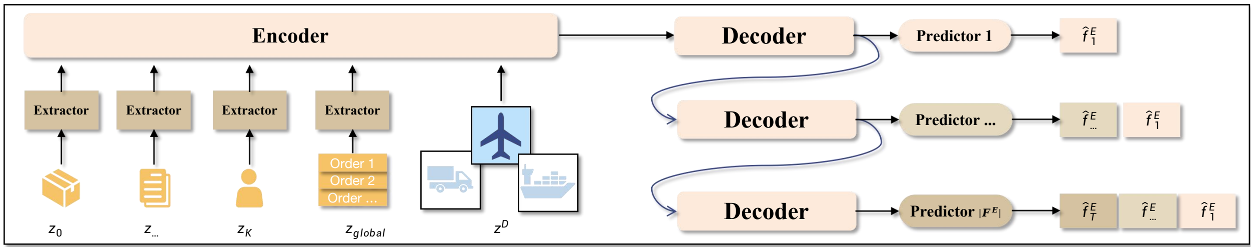


Figure 2: Simulator

Traditional simulation approaches in supply chain shipping mode management heavily rely on expert-driven models with predefined heuristics, making them rigid and difficult to adapt to evolving market conditions. To address these limitations, we propose a *generative simulator* that models simulation as a generative AI task by learning directly from offline historical data. This approach eliminates the need for extensive expert knowledge while allowing for adaptive, high-fidelity simulations.

Processing Order Information. To simulate order dynamics, we categorize *order information* (F^I) into multiple groups, each representing distinct aspects of an order's attributes: (1) Product-Related Attributes $F^{I,p}$: characteristics of a shipped item, such as weight and product category. (2) Customer-Related Attributes $F^{I,c}$: characteristics of an order recipient, including location, priority status, and historical purchase behavior. (3) Shipping-Related Attributes $F^{I,s}$: order-specific constraints, such as scheduled delivery time, shipping restrictions, and previous shipping history. (4) Order-Related Attributes $F^{I,o}$: statistical order attributes, including order type, order placement time, and payment method.

Since individual shipping events are interconnected, we introduce a global order group $F^{I,\text{global}}$ as an additional attribute group that captures dependencies among multiple orders processed within the same time frame. This global representation is constructed by *pooling* information from all orders in a given batch, ensuring that system-wide effects:

$$F^{I,\text{global}} = \text{Pool}(\{F^I_n\}_{n=1}^N), \tag{3}$$

where $\text{Pool}(\cdot)$ aggregates the attribute representations of all N orders in a batch, capturing an overall system-wide summary. This provides a holistic context that informs the decision-making process. To ensure a unified attribute processing pipeline, we treat the global representation as an additional attribute group, alongside other groups. Each attribute group, including the global representation, is processed using a separate linear transformation layer:

$$\mathbf{z}_n^k = \mathbf{W}^k F_n^{I,k} + \mathbf{b}^k, \quad \forall k \in \{p, c, s, o, \text{global}\}, \tag{4}$$

where k denotes different attribute groups, $\mathbf{W}^k$ and $\mathbf{b}^k$ the learnable vector and $\mathbf{z}_n^k$ is the representation of group k. This transformation ensures that all attribute groups, including the global representation, are mapped into a consistent latent space before being fed into the downstream model. The processed attribute embeddings are then treated as input tokens for the encoder, allowing it to simultaneously capture both individual shipment details and system-wide patterns.

Processing Shipping Mode. To simulate system dynamics conditioned on shipping decisions, we embed the selected shipping mode f_n^D into a latent representation $\mathbf{z}_n^D$. We implement this via an embedding lookup:

$$\mathbf{z}_n^D = \text{Embedding}(f_n^D), \quad f_n^D \in F^D, \tag{5}$$

where the embedding table is initialized using samples from a normal distribution.
Fusing and Encoding. We utilize the Long Short-Term Memory (LSTM) model as an encoder to encode the transformed representations of both order information and shipping mode into a fused embedding. Formally, the encoding process is given by:

$$\mathbf{z}_n = \mathrm{LSTM}(\mathbf{z}_n^p, \mathbf{z}_n^c, \mathbf{z}_n^s, \mathbf{z}_n^o, \mathbf{z}_n^{global}, \mathbf{z}_n^D). \tag{6}$$

