Advancing Sustainability in Global Supply Chains through Agent-based Simulation

Haoyu Wang
NEC Labs America
Princeton, USA
haoyu@nec-labs.com

Erhu He, Can Zheng
University of Pittsburgh
Pittsburgh, USA
{erh108,caz51}@pitt.edu

Lu-An Tang, Zhengzhang Chen, Xujiang Zhao
NEC Labs America
Princeton, USA
{ltang,zchen,xuzhao}@nec-labs.com

Xiaowei Jia
University of Pittsburgh
Pittsburgh, USA
xiaowei@pitt.edu

Haifeng Chen
NEC Labs America
Princeton, USA
haifeng@nec-labs.com

***Abstract*—In today's world, with its complex global supply chains, the difficulties and uncertainties we face offer both challenges and opportunities for making things better, especially in terms of efficiency and sustainability. These challenges grow due to unpredictable events, such as natural disasters, unexpected incidents, and unusual business practices, pushing us towards more advanced modeling methods that focus on reducing risks and enhancing sustainability. In this paper, we present a new agent-based simulation approach that goes beyond the usual limits of supply chain simulations by incorporating sustainability directly into supply chain operations using Reinforcement Learning (RL) algorithms. We introduce MOGI, derived from the Japanese word for 'simulation', a sustainable supply chain simulation system that takes carbon emissions into account in its main operations. Additionally, we examine how effective a multi-agent RL strategy is in dealing with the complex and uncertain nature of supply chains that span multiple levels. By comparing this strategy with traditional heuristic methods, our study looks at how well single versus multiple RL agents can manage risks and improve sustainability in both the beginning and end parts of the supply chain. The results of our experiments show that strategies based on RL are much better than traditional methods at managing risks, making profits, and achieving sustainability goals.**

***Index Terms*—Agent-based Simulation, Supply Chain, System Optimization**

I. Introduction

In the evolving landscape of global Supply Chain Management (SCM), mitigating carbon emissions has emerged as a critical concern. This imperative addresses not only environmental sustainability but also operational efficiency and regulatory compliance. The complexity of modern supply chains, characterized by intricate networks of suppliers, manufacturers, and retailers across diverse regions, poses significant challenges in accurately quantifying and managing carbon footprints. With ambitions toward achieving a net-zero economy, numerous countries are adopting varied sustainability policies [2], [3]. To meet internationally agreed-upon climate goals, optimizing supply chain management by integrating carbon emissions considerations is essential.

Efforts to reduce carbon footprints in supply chain management necessitate a comprehensive approach that incorporates robust strategies to address traditional uncertainties while actively striving for sustainability and carbon neutrality. This approach ensures that supply chains not only achieve their economic objectives but also contribute positively to environmental stewardship. Machine Learning (ML) has become increasingly prevalent in enhancing SCM, particularly in improving demand forecasting and sales predictions [7], [12], [14], estimating commercial partnerships [10], [11], and optimizing inventory management [9], [15]. However, reliance solely on ML techniques presents certain limitations, including the lack of transparency in the decision-making process and the intensive requirements for training data and computational resources. These challenges necessitate either advanced domain expertise for developing sophisticated and experiential strategies or substantial datasets for model training, which can be particularly challenging to acquire, especially in the context of proprietary commercial data.

Considering the identified limitations of ML methods in addressing supply chain challenges, an increasing number of researchers are integrating simulation-based methods with ML to tackle these issues [1], [5]. To this end, we introduce MOGI, a simulation tool tailored for general complex problems. MOGI encompasses three critical components: a comprehensive agent-based simulation engine, a resource management system, and an interactive platform for the implementation and testing of policies. Designed for efficiency and scalability, MOGI is adept at simulating complex scenarios involving numerous agents and complex resource flows. Owing to its capability to monitor every detail of each component within the simulation framework, MOGI facilitates the calculation of product-level carbon emissions with a precision that surpasses previous methods.

Reinforcement Learning (RL) has emerged as a critical technique for optimizing agent-based simulations in supply chain management, attributed to its unparalleled capability to navigate complex, uncertain environments. Firstly, supply

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-169-5

chain management necessitates sequential decision-making amidst uncertainty, a domain where RL excels by optimizing decisions across time to favor long-term rewards over immediate gains. This approach is vital for supply chain decisions, considering that short-term actions may lead to enduring consequences. Secondly, RL can model and learn complex behaviors directly from agent-interaction data, obviating the need to explicitly enumerate all conceivable states and actions—a task impractical for complex systems. In this study, we apply RL to each simulation agent and assess various RL algorithms to facilitate optimization of supply chain management towards achieving carbon neutrality.

