www.ijecs.in
International Journal Of Engineering And Computer Science
Volume 10 Issue 12 December 2021, Page No. 25586-25605
ISSN: 2319-7242 DOI: 10.18535/ijecs/v10i12.4666

Just-in-Time Inventory Management Using Reinforcement Learning in Automotive Supply Chains

Dwaraka Nath Kummari
Software Engineer, ORCID ID: 0009-0000-4113-2569

Abstract
Automotive supply chains are currently undergoing change management processes to meet the challenges posed by electric vehicles, connectivity, and digitalization. Cyber-physical intelligent systems are no longer just a vision but increasingly reality. To do so, companies want to shift from Just-in-Case Inventory Management to Just-in-Time Supply Chain Management. However, this is only possible when the technologies behind the Internet of Things are fully implemented in real-world supply chains. The consequence is that the demand signals are available in real time, and supply delays no longer occur. In practice, however, there is a so-called spaghetti supply chain as a result of the geographically networked and organizationally decoupled setup of automotive supply chains. In addition, not all suppliers are sufficiently digitized to be fully integrated into intelligent supply chain networks. As a result, localized Just-in-Care Inventory Management is still necessary in practice. Failure and supply delays can still occur but over time. Suppliers still face the challenge of matching a velocity-oriented Just-in-Time Inventory Management System that requests components "Just-in-Time" with a replenishment capability that is located "Just-in-Case".
The goal of this paper is to design a Just-in-Time Inventory Management System for localized spawning suppliers using Reinforcement Learning that deliberately allows interruptions of Just-in-Time Supply Chains by creating incentives to do so. We show how key performance indicators can be designed to learn a localized Just-in-Time Inventory Management System that causes targeted interruptions from the components of the automotive supply chain. These interruptions can be used as control levers for delay management by digital supply chain control towers. The control towers use the data available through reliable suppliers to localize the occurrence of speaking components that are still selected for Just-in-Care Inventory Management. These delays not only lead to direct additional costs but also to the threat of being disqualified as a supplier.

Keywords: Automotive Supply Chains, Change Management, Electric Vehicles, Connectivity, Digitalization, Cyber-Physical Systems, Just-in-Time Management, Just-in-Case Inventory, Internet of Things, Real-Time Demand Signals, Spaghetti Supply Chain, Supplier Digitization, Intelligent Supply Networks, Localized Inventory, Supply Delays, Reinforcement Learning, Inventory Interruptions, Control Towers, Delay Management, Supplier Disqualification

1. Introduction

Innovations in warehousing and logistics activities have a strong impact on supply chain performance but have mainly been limited to lowering costs and enhancing performance through visible global

supply chain management decisions. However, these are not the only supply chain performance criteria, and outsourcing production to lower-cost regions has contributed to more complex logistics systems that are neither cost-effective nor quickly responsive to changes in demand or unexpected supply disturbances. This paper focuses on the supply chain challenges of automotive assemblers who are responsible for satisfying an unpredictable final demand for their products by acquiring components from suppliers with long lead times and with uncertain production schedules, who in turn must organize themselves into multi-tier supply networks that can respond to requirement changes. These supply requirements can be planned a fixed time in advance but frequently vary unpredictably and so cannot be planned in their entirety and have to be met through just-in-time inventory management. The objective of this work is to model the information flows needed to manage the flow of required components through an automotive supply network, ahead of time for steady-state conditions and, as necessary, in real time for unsteady-state conditions. The paper combines the ideas of computer-controlled order processing conditions generally agreed upon by supply chain partners and computer intelligence applied to processing data regarding their supply and demand point inventories, lead times, and momentarily current order backlog inventory set out in manufacturers' electronic catalogs.

By applying a dynamic programming reinforcement learning approach deriving from artificial intelligence machine learning to the integrated inventory model, it attempts to extend simple two-level models to cover nested tiered structures composed of multiple tier levels. In summary, this raised the following research questions: Can multi-tier VMI evolve into a form of automotive supply chain VMI where supplier inventories are utilized, at JIT rates, to support the automaker in meeting final demand fluctuations? If so, what sort of replenishment policy would be appropriate? How could such policies be developed and utilized by knowledgeable supply chain partners?

1.1. Overview of the Study's Objectives and Relevance

The study proposes a methodological contribution in designing a just-in-time (JIT)-inspired inventory management policy, based on reinforcement learning (RL), which may help practitioners as an additional tool for decision support in complex systems such as a supply chain (SC). Furthermore, a computational experiment is performed to offer answers to academic researchers and professional managers who may want to use RL as a potential tool for SC strategy modeling and evaluation. The method is implemented in a numerical simulation set in an automotive SC multi-company framework, including fed suppliers, a manufacturing central plant, and assembly factories, in which small policies are decentralized per SC company. The results suggest the ability of RL to discover novel decentralized opportunistic inventory control solutions for automotive SC and consider factors that traditional approaches do not while sometimes affecting even positively global SC performance.