Generating Order Status Changes. Order status changes describe how orders in the system change when the decision maker chooses different shipping modes. Understanding what will happen after taking a decision can enable us to measure the utility of a decision and adjust the policies of the decision maker. We see the simulation as a task of generating evolutionary attributes given the necessary information, i.e., order information and shipping mode. In particular, we utilize the LSTM decoder to autoregressively generate evolutionary attributes. Using the fused embedding $\mathbf{z}_n$, the decoder sequentially generates the latent representations of changes in order status. Formally, we generate the embedding $\mathbf{z}_{e,n}^E$ of the e^{th} order status attribute of the order n by:

$$\mathbf{z}_{e,n}^E = \mathrm{LSTM}(\mathbf{z}_n, \mathbf{z}_{<e,n}^E), \tag{7}$$

where $\mathbf{z}_{<e,n}^E$ denotes the embedding of the order status attribute from 1^{st} to $(e-1)^{th}$. Finally, a dedicated predictor is used to predict the value of a specific order status attribute. Formally, given the embedding $\mathbf{z}_{e,n}^E$ of the e^{th} evolutionary attribute, let $\mathscr{P}_e$ be the dedicated predictor of e^{th} order status attribute, the generated/simulated value $\hat{f}_{e,n}^E$ of the e^{th} order status attribute is given by:

$$\hat{f}_{e,n}^E = \mathscr{P}_e(\mathbf{z}_{e,n}^E). \tag{8}$$

Final Optimization Objective of Simulator. The simulator's ability to accurately reflect real-world dynamics is essential for reliable analysis and informed decision-making. To ensure fidelity, we leverage real-world historical data as ground truth to train the model. Each order status is optimized by minimizing the following objective function:

$$\mathscr{L}_{\mathscr{S}} = \sum_{e=1}^{T} \sum_{n=1}^{N} \left(\hat{f}_{e,n}^E - f_{e,n}^E\right)^2, \tag{9}$$

where $f_{e,n}^E$ is the true value of the e^{th} order status attribute of order n, and $\hat{f}_{e,n}^E$ is the predicted value. By minimizing this loss function, the simulator learns to accurately reflect the dynamics of the system, providing valuable insights and guidance for decision-makers.

4.2 Decision Maker via Policy Neural Networks

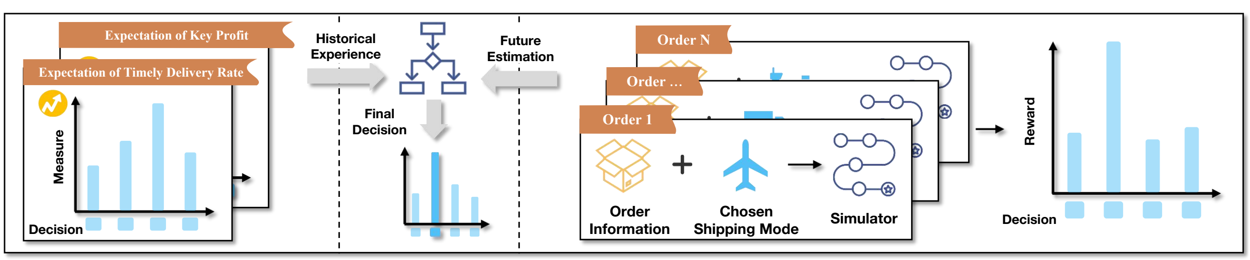


Figure 3: Decision-Maker

The simulator captures system patterns from real data and can flexibly reflect the state changes of the system under decision-making. However, simple observation is not enough to support the need to reduce decision risks in a traceable way in the supply chain, so we propose a decision-maker model in this section, which elevates the observation of the data set to the level of intervention. By modifying the strategy in a targeted manner and observing the counterfactual scenarios generated by the simulator, we can finally obtain a highly capable decision-maker model. Specifically, as shown in the Figure 3, our decision-makers combine the experience of historical decisions and the estimation of future benefits to give the final optimal decision. With the linkage to the simulator, the decision-makers have more experience and a clearer estimation of the benefits of the decision, and gradually iterate to learn better decision strategies.