The contributions of this paper are summarized as follows:

- We have developed an agent-based supply chain simulator, MOGI, capable of simulating detailed interactions among supply chain components, with an emphasis on sustainability attributes.
- We investigate the potential and limitations of multi-agent reinforcement learning algorithms in reducing supply chain uncertainty by extending the supply chain to include participants across various tiers.

This paper is organized as follow. Section II introduces the previous work about supply chain system simulation and reinforcement learning. In Section III, we defined the supply chain system optimization problem. In Section IV, we introduce MOGI simulator in detail including all the key components. In Section V and Section VI, we introduce the reinforcement learning used in MOGI simulation evaluation and the experiments respectively. In Section VII and Section VIII, we conclude this paper and introduce the future work.

II. Related work

Agent-based simulation tools are increasingly employed in supply chain management to facilitate the exploration of complex interactions among individual agents, which may represent companies, consumers, or products, within the supply chain network. Tools such as AnyLogic [1], Simio [5], and MATSim [4] exemplify agent-based supply chain simulation platforms. Nevertheless, these tools do not explicitly focus on sustainability within the supply chain, nor do they extend to the precise calculation of product-level carbon emissions.

Since there was a lack of relevant research on supply chain management in sustainability, based on our exploration, the closest previous work is the application of RL in inventory management. In this type of problem, the RL-agent first observes the current state of the system, including current inventory levels, demand patterns, lead times, etc. The RL-agent is then required to determine order quantities or reorder points, and the environment responds by generating new states and providing rewards or penalties to guide the learning process. As a typical case of downstream uncertainty, the variant demands are considered as the drive for reinforcement learning solution in many researches, and the adaptive balance between customer satisfaction and storage cost need to be found. Zwaida [6] propose an online solution with deep Q-network (DQN) algorithm to prevent drug shortage problem in hospital by deciding the refilling time and the amount of ordered drugs, balancing the shortage cost and overstock cost. Ganesan et al. [8] train the RL-agent to select the optimal strategy from five pre-defined policies by considering the combination of shortages, frequency of shortages and surplus inventories over the past n periods. Sedamaki et al. [13] classify suppliers to four risk indices and train the RL-agent in a custom-modeled environment to slit an order among multiple suppliers while minimizing the delays.

III. Problem Definition

The goal of our work is to develop a simulation tool to model supply chain system behaviors with a focus on sustainability, aiming to optimize supply chain decision-making for lower carbon emissions and reduced uncertainty. To assess the MOGI simulation tool and the RL optimization methods for system sustainability, we transform a real-world supply chain system focusing on sustainability into an optimization problem aimed at lowering carbon emissions. Consider a supply chain system with I retailers (agents in the simulation) interconnected in a specific topology. At time point t within period T, the i^{th} retailer purchases $m_{ij}(t)$ units of the product from the j^{th} supplier at price $n_{ij}(t)$. Subsequently, the i^{th} retailer sells $m_{ik}(t)$ units of the product to the k^{th} customer at price $n_{ik}(t)$. Given that carbon emissions are predominantly calculated during the manufacturing process, the carbon emissions associated with the transaction between the i^{th} retailer and the j^{th} supplier are represented as $m_{ij} * E_{ij}$, where E_{ij} is the product-specific carbon emission factor. Thus, the objective is to maximize the profit earned by all retailers while accounting for the equivalent carbon emissions, as illustrated below:

$$max \sum^{T} (\sum^{I} m_{ij}(t) * n_{ij}(t) - \sum^{K} E_{ik} * m_{ij}(t)). \quad (1)$$

For clarification, we use the term “agent” in this article to refer to the agent both in simulation system and real-world environment and “RL-agent” to refer to the agent in RL only.

IV. MOGI: Supply Chain Simulation

In this section, we introduce MOGI, an agent-based simulation tool designed for general complex system modeling. This paper demonstrates the application of MOGI in supply chain management with a focus on sustainability. We use the supply chain system as a case study to elucidate MOGI’s rationale and the methodology for mapping real-world systems into the simulation environment.