We aim to present this study by contributing to augmenting the current sparse literature on RL applied to SC research by investigating a potential line of research on RL as a competitive tool for SC strategy design. Our hope is also that this research may generate additional interest for RL applied to SC since the potential to solve complex SC models realistically is even increasingly reachable today due to the current availability of data. The original contributions of this paper to the field are two-fold: it proposes a JIT-inspired reinforced learning (RL)--based decentralized policy for the automotive supply chain (SC) and provides further insights into the use of RL to model SC strategy design and evaluation. Its empirical setting is based on real automotive SC dynamics and provides novel decentralized solutions based on RL.

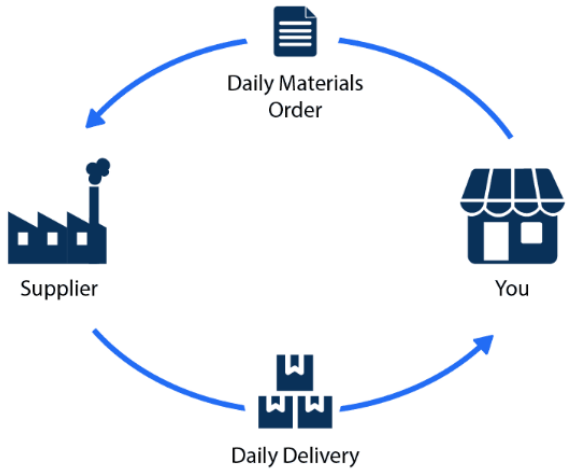

Fig 1 : Just-in-Time Inventory

2. Literature Review

1. Just-in-Time Inventory Management

Just-in-Time (JIT) inventory management is popular for its responsiveness to customers' needs and cost reductions associated with virtually eliminating inventory carrying costs. In JIT, adequate inventories are delivered to assembly plants "just in time" for their use in vehicle assembly. Although conceptualized as a means to reduce waste, JIT has its roots in the classic statistical methods of determining safety stock levels that are optimal for the minimization of costs associated with ordering costs, inventory carrying costs, and stock outs. JIT disruptions are the consequence of supply chain uncertainty resulting from the interaction between demand uncertainty and supply uncertainty. Demand uncertainty is reflected in variable customer purchase intentions that are correlated over time. Supply uncertainty is reflected in variable supplier lead times that are correlated over time, which is a common occurrence in supply chains in which primary suppliers must balance production capacity between the demands of their customers and the needs of their industry partners. In such orderly JIT systems, the primary suppliers isolate the true demand signal and provide "milestone shipments" of base vehicle kits at fixed time intervals by assembly plant schedules, thus allowing the JIT system to function properly.

2. Reinforcement Learning Fundamentals

Model-free RL has with great success proven to solve complex sequential decision problems as underlying Markov Decision Processes. To this end, buffer level-dependent partial state aggregation is often used to account for massive state space size. The key idea of reinforcement learning is that the agent does not observe rewards on all transitions. The agent therefore has to explore, trying out different actions and randomizing among them, so that it learns from the positive reinforcement of rewards received. To address exploration challenges, RL algorithms typically build on the actor-critic framework, which improves stability due to the critic component being updated by mini-batch standard off-policy methods. Since, in any JIT Control Model, two types of actions can be taken, namely stabilizing or exacerbating, both discrete- and continuous-action spaces with high-dimensional action space can occur. In addition, such policies must be learned under time-varying parameters on uncertain time scales which often result in highly nonstationary systems.

2.1. Just-in-Time Inventory Management

Since the early studies of inventory and production systems, it has been recognized that the main role of inventory is to buffer against variate demand and supply. This role has been questioned by several researchers over the years, but particularly since JIT systems have started to become popular with manufacturing companies. The managers of these firms insist and are supported by several proponents who believe that the inventory needed to absorb variability is counterproductive and increases costs. They advocate maintaining only enough inventory to handle what is termed "constant demand". In these cases, usually, the demand is during production lead times. With constant demand, it is not necessary to hold inventory. However, demand typically fluctuates and manufacturing lead times are stochastic as well, introducing variability. The answer to the question, "Is it better to hold inventory or to have lead time slack?" has historically reflected the attitudes toward inventory.

The JIT philosophy argues that extra costs occur while holding high levels of inventory.

Given the major effort made towards eliminating inventories and associated costs, it is perhaps surprising that the management of production and inventory systems remains an important topic. Some firms remain skeptical about JIT systems but have not entirely abandoned the strategies of holding inventories. Others have implemented JIT systems, but have not been particularly successful in managing costs without inventories. On the other hand, production control systems typically assume that production runs need to be controlled, but do not explicitly address the question of how much inventory should be held.