Decision Network. We construct a learnable decision-making network $\mathscr{M}$ to generate the probability of each potential shipping mode for each order n:

$$\hat{\mathbf{R}}_n = \mathscr{M}(F_n^I;\Phi) = (R_n^1,\ldots,R_n^{|F^D|}), \tag{10}$$

where R_n^d denotes the probability to choose the d^{th} shipping mode.

Historical Experience. History contains rich experience, and we give decision networks the ability to learn from history. We encourage the selection of the shipping mode with the highest expected utility based on historical data. For each order event n, the decision-maker observes the system state F_n^I and outputs a probability distribution $\hat{\mathbf{R}}_n$. A decision $\hat{f}_n^D$ is then sampled from this distribution using the Gumbel-Softmax reparameterization method (Jang et al. 2017) to ensure the entire process is differentiable.

To evaluate each possible decision, we compute the expected values of T^{Timely} and T^{Profit} for that decision based on historical data. The reward $E_n^d(f_d^D)$ for taking decision f_d^D is defined as:

$$E^d(f_d^D) = \mathbb{E}[T^{Timely}|f_d^D] + \mathbb{E}[T^{Profit}|f_d^D]. \tag{11}$$

where $\mathbb{E}[T^{Timely}|f_d^D]$ and $\mathbb{E}[T^{Profit}|f_d^D]$ are the expectation of T^{Timely} and T^{Profit} when choose shipping mode f_d^D calculated from historical data. The decision network $\mathscr{M}$ is trained using the loss function, $\mathscr{L}_{\text{h}}$, which encourages selecting decisions with the highest expected reward:

$$\mathscr{L}_{\text{h}} = -\sum_{n=1}^{N}\sum_{d=1}^{|F^D|}\hat{\mathbf{R}}_n^d \cdot E^d(f_d^D), \tag{12}$$

where $\hat{R}_n^d$ is the probability of selecting decision f_d^D for order n, obtained from the probability distribution $\hat{R}_n^d = \mathscr{M}(F_n^I;\Phi)$. This loss formulation directly optimizes the parameters Φ of $\mathscr{M}$, training the network to prioritize the shipping mode that maximizes the expected combined $E^d(f_d^D)$ from historical experience.

Future Estimation. In addition to learning from experience, having a foresight into the future is also necessary for making good decisions. We model the decision-making task as a contextual bandit problem (Bouneffouf et al. 2020) to estimate the future chosen shipping mode of each order. This approach is compatible with scenarios where the computation of key measures is non-differentiable, such as retrieving profit or other system-level metrics from historical data.

The decision network $\mathscr{M}$ can be regarded as a value network that estimates the overall batch reward resulting from a particular shipping mode applied to an order. For each order n, the chosen shipping mode also sampled from the distribution $\hat{\mathbf{R}}_n$ which can be understood as a normalized reward score:

$$\hat{f}_{d,n}^D = \text{GumbelSoftmax}(\sigma(\mathscr{M}(F_n^I;\Phi))), \tag{13}$$

Herein, $\hat{f}_{d,n}^D$ denotes the shipping mode chosen for order n^{th} is d^{th} shipping mode. For each order n, the simulator predicts statuses $\hat{f}_{e,n}^E$ based on the inherent attributes F_n^I and the selected decision $\hat{f}_{d,n}^D$:

$$\hat{F}_n^E = \mathscr{S}(F_n^I, \hat{f}_{d,n}^D). \tag{14}$$

The simulator is pre-trained, with its parameters frozen during decision-making. This design ensures that the simulator provides a stable and reliable risk-free environment, allowing us to freely explore and evaluate a wide range of decisions without concern for real-world consequences. The orders are grouped according to the selected shipping mode $\hat{f}^D_{d,n}$. We calculate the timely delivery rate and profit for each shipping mode using the necessary information, including order information, selected shipping mode, and the simulated order status. We use T_d^{Timely} and T_d^{Profit} to denote the timely delivery rate and the profit for those orders that choose the shipping mode f^D_d. The total reward for shipping mode f^D_d in a batch is computed by:

$$R^d_{\text{batch}} = T_d^{Timely} + T_d^{Profit}. \tag{15}$$

The value network is trained to minimize the prediction error of batch rewards by optimizing the following loss:

$$\mathscr{L}_{\text{f}} = \frac{1}{N}\sum_{n=1}^{N}\sum_{d=1}^{|F^D|}\left(\hat{R}^d_n - R^d_{\text{batch}}\right)^2. \tag{16}$$

Through iterative feedback, the decision-maker network refines its understanding of the relationship between individual decisions and their impact on batch-level performance.