A. Overview of MOGI Simulation

A general complex system is comprised of interacting, autonomous components. Unlike simple systems, complex adaptive systems possess the ability for agents to adapt at the individual or population levels. This exploration into complex systems forms the basis for understanding self-organization, emergent phenomena, and the origins of adaptation in nature. Conceptually, the decomposition of a general complex system into three primary components—Agents, Resources, and Topology—is derived from a holistic approach to modeling and comprehending the intricate interactions and dynamics within such systems. Agents within the system have the capacity to act, interact, and make decisions based on predefined rules or through adaptive learning mechanisms. Resources include the various elements and assets that can be consumed, transformed, or produced by agents within the system. Topology refers to the arrangement and connectivity of elements within the system, highlighting the structural aspect of complex systems. It delineates how agents are linked and the manner in which they can interact with one another. This framework not only facilitates the conceptual understanding of complex systems but also enables structured simulations to explore system dynamics, predict behavior under diverse scenarios, and devise interventions to achieve specific objectives. The diagram shown in Fig. 1 exemplifies MOGI's functionality, orchestrating resource flow dynamically through the supply chain. The module is pivotal, facilitating the simulation of diverse supply chain strategies and their impacts on efficiency and sustainability.

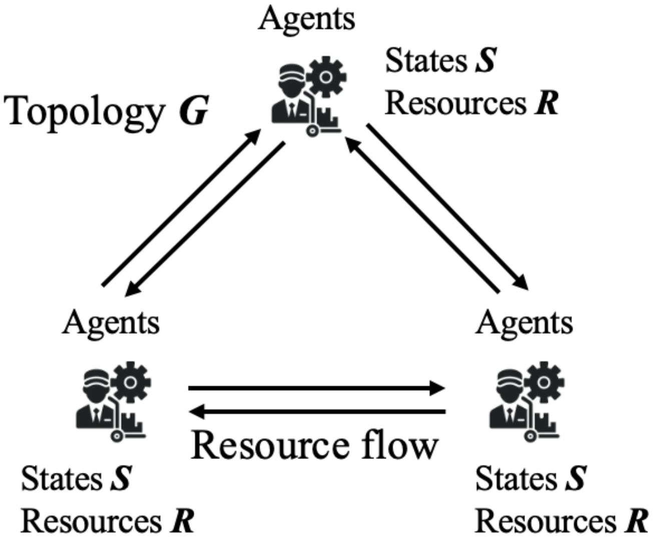
Agents
States $\boldsymbol{S}$
Resources $\boldsymbol{R}$
Topology $\boldsymbol{G}$
Agents
Agents
Resource flow
States $\boldsymbol{S}$
Resources $\boldsymbol{R}$
States $\boldsymbol{S}$
Resources $\boldsymbol{R}$

Fig. 1. MOGI in supply chain simulation.

B. Agent

Agents are entities within the system capable of acting, interacting, and making decisions based on predefined rules or adaptive learning mechanisms. Each agent is endowed with the ability to process information, utilize resources, and potentially alter the topology through their actions. The complexity of real-world systems emerges from the collective behaviors of agents, leading to phenomena such as self-organization, adaptation, and evolution.

The design and description of agents within a simulation are predicated on several essential characteristics. First, an agent is a self-contained and uniquely identifiable entity with attributes that enable it to be distinguished from and recognized by other agents, facilitating interaction. Second, an agent is autonomous and self-directed, capable of operating independently within its environment and in interactions with other agents. An agent's behavior, which bridges sensed information to decisions and actions, can range from simple rules to complex models, including RL mechanisms that adapt inputs to outputs. Third, an agent possesses a state that evolves over time or in response to external changes. In MOGI, we employ a state machine mechanism within each agent to represent its state. This mechanism is chosen for its inherent ability to model the discrete states and transitions that define the operational and decision-making processes of agents. In the context of supply chain systems, this approach is particularly apt, as it mirrors the operational stages and decision-making sequences in procurement and manufacturing processes, among others.

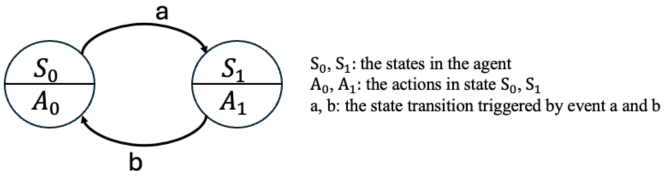
a
S_0
A_0
S_1
A_1
b
S_0, S_1: the states in the agent
A_0, A_1: the actions in state S_0, S_1
a, b: the state transition triggered by event a and b

Fig. 2. State machine in MOGI agent.

A general example of a state machine within an agent is depicted in Fig. 2. This figure shows a state machine comprising two states: State 1 and State 2. Each state triggers a specific action, denoted as Action 1 and Action 2, respectively. The diagram also illustrates a uni-directional transition from State 1 to State 2, initiated by a designated event. This transition symbolizes a shift in behavior, as indicated by the distinct actions associated with each state.