Equation 1 : State-Action Value Function (Q-Learning):

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]$$

where:

- $Q(s,a)$ = Expected value of taking action a in state s
- α = Learning rate
- r = Reward received after taking action
- γ = Discount factor for future rewards
- s', a' = Next state and action

2.2. Reinforcement Learning Fundamentals

Much of reinforcement learning drew previous research in the fields of control theory, dynamic programming, and optimal control. The Markov Decision Process model is the core concept of reinforcement learning. Markov Decision Process is defined as a five-tuple. S is the finite set of states. A is the finite set of actions. P is the finite set of state-transition probabilities. R is the finite set of rewards. P_0 is the state distribution at time 0. A policy is any mapping from states to actions. When a policy is deterministic, it defines a run of the process: a sequence of random variables where in step t the action is chosen based on the previous state through the policy, the result is that the next state is and the reward is, and distributions define the random components of the process.

In reinforcement learning, the Agent object performs the interactions with the environment. Internally, the Agent maintains a policy and a value function. The two functions can also be passed back and forth between the interaction and learning modules. At the beginning of an episode, the initial state is typically generated from a distribution. It then interacts with the environment, at each time step, parsing the feedback to make updates and returning to the learning module. The learning module then makes changes to the current policy, value function, or both and returns the new policy or value function to the Agent. The interaction and learning iterations shut down when the computed policy or value function reaches the desired degree of accuracy. These two functions are learned based on data observed through this interaction. The learning problem is that of inferring the (optimal) policy and/or value function from the observed data. Once the agent has learned the policy or the value function, it can be used either to make decisions or take actions in the environment.

2.3. Automotive Supply Chain Dynamics

The automotive industry is traditionally characterized by complex manufacturing processes and high-volume-low-variety production. Corporations undergo extensive planning phases and emphasize standardization to ensure the timely delivery of vehicles. Automotive products are consequently assembled from a vast number of parts and components. The finished vehicles are then delivered to a dealer network, which sells the vehicles to consumers while providing after-sales support. The automotive supply chain, which denotes the flow of goods from suppliers to consumers, is one of the most sophisticated supply chains and is often subject to delayed order fulfillment and excess inventories.

The complexity of the automotive supply chain can be attributed to several factors. Production is based on forecasting sales that are subject to unforeseen changes owing to the economic situation and market trends. Due to a relatively high degree of

customization, production and sales rates are normally not in sync, which leads to fluctuations in demand across the supply chain. Goods are transported from suppliers to manufacturers and subsequently delivered from manufacturers to the dealer network and sold to consumers. These shipments take considerable time due to transport distance and the logistics process. Any delays throughout the chain thus contribute to uncertainty. Automotive supply chains are also characterized by a long product life cycle, during which product recalls, seasonal business and technological changes are quite common.

Such dynamics drive many automotive manufacturers to carry large inventories throughout the entire supply chain to accommodate demand uncertainties and fluctuations. Finished vehicle stocks, for instance, are maintained at the manufacturer level to buffer against delays in orders or shipments of vehicles. Order fulfillment is however linked to the agents involved in the supply chain through a sequence of interrelated financial transactions and efforts to increase service levels and minimize cycles during times of economic prosperity result in increased costs.

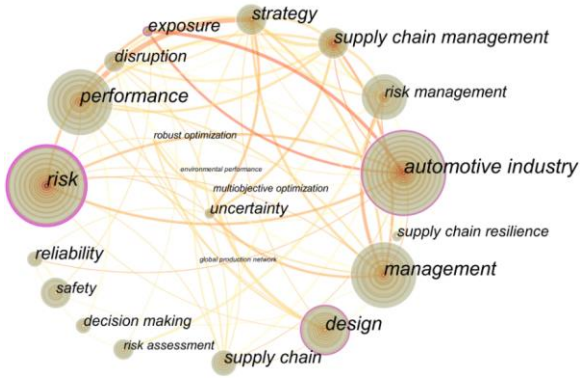

Fig 2 : Automotive Supply Chain Disruption Risk Management

3. Research Methodology

This paper applies an empirical case study, action research, and simulation-based development approach to develop a Just-in-Time (JIT) inventory management control technology based on reinforcement learning as a behavioral model to study how agents learn and interact with the environment. We analyze the development of a JIT inventory management control technology for a US automotive original equipment manufacturer and a group of its tier 1 suppliers. Action research is the commonly applied methodology in collaborative empirical management information systems research, where the design of information technology artifacts is an important part. Empirical case studies offer the possibility to study a real-life Information Technology innovation in context. Simulation-based artifact design is often used where actual field testing is not appropriate.