Combined Loss Function and Decision-Making. To ensure that the model can benefit from both historical experience and future estimates, the overall loss function combines the objectives of the two levels:

$$\mathscr{L}_{\mathscr{M}} = \mathscr{L}_{\text{f}} + \lambda \cdot \mathscr{L}_{\text{h}}, \tag{17}$$

where λ is the hyperparameter that balances the importance of historical experience and future predictions.

Through the integration of two optimization objectives, we develop a multi-perspective decision network $\mathscr{M}$. This network effectively combines lessons learned from history with visionary predictions for the future. Given the inherent attributes F^I_n for each order n, $\mathscr{M}$ outputs a value for each potential decision f^D_d, denoted as $\hat{R}_n = \mathscr{M}(F^I_n;\Phi)$. This value can be interpreted as either the probability or the reward associated with each shipping mode. The final decision $\hat{f}^D_d$ is determined by selecting the decision corresponding to the maximum value output by the network:

$$\hat{f}^D_d = \arg\max_{f^D_d \in F^D} \mathscr{M}(F^I_n;\Phi). \tag{18}$$

5 EXPERIMENTS

5.1 Experimental Setting

Datasets Description. We conducted experiments on three real-world supply chain datasets: DataCo (Fabian Constante and Fernando Silva and António Pereira 2019), Global-Store (Anandaram G 2025), and OAS (Vinay34 2024). These datasets cover different logistics and transportation settings, including various shipping modes. Each dataset includes detailed order records, such as order information, selected shipping modes, and delivery statuses, providing a realistic testbed for evaluating Sim-to-Dec. To ensure a fair and consistent evaluation, each dataset is randomly divided into training, validation, and test sets in an 8:1:1 ratio. This split helps mitigate potential temporal biases and improve generalization by avoiding overfitting to chronological patterns. To further reduce overfitting risk, we apply standard regularization techniques, including early stopping based on validation loss and ℓ_2 weight penalties during training. Both the simulator and the decision network are trained only on the training set, and the simulation performance and post-decision metrics on the test set are reported. Each experiment is repeated five times with different random seeds, and we report average results. The dataset statistics are summarized in the first row of Table 1 in the form (#Features, #Instances).

Evaluation Metrics. For the simulator, we evaluate its accuracy by comparing the predicted order status attributes (delay risk, delivery time, and on-time status) against the ground-truth values in the test set. Accuracy is computed for each attribute individually, and an overall accuracy is reported as the unweighted average across the three prediction tasks. This provides a holistic measure of the simulator's fidelity. For the decision-maker, we compare the average profit and on-time rate of test set orders. We normalize both

Table 1: Comparison of simulation results. Overall represents the average accuracy of all evolutionary features.

Dataset	Dataco (43, 165445)				GlobalStore (27, 51290)				OAS (22, 28136)			
Method	Markov	Prediction	Generation	Sim-to-Dec	Markov	Prediction	Generation	Sim-to-Dec	Markov	Prediction	Generation	Sim-to-Dec
f^E_{risk}	0.4978	0.7019	<u>0.7024</u>	**0.9508**	0.4961	0.8440	<u>0.9366</u>	**0.9743**	0.5100	<u>0.7157</u>	0.7149	**0.7215**
f^E_{time}	0.1487	0.3395	<u>0.3485</u>	**0.8851**	0.1355	0.6767	<u>0.8066</u>	**0.9255**	0.0011	0.3706	<u>0.3916</u>	**0.3985**
f^E_{status}	0.5040	<u>0.8161</u>	0.8156	**0.9695**	0.4934	0.8430	<u>0.9355</u>	**0.9756**	0.5068	<u>0.7510</u>	0.7503	**0.7574**
Overall ↑	0.3835	0.6191	<u>0.6221</u>	**0.9351**	0.3750	0.7879	<u>0.8929</u>	**0.9585**	0.3393	0.6124	<u>0.6189</u>	**0.6258**

indicators to the [0, 1] range, and report two aggregate metrics: the absolute difference between the two objectives (Diff), and their sum (Overall), to comprehensively assess decision quality.