Within MOGI, agents function according to a behavioral model that promotes autonomy and responsiveness to other agents. This model incorporates decision-making algorithms that enable agents to adapt to the evolving conditions of the simulation environment, thus mirroring the uncertainties and dynamics typical of real-world supply chain operations. Agents evaluate their performance metrics, such as delivery times and production rates, and adjust their strategies accordingly to optimize these variables. The model guarantees that agents' actions are responsive to changes in resource availability and demand, creating a self-regulating system that adapts based on simulation inputs and inter-agent interactions.

C. Resource

The resource component manages the both tangible and intangible resources within the connections among all the agents or produced by agents, employing algorithms that adapt to simulation conditions. Resources are allocated based on supply and demand, with the simulation tracking their utilization and wastage. It also simulates the exchange of resources among agents, incorporating factors like market trends and demand forecasts. It ensures a balance between resource consumption and replenishment, aligning with the sustainability metrics modeled within the simulation. Resource dynamics, such as scarcity, competition, and allocation, play a critical role in the agent's behavior and interactions and consequently, in the emergent properties of the system.

The nature and dynamics of resource can greatly impact agent behavior especially under different interaction such as cooperation and competition. For the **cooperation**, the agents work together to share, allocate, or optimize resource in the same direction such as shared benefit or similar goal. Resource in such settings should be designed to encourage collaborative strategies, such as pooling resource to complete a task that no single agent could accomplish alone. Meanwhile, the simulation can explore how cooperation leads to efficient resource use and mechanisms for fair distribution and sustainability. For **competition**, the competitive resource settings can simulate the real-world phenomena such market dynamics, ecological survival strategies, or social competition. The focus can be on how agents adapt the strategies in response to resource scarcity, the impact of competition on resource distribution.

D. Topology

The topology component in MOGI simulates the dynamic connections and interactions between agents and resources. It concerns the arrangement and connectivity of elements within the system, highlighting the structural dimension of complex systems. Topology determines how agents are interconnected, thereby influencing their potential interactions. The configuration of a system's topology plays a crucial role in its dynamics by dictating the channels for information or resource flow and impacting overall system performance. As interactions between agents and resources unfold, the topological structure adapts, shedding light on optimal system configurations.

Topology within simulations can be categorized into **static/dynamic** and **physical/virtual**, accommodating various real-world system types. **Static topology** features a spatial structure that remains constant throughout the simulation period, streamlining the analysis of agent interactions and the influence of spatial arrangements on system dynamics. It suits the study of systems with stable spatial relationships over time, such as those in organization-based simulations, allowing a concentrated examination of other dynamics.

Conversely, **dynamic topology** supports modifications to spatial structures during the simulation, including changes in agent positions, modifications in agent connections, or variations in spatial configurations. This type of topology is crucial for simulating systems where adaptability, movement, or structural changes are integral to behavior, exemplified by social network evolution simulations.

Physical topology deals with the spatial arrangement of agents and resources, taking into account distances, barriers, or spatial distributions that influence interaction probabilities and dynamics. It is applied to simulate real-world spatial dynamics, such as urban traffic patterns. **Virtual topology**, on the other hand, defines connections among agents based on relationships, communication paths, or other non-physical links. It is vital for studying systems where physical locations are secondary to the connections between entities, as seen in simulations of idea development or virtual networks.

V. Optimization Method

A. Scenario Setup

In this study, our objective is to investigate the impact of supply chain depth on uncertainty management, focusing on a scenario that incorporates multiple suppliers and customers. At the heart of this scenario is an intermediary entity, such as a retailer, which is represented by a decision-making agent (as depicted in Fig. 3). This agent aims to maximize profits through strategic purchasing and selling activities, with a keen consideration for sustainability, herein represented by carbon emissions.

The initial simulated scenario involves three suppliers connected to a central agent, which in turn is connected to three customers (as illustrated in Fig. 3a). These connections symbolize contracts established for the trading of products. To inject an element of uncertainty into the simulation, the connection between a given supplier i and the central agent j may become disabled with a certain probability p_{ij} at any given time step t. Customer demand directed towards agent j is modeled as a random variable that follows a Poisson distribution, represented by d_j. This setup allows the agent to purchase products from suppliers at a quoted price and then sell them to distributors at a price determined by the agent, effectively simulating the dynamic and uncertain nature of customer demand.

To further examine the influence of having multiple multi-level agents in the supply chain on sustainability, the scenario is expanded as shown in Fig. 3 (b). In this more complex setup, multiple agents share the uncertainties, each facing a disruption probability p_{ii} when interacting with another agent. The demand requested by a downstream agent is denoted as d_{ik}, illustrating the extended network and layered interactions designed to explore deeper aspects of supply chain sustainability and uncertainty management.

B. RL method

Because of the traditional optimization methods that struggle to cope with the stochastic nature and the high dimensionality of decision spaces, we apply RL-agent on the agent to make decisions due to its capability to learn optimal strategies through interaction with a real-world system. Meanwhile, the adoption of RL can learn from simulation without real-world risks, deal with uncertainty and partial observability,and facilitate continuous improvement. Next, we introduce the detail of RL (DQN) (as shown in Fig. 4) applied in MOGI.

The learning process of RL-agent can be described by a tuple $(s, a, s', r,)$, where s denotes the current state, a denotes the action will take, s' and r is the new state and reward returned from the environment, respectively, once a is acted 4. We also denote S and A as the set of possible states and actions respectively. The Q-learning algorithm aims to estimate the action by mapping state and corresponding action, $Q : S \times A \to \mathbb{R}$, to a real number so called Q-values. The agent

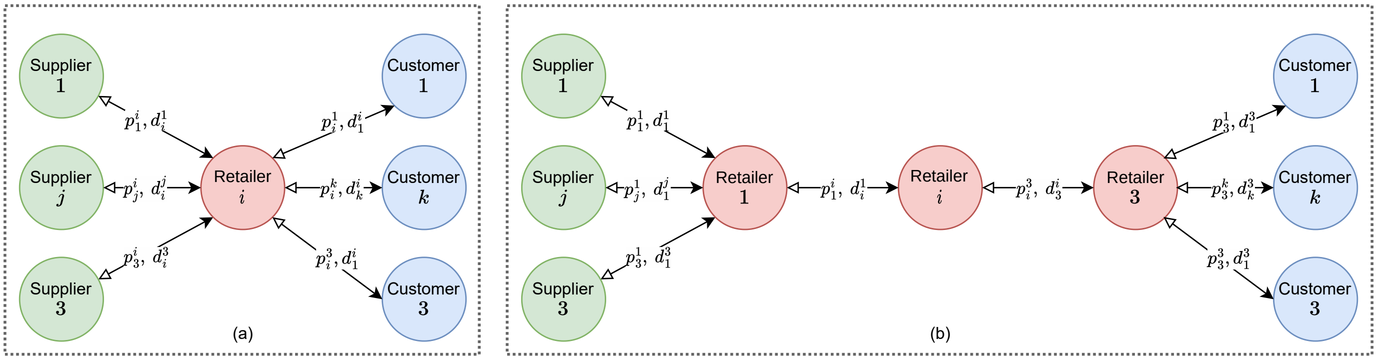

Fig. 3. (a) The basic topology of the supply chain with a single agent balancing with uncertainty from both upstream and downstream entities. (b) A multi-layer agents involved supply chain for uncertainty sharing.

can execute the action with highest Q-values based on the optimal Q-values function, Q^*, to achieve highest accumulated rewards. The optimal Q-values function can be optimized by minimizing the temporal difference error,

$$\delta = r + \gamma \max_{a'} Q(s', a') - Q(s, a) \tag{2}$$

where γ is the discount factor determining the importance of future rewards. Therefore, the we can update the Q-function by,

$$Q(s, a) \leftarrow Q(s, a) + \alpha\delta \tag{3}$$

In the setting of DQN, the Q-value function a learnable deep neural network parameterized by θ instead of the tabular encoding used in standard Q-learning. We can optimize it by

$$L(\theta) = \mathbb{E}\left[\left(Q(s, a; \theta) - (r + \gamma \max_{a'} Q(s', a'; \theta^-))\right)^2\right] \tag{4}$$

where (s, a, r, s') is sampled from the memory buffer D, and θ^- denotes to the parameters of target network.

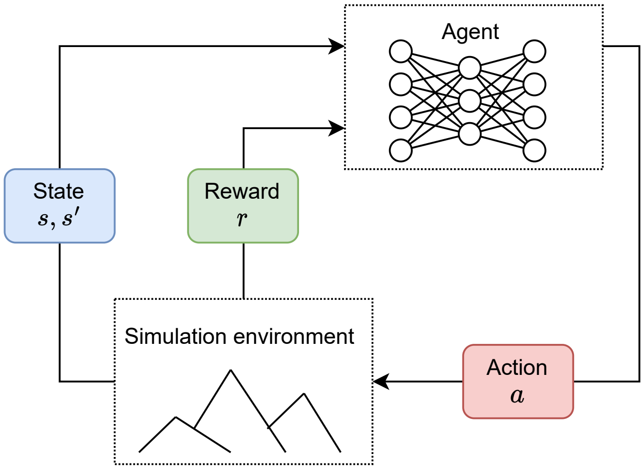

Fig. 4. The comparison of storage level and the number of purchased products.