Baseline Algorithms. To evaluate simulation methods, we selected three paradigms to simulate the supply chain: (1) *Markov-based simulation* (Gagniuc 2017), representing traditional approaches that use state-transition probabilities to model system dynamics; (2) *Prediction-based simulation* (Caruana 1997), which adopts a multi-task framework to predict evolutionary features individually based on input conditions; and (3) *Non-autoregressive generation-based simulation* (Gu et al. 2018), which generates multiple evolutionary features simultaneously in one step. For simplicity, we refer to this paradigm as *Generation* in the following sections. For decision-making based on simulation, we selected three paradigms: (1) *Linear Programming (LP)* (Dantzig 2002), a traditional optimization method for solving predefined decision problems; (2) *Reinforcement Learning*, where a DQN-based RL agent iteratively optimizes strategies through interaction with the environment; and (3) *LLM-based decision-making* (Brown et al. 2020), which leverages the expert knowledge embedded in ChatGPT-3.5, a large language model (LLM), to make decisions under a zero-shot setting without task-specific fine-tuning.

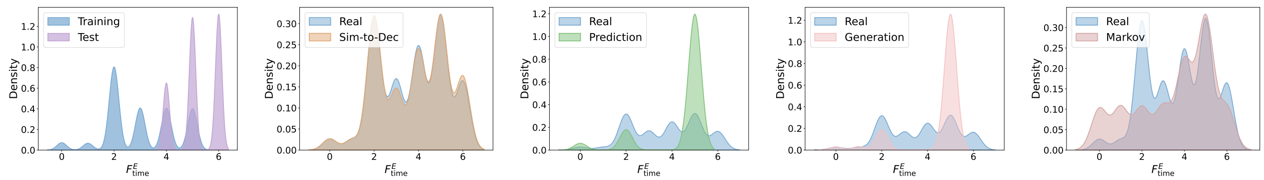
5.2 Experimental Results

A Study of Generative Simulator Accuracy. As shown in Table 1, we compare our model with other baselines on three real-world supply chain datasets. We have the following observations: (1) Sim-to-Dec outperforms all baseline methods on all datasets. Specifically, in terms of overall accuracy, our method improves the strongest baseline by 50.3%, 7.3%, and 1.1% in DataCo, GlobalStore and OAS respectively. We attribute this to our unique insight that enables the simulator to sequentially reproduce realistic changes to the system at a fine-grained level. (2) Compared with simulation methods based on Markov chains, data-driven simulation has improved accuracy. For example, on the DataCo dataset, prediction and generation and Sim-to-Dec have improved 61.4%, 62.2% and 143.8% respectively. This shows that data-driven methods can mine the potential laws in the data to better model the system operation.

Table 2: Comparison of decision-making results.

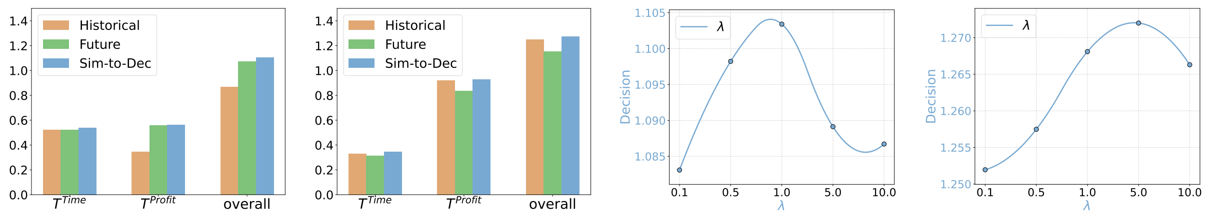
Method	DataCo				Global-Store				OAS			
	T^{Timely} ↑	T^{Profit} ↑	Diff ↓	Overall ↑	T^{Timely} ↑	T^{Profit} ↑	Diff ↓	Overall ↑	T^{Timely} ↑	T^{Profit} ↑	Diff ↓	Overall ↑
Real	0.5244	0.0364	0.4880	0.5608	0.3320	0.0848	**0.2472**	0.4168	0.4800	0.0000	0.4800	0.4800
LP	0.5162	0.5434	<u>0.0272</u>	<u>1.0596</u>	0.3552	0.6001	<u>0.2449</u>	0.9554	0.5037	0.1043	<u>0.3994</u>	<u>0.6080</u>
RL	0.5276	0.2071	0.3205	0.7347	0.2827	0.9326	0.6499	<u>1.2153</u>	0.4817	0.0000	0.4817	0.4817
LLM	0.5258	0.2459	0.2800	0.7717	0.3298	0.0439	0.2859	0.3736	0.4844	0.0000	0.4844	0.4844
Sim-to-Dec	0.5397	0.5637	**0.0240**	**1.1034**	0.3446	0.9278	0.5828	**1.2724**	0.4882	0.1611	**0.3271**	**0.6493**