C. RL Settings

1) State: The state s is defined as a vector embedding the information that can be gained by the agent from the environment. We denote the state of agent i as $s_i = [c_i, \mu_i, x_i, y_i] \in \mathbb{R}_+^{|\mathcal{S}_i|+|\mathcal{R}_i|+l_i+1}$. $c_i \in \mathbb{R}_+^{|\mathcal{S}_i|}$ is the product price from i's supplier; $\mu_i \in \mathbb{R}_+^{|\mathcal{R}_i|}$ is the anticipated demand from i's retailer; x_i is the current inventory level; and $y_i \in \mathbb{R}_+^{l_i}$ represents the product transition line, and $[y_i(t)]_n$ is the replenishment that arrives at time $t + n$.

2) Action: In each time point t, an agent can decide the quantity of purchasing, m_ij, and the selling price n_ik, forming the action vector $a_i = [m_ij, n_ik] \in \mathbb{R}_+^{2N}$. So, m_{ij} is the products amount ordered by agent i from its supplier j.

3) Rewards: We define the rewards in terms of total revenue, order cost, holding cost, backlog cost and equivalent carbon emissions,

$$\begin{aligned} r_i(t) = & \underbrace{\sum_j n_{ij}(t)d_{ij}(t)}_{\text{total revenue}} - \underbrace{\sum_k c_{ik}(t)d_{ki}(t)}_{\text{order cost}} - \\ & \underbrace{h_i\left(x_i(t)\sum_j d_{ij}(t)\right)_+}_{\text{holding cost}} - \underbrace{w_i\left(\sum_j d_{ij}(t) - x_i(t)\right)_+}_{\text{backlog cost}} - \\ & - \underbrace{\sum_k E_{ik}d_{ki}(t)}_{\text{carbon emission}} \end{aligned} \tag{5}$$

Where E_{ik} is the carbon emission for the product. The agent earn the profit by selling products that purchased from suppliers (order cost) to retailers (total revenue) and it aims to reduce the total equivalent carbon emission in the whole process.

VI. Experiments

A. Implementation Details

In the experiment, as shown in Fig. 3, the supply chain system includes three suppliers and three customers. An agent can purchase products from the suppliers at quoted prices and sell them to distributors at self-determined prices. This model effectively simulates the dynamic and uncertain nature of customer demands. We implement RL (DQN) in the supply chain with one single RL-agent or three RL-agents in Fig. 3 (a) and (b), respectively. In the RL method, the states include the product price from each supplier, the selling price to each customer, and the current inventory amount in each agent. The actions include setting the buying and selling prices of the product and the amount of product purchased by the agent for the next time period. The reward is calculated as shown in Equation 5. Furthermore, we considered the instability of suppliers and transitions, aiming to mimic scenarios where abnormal events occur, which ultimately affect the transaction amount. Therefore, we add a random discount factor to d_{ij},

$$d'_{ij} = d_{ij} \cdot p_{ij}$$

where p_{ij} is a random variable evenly ranging from 0 to 1. We assume customer demand is price-sensitive, such that $Q(n_k) = 10 - 2n_k + 0.05\epsilon$, where $\epsilon \sim \mathcal{N}(0, 1)$ represents Gaussian noise. The RL (DQN) configuration includes two layers with 128 units each for both the value and the advantage streams. We use the Adam optimizer and set the learning rate to 0.001. The discount factor, gamma, was set to 0.99. An epsilon-greedy strategy was employed for action selection, with an initial epsilon of 1.0, which decayed exponentially to a final epsilon of 0.01 over 50,000 steps. Experience Replay was employed to stabilize the learning process. The buffer size for the experience replay was set to 10,000. A batch size of 32 was used to sample experiences from the replay buffer for updating the Q-network. The target network was updated with the weights of the policy network after each episode.

To compare with the RL method, we designed a naive threshold-based heuristic strategy that determines the decision according to a certain threshold. We maintain a safety storage range, composed of sto^+ and sto^- (5000 and 1000 in the experiment, respectively). If the current storage level is lower than sto^-, the agent will purchase the differential product from suppliers, with the order amount being the same. If the current storage level exceeds the range, the agent will stop purchasing and attempt to satisfy all the customers. In other cases, the agent will evenly purchase products from suppliers and distribute them evenly among customers.

B. Result

A single agent implemented with DQN is able to handle uncertainties. Fig. 5 shows the learned purchasing strategy of the agent in the basic supply chain shown in Fig. 3 (a). We can see that the agent purchases products quickly at the beginning and fills the store to a comfortable level around 100 to avoid future shortages. After that, it focuses on selling products and does not make purchases for a long period. When the storage level reaches a median value (~ 50), the agent frequently trades products with suppliers to dynamically satisfy customer demands. The sawtooth fluctuations are caused by the connection disruption. In Fig. 6, we present a comparison of the number of products sold and the total demands from downstream customers. It indicates that the learned agent can satisfy customers well in such an uncertain environment. Most of the time, the storage level is equal to or greater than the demands, ensuring that sufficient products can be sold.

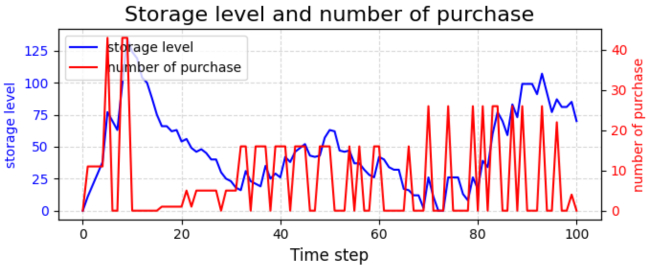

Fig. 5. The comparison of storage level and the number of purchased products.

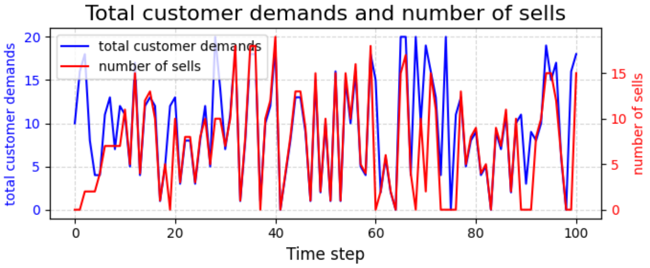

Fig. 6. The comparison of total customer demands and the number of sold products.

Comparison between an RL-agent and an agent driven by a heuristic strategy, we conclude the average profits in 200 simulations (Table I). For single-agent method, the heuristic strategy yielded an average profit of \$183.05 with a standard deviation of \$12.86, indicating a relatively stable performance. In contrast, the single agent employing RL outperformed the heuristic approach with a significantly higher average profit of \$267.87, albeit with a larger standard deviation of \$63.32, suggesting higher variability in the outcomes.

The multi-agent method followed a similar pattern, with the heuristic strategy achieving an average profit of \$215.43 and a standard deviation of \$23.68. The multi-agent strategy utilizing RL demonstrated superior performance with the highest average profit of \$307.19 among all strategies tested, but also exhibited the highest standard deviation of \$79.31, indicating the greatest variability in profit outcomes.

These results underscore the enhanced performance potential of RL strategies over heuristic in both single and multiple agent settings, as evidenced by the higher average profits. However, the increased standard deviations associated with RL strategies also highlight the greater risk in profits, which may be attributed to the dynamic and possibly complex decision-making processes intrinsic to RL algorithms.

VII. Future work

This research lays a foundational framework for integrating sustainability considerations with reinforcement learning to

TABLE I
AVERAGE PROFITS OBTAINED BY AGENT WITH DIFFERENT STRATEGIES.

MethodS	Average profits (USD)
single agent (heuristic)	**$183.05**±12.86
single agent (RL)	**$267.87**±63.32
multiple agents (heuristic)	**$215.43**±23.68
multiple agents (RL)	**$307.19**±79.31

enhance supply chain resilience and sustainability. However, the dynamic and multifaceted nature of global supply chains presents numerous avenues for further investigation. Future work will focus on several key areas to extend the contributions of this study:

Enhanced Model Complexity: Expanding the complexity of the MOGI simulation system to incorporate more granular sustainability metrics, such as water usage, land use, and biodiversity impact. This would allow for a more comprehensive assessment of environmental stewardship across the supply chain.

Advanced Reinforcement Learning Algorithms: Investigating the application of more advanced reinforcement learning algorithms, including deep reinforcement learning and multi-agent reinforcement learning strategies, to better capture the complexities and dynamics of global supply chains.