Robustness Check of Generative Simulator against Distribution Shift. The dynamic complexity of the real world means that the environment in which the system operates is constantly changing. Mining potential patterns from data and being able to accurately simulate when the operating environment changes or is disturbed is an ideal property of a robust simulator. We designed an experiment to verify the performance of our method in this scenario. We repartitioned the DataCo dataset according to f^E_{time}, so that the data distribution of the training set and the test set shifted, as shown in Figure 4(a). We performed simulations in this changing environment. As shown in Figure 4(b)-(e), we compared the f^E_{time} distributions generated



(a) Training&Test (b) Sim-to-Dec (c) Prediction (d) Generation (e) Markov

Figure 4: Simulation under distribution shift on DataCo dataset



(a) Decision-Maker Ablation on DataCo (b) Decision-Maker Ablation on GlobalStore (c) Sensitivity Analysis on DataCo (d) Sensitivity Analysis on GlobalStore

Figure 5: Ablation Study and Sensitivity Analysis

by different simulation methods with the distribution in the real test set. The experimental results show that our method can still capture the underlying patterns of the data when the distribution shifts, and deduce evolutionary variables that conform to the actual situation, verifying the practicality of our method.

A Study of Decision-Maker. We first compare the performance of different decision-makers when improving key metrics at the same time. As shown in Table 2, we have the following observations: (1) Our method shows the best performance when considering both the overall measure (Overall) and the gap between the two measures (Diff). For example, compared with the strongest baseline, our method improves on overall by 4.1% and 6.8%, and the Diff is reduced by 11.8% and 18.1% on the DataCo and OAS datasets. This shows that the decision-maker we proposed has strong decision-making ability and can balance conflicting optimization goals to achieve common improvement. (2) LP performs well on some datasets. However, its performance is achieved by enumerating all possible strategies and their associated rewards on the test set, rather than learning a generalizable policy during training. While this approach leverages more task-specific information, it significantly limits the method's generalization capability and results in substantial computational overhead, making it impractical for large-scale scenario.

5.3 In-depth Analysis: Ablation Studies, Parameter Sensitivity, Computational Efficiency

We evaluate the time complexity of our framework and baseline methods. Tables 3 shows that our simulator demonstrates competitive training efficiency. Compared with lightweight but less expressive baselines like Markov models, our method achieves fine-grained learning in a reasonable time, balancing accuracy and runtime cost. Figure 5(a) and 5(b) show that decisions based on historical experience and future estimates work differently on different data sets, but a combination of the two usually leads to better

Table 3: Time Cost of Simulator Training

Dataset	Method	Time/Epoch (s)	# Epoch	Total Time (s)
DataCo	Markov	-	-	53
	Prediction	0.86	60	70
	Generation	0.74	110	81
	Sim-to-Dec	0.90	350	315
GlobalStore	Markov	-	-	14
	Prediction	0.12	210	25
	Generation	0.19	200	38
	Sim-to-Dec	0.24	200	48

decisions. As shown in Figure 5(c) and 5(d), we use λ to balance the proportion of historical experience and future estimates in decisions. As λ increases, the performance first increases and then decreases, indicating that a suitable λ should be selected so that the decision can benefit from both perspectives. For decision-making, our method requires only 1.2 seconds on the DataCo dataset and 0.2 seconds on Global-Store, significantly faster than the LLM-based approach (1856s and 655s). LP methods take 8s and 2s respectively, while RL is fastest (0.5s and 0.1s) but less effective in decision quality. Overall, our approach offers a strong trade-off between efficiency and performance.