Supply Chain Collaboration Mechanisms: Developing mechanisms for enhanced collaboration and information sharing among supply chain participants. This includes exploring the role of blockchain and other decentralized technologies in fostering transparency and trust in sustainable supply chain practices.

Policy and Regulatory Impact Analysis: Analyzing the impact of policies and regulations on supply chain sustainability and resilience. Future research could model the effects of different regulatory frameworks on supply chain decisions and outcomes, providing insights for policymakers.

LLM-Enabled Agent-Based Simulation: Building upon the integration of advanced AI techniques, future research will explore the application of Large Language Models (LLMs) within the agent-based simulation framework to facilitate more sophisticated communication and decision-making processes among agents. LLMs can be utilized to enable agents to process and interpret natural language data, allowing them to extract actionable insights from unstructured data sources such as news articles, social media feeds, and industry reports. This capability will significantly enhance the agents' ability to anticipate and react to real-world supply chain disruptions and trends by understanding the context and sentiments expressed in global news and market analyses.

VIII. CONCLUSION

This paper studied the complexities and challenges inherent in today's global supply chains, underscoring the need for innovative approaches to manage uncertainties and enhance sustainability. By introducing the MOGI sustainable supply chain simulation system and employing a multi-agent reinforcement learning strategy, we have a significant step forward in addressing these challenges. Our findings reveal that reinforcement learning, when applied across a multi-level supply chain topology, not only improves risk management and profit margins but also significantly advances environmental, social, and economic sustainability objectives. The comparative analysis with heuristic strategies further emphasizes the superiority of reinforcement learning in navigating the uncertainties that plague global supply chains. This research contributes to the broader discourse on sustainable supply chain management, showcasing the potential of advanced simulation techniques to fortify supply chain resilience and sustainability amidst a volatile global landscape.

REFERENCES

[1] Anylogic. https://www.anylogic.com/. 2023.

[2] Cop 27. https://unfccc.int/event/cop-27. 2022.

[3] Cop 28. https://unfccc.int/cop28. 2023.

[4] Matsim. https://www.transitwiki.org/TransitWiki/index.php/MATSim. 2023.

[5] Simio simulation software. https://www.simio.com/applications/supply-chain-simulation-software/. 2023.

[6] T. Abu Zwaida, C. Pham, and Y. Beauregard. Optimization of inventory management to prevent drug shortages in the hospital supply chain. *Applied Sciences*, vol. 11:pp. 2726, 2021.

[7] R. Carbonneau, K. Laframboise, and R. Vahidov. Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, vol. 184:pp. 1140–1154, 2008.

[8] V. Kumar Ganesan, D. Sundararaj, and A. Padmanaba Srinivas. Adaptive inventory replenishment for dynamic supply chains with uncertain market demand. *Industry 4.0 and Advanced Manufacturing: Proceedings of I-4AM*, 2021.

[9] A. Taskin Gumus, A. Fuat Guneri, and F. Ulengin. A new methodology for multi-echelon inventory management in stochastic and neuro-fuzzy environments. *International Journal of Production Economics*, vol. 128:pp. 248–260, 2010.

[10] H. Li, J. Sun, J. Wu, and X. Wu. Supply chain trust diagnosis (sctd) using inductive case-based reasoning ensemble (icbre): The case of general competence trust diagnosis. *Applied Soft Computing*, vol. 12:pp. 2312–2321, 2012.

[11] J. Mori, Y. Kajikawa, H. Kashima, and I. Sakata. Machine learning approach for finding business partners and building reciprocal relationships. *Expert Systems with Applications*, vol. 39:pp. 10402–10407, 2012.

[12] A. Ning, H. CW Lau, Y. Zhao, and T. Wong. Fulfillment of retailer demand by using the mdl-optimal neural network prediction and decision policy. *IEEE Transactions on Industrial Informatics*, vol. 5:pp. 495–506, 2009.

[13] K. Sedamaki and A. Kattepur. Supply chain delay mitigation via supplier risk index assessment and reinforcement learning. In *2022 IEEE 1st International Conference on Data, Decision and Systems (ICDDS)*. IEEE, 2022.

[14] S. Thomassey. Sales forecasts in clothing industry: The key success factor of the supply chain management. *International Journal of Production Economics*, vol. 128:pp. 470–483, 2010.

[15] X. Wan, J. F Pekny, and G. V Reklaitis. Simulation-based optimization with surrogate models—application to supply chain management. *Computers & chemical engineering*, vol. 29:pp. 1317–1328, 2005.

Courtesy of IARIA Board and IARIA Press. Original source: ThinkMind Digital Library https://www.thinkmind.org