6 CONCLUSION

In this work, we propose a unified framework that tightly integrates a *generative simulator* with a *feedback-driven decision-maker* to improve responsiveness in supply chain transportation. The simulator models order dynamics through autoregressive learning, enabling fine-grained prediction of shipment evolution under different transportation strategies. The decision-maker iteratively refines shipping mode selection by combining historical patterns with forward-looking reward estimation, guided by simulated feedback. This tight simulation–decision coupling overcomes the limitations of static models and manual heuristics, providing enhanced flexibility and adaptability in dynamic logistics environments. Extensive experiments on real-world supply chain datasets demonstrate the superiority of our approach in balancing timely delivery and cost efficiency, even under distribution shifts. These results underscore the potential of our framework for broader applications in logistics optimization and adaptive decision-making systems.

ACKNOWLEDGMENTS

This work was supported by the U.S. National Science Foundation under Grant Nos. 2426340, 2416727, 2421864, 2421865, and 2421803. The authors thank all members of the research group for their valuable feedback and insightful discussions during the development of this work.

REFERENCES

Adobor, H. 2020. "Supply chain resilience: an adaptive cycle approach". *The International Journal of Logistics Management* 31(3):443–463.

Bai, H., G. Chen, W. Ying, X. Wang, N. Gong, S. Dong, *et al*. 2025. "Brownian Bridge Augmented Surrogate Simulation and Injection Planning for Geological CO _2 Storage". *arXiv preprint arXiv:2505.18204*.

Barykin, S. Y., A. A. Bochkarev, O. V. Kalinina, and V. K. Yadykin. 2020. "Concept for a supply chain digital twin". *International Journal of Mathematical, Engineering and Management Sciences* 5(6):1498–1515.

Bi, M., G. Chen, D. M. Tilbury, S. Shen, and K. Barton. 2022. "A Model-based Multi-agent Framework to Enable an Agile Response to Supply Chain Disruptions*". In *18th IEEE International Conference on Automation Science and Engineering, CASE 2022, Mexico City, Mexico, August 20-24, 2022*, 235–241: IEEE https://doi.org/10.1109/CASE49997.2022.9926559.

Bouneffouf, D., I. Rish, and C. C. Aggarwal. 2020. "Survey on Applications of Multi-Armed and Contextual Bandits". In *IEEE Congress on Evolutionary Computation, CEC 2020, Glasgow, United Kingdom, July 19-24, 2020*, 1–8: IEEE https://doi.org/10.1109/CEC48606.2020.9185782.

Brailsford, S. C., T. Eldabi, M. Kunc, N. Mustafee, and A. F. Osorio. 2019. "Hybrid simulation modelling in operational research: A state-of-the-art review". *European Journal of Operational Research* 278(3):721–737.

Brown, T. B., B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, *et al*. 2020. "Language Models are Few-Shot Learners". In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, edited by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin.

Caruana, R. 1997. "Multitask learning". *Machine learning* 28:41–75.

Chen, Y., S. Wang, L. Zhang, and S. Zhang. 2024. "Exploration of an Intelligent Decision-making System for International Freight Forwarding Based on Simulation Optimization". In *Proceedings of the 2024 9th International Conference on Intelligent Information Processing*, 251–256.

Churchman, C. W., R. L. Ackoff, and E. L. Arnoff. 1957. "Introduction to operations research.".

Fabian Constante and Fernando Silva and António Pereira 2019. "DataCo SMART SUPPLY CHAIN FOR BIG DATA ANALYSIS" https://doi.org/10.17632/8gx2fvg2k6.5.

Correa-Martinez, Y. C., and M. Seck. 2023. "A generic representation of supply network resilience using simulation based experimentation". *Journal of Simulation* 17(3):326–359.

Dantzig, G. B. 2002. "Linear programming". *Operations research* 50(1):42–47.

Daroń, M. 2022. "Simulations in planning logistics processes as a tool of decision-making in manufacturing companies". *Production Engineering Archives* 28(4):300–308.

Dimov, I. T. 2008. *Monte Carlo methods for applied scientists*. World Scientific.

Dixit, V., P. Verma, and M. K. Tiwari. 2020. "Assessment of pre and post-disaster supply chain resilience based on network structural parameters with CVaR as a risk measure". *International Journal of Production Economics* 227:107655.

Du, S. S., S. M. Kakade, R. Wang, and L. F. Yang. 2020. "Is a Good Representation Sufficient for Sample Efficient Reinforcement Learning?". In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*: OpenReview.net.

Forrester, J. W. 1997. "Industrial dynamics". *Journal of the Operational Research Society* 48(10):1037–1041.

Anandaram G 2025. "Global Superstore".

Gagniuc, P. A. 2017. *Markov chains: from theory to implementation and experimentation*. John Wiley & Sons.

Gu, J., J. Bradbury, C. Xiong, V. O. K. Li, and R. Socher. 2018. "Non-Autoregressive Neural Machine Translation". In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*: OpenReview.net.

Hosseini, S., D. Ivanov, and A. Dolgui. 2019. "Review of quantitative methods for supply chain resilience analysis". *Transportation research part E: logistics and transportation review* 125:285–307.

Huang, C., G. Wang, Z. Zhou, R. Zhang, and L. Lin. 2022. "Reward-adaptive reinforcement learning: Dynamic policy gradient optimization for bipedal locomotion". *IEEE transactions on pattern analysis and machine intelligence* 45(6):7686–7695.

Jang, E., S. Gu, and B. Poole. 2017. "Categorical Reparameterization with Gumbel-Softmax".

Kristofik, P., J. Kok, S. de Vries, and J. van Sten-van't Hoff. 2012. "Financial supply chain management–Challenges and obstacles". *Proceedings in Finance and Risk Perspectives* 12.

Law, A. M., W. D. Kelton, and W. D. Kelton. 2007. *Simulation modeling and analysis*, Volume 3. Mcgraw-hill New York.

Rolf, B., I. Jackson, M. Müller, S. Lang, T. Reggelin, and D. Ivanov. 2023. "A review on reinforcement learning algorithms and applications in supply chain management". *Int. J. Prod. Res.* 61(20):7151–7179 https://doi.org/10.1080/00207543.2022.2140221.

Routroy, S., and A. Behera. 2017. "Agriculture supply chain: A systematic review of literature and implications for future research". *Journal of Agribusiness in Developing and Emerging Economies* 7(3):275–302.

Saisridhar, P., M. Thürer, and B. Avittathur. 2024. "Assessing supply chain responsiveness, resilience and robustness (Triple-R) by computer simulation: a systematic review of the literature". *Int. J. Prod. Res.* 62(4):1458–1488 https://doi.org/10.1080/00207543.2023.2180302.

Sutton, R. S., and A. G. Barto. 1998. "Reinforcement Learning: An Introduction". *IEEE Trans. Neural Networks* 9(5):1054–1054 https://doi.org/10.1109/TNN.1998.712192.

Vinay34 2024. "Supply Chain Analysis Dataset".

Yu, Y., X. Wang, R. Y. Zhong, and G. Q. Huang. 2017. "E-commerce logistics in supply chain management: Implementations and future perspective in furniture industry". *Industrial Management & Data Systems* 117(10):2263–2286.

Zhou, C., H. Li, W. Liu, A. Stephen, L. H. Lee, and E. P. Chew. 2018. "Challenges and opportunities in integration of simulation and optimization in maritime logistics". In *2018 Winter Simulation Conference (WSC)*, 2897–2908. IEEE.

AUTHOR BIOGRAPHIES

HAOYUE BAI is a PhD student at Arizona State University. His email address is baihaoyue621@gmail.com.

HAOYU WANG is a researcher in the DSSS department at NEC Labs America. His email address is haoyu@nec-labs.com.

NANXU GONG is a PhD student at Arizona State University. His email address is nanxugong@outlook.com.

XINYUAN WANG is a PhD student at Arizona State University. His email address is wangxinyuan0921@gmail.com.

WANGYANG YING is a PhD student at Arizona State University. His email address is yingwangyang@gmail.com.

HAIFENG CHEN is the head of the DSSS at NEC Laboratories America. His email address is haifeng@nec-labs.com.

YANJIE FU is an associate professor at Arizona State University. His e-mail address is yanjie.fu@asu.edu.