The methodological triangulation we apply has the advantage of using action research to collect data within a case study, which increases the depth and richness of our dataset. Further, it enables the researchers to collect data at multiple levels within and across organizations, and the artifact design offers the opportunity to collect data in a structured manner at multiple. The motivation for this research study is that academics and managers urgently want to find answers concerning the feasibility of deep learning or reinforcement learning-based JIT technology in practice. Despite an increasing number of experimental reinforcement learning-based inventory studies, many practical JIT conceptualizations are still mostly word-of-mouth, rarely conceived as algorithmic decision-making models requiring thorough testing in realistic empirical settings.

3.1. Research Design

We adopt a case-study approach to explore in-depth the relevance and applicability of JIT inventory management using Reinforcement Learning agents within an automotive supply chain. Supply chains often operate in complex environments, facing exogenous shocks, uncertainty, lack of transparency, conflicts of interest, co-variability, and systematic unreliability about demand and supply distributions and members' lead times, as well as supplier switching. The automotive supply chain is characterized by uncertainty and variability in demand, high pressure to reduce costs, and long

execution lead times. In addition, the supply chain is composed of several members who are interlinked and interdependent in terms of their production activities. These characteristics contribute to a potential mismatch between member activities and resources, with a tendency towards excess inventory at some points in the supply chain, while other members may have shortages causing delays to manufacturing. To accomplish these goals, close collaboration and exchange of information throughout the supply chain are needed. A JIT approach has gained acceptance within the automotive industry, which promotes mutually beneficial partnerships between auto manufacturers and high-performance suppliers. Achieving outstanding results by establishing continuous improvement customizable systems in close collaboration and partnership with their suppliers, getting them synchronized around Takt time reductions. By exporting this model, a competitive advantage has been gained. The close interlinking of production schedules, delivery schedules, and manufacturing lead times means that, without this management and support activity, partners become adversaries. Partnership relations may also not be available for the research, which is why an optimized way of integrating columns into a JIT environment and controlling the JIT environment to the benefit of all partners on the JIT chain is needed to resolve any potential conflicts, as the collaboration works both ways, requiring commitment from both sides.

3.2. Data Collection Methods

A qualitative approach and an exploratory method are deemed suitable for the research objectives since the literature has not been completed previously on the topic. This multiple-case exploratory study investigates the automotive supply chain context to advance the understanding of just-in-time inventory management about current policies, issues, dynamics, and especially data availability – both historical and real-time. To achieve this goal, qualitative semi-structured interviews are conducted with five leading industry practitioners of the prospective automotive supply chain case company. To learn more about the industry in general, three further interviews with senior industry executives are performed, increasing the representativeness of the research. For analysis and pattern-matching purposes, critical incidents and process-tracing techniques are established, as well as specific interview templates for JIT, data, and potential solution design.

The knowledge bases of the interviewees can be determined by years of industry experience. Five interviews are carried out with a total duration of 350 minutes with identifiable experts in their subject area. Potential confounding biases are controlled using cross-validating techniques and willingness-to-pay real JIT operations discussions. For further refinement of the study, broad external wrapping knowledge is gained from two additional senior executives who are not directly participating in the case company's day-to-day operations but are mostly working and consulting within the automotive supply chain. The industry relativizes the influence of further parties being interviewed from incongruent backgrounds, as they are carrying on discussions and structure capabilities around practical application examples with different OEMs, which are also continuously in the refinement process themselves.

3.3. Analytical Framework

This study proposes an analytical framework for just-in-time inventory implementation in automotive supply chains using reinforcement learning methods. Our framework is predicated on data-driven reinforcement learning methods such as Q-learning or related functions instead of common statistical modeling methods that are dominated by linear regression methods. Reinforcement learning methods are expected to outperform statistical modeling methods since a speed-up of data collection is possible in a typical automotive supply chain. Additionally, statistical modeling methods such as linear regression cannot capture the

complexities unique to automotive supply chains, thus reinforcement learning methods are more appropriate. The adoption of mobile cloud computing and the availability of new and diverse sources of data, such as social media and GPS tracking have also favored the adoption of data-driven, scalable machine learning methods. New sensor techniques, such as RFID chips, have enabled the continuous collection of large amounts of a variety of data.

In our framework, inventory levels are managed by using available inventory at adjacent periods as well as relevant data from past observations, in particular demand and predictive factor datasets. Reinforcement learning is often referred to as the generalization of Q-learning, where the action values of the Q-function are approximated. Instead of mapping all possible states directly to action values, a class of functions is used to represent the state-to-action mapping. The reinforcement learning method effectively learns the action value function that maps states to action values.

4. Reinforcement Learning Techniques

Reinforcement learning is one of the most exciting paradigms for generating intelligent behavior in complex dynamic stochastic environments. In RL, an agent interacts with its environment, observing states and receiving rewards, to develop a policy—a probability distribution over actions conditioned on each state—that maximizes the expected discounted sum of rewards. To develop this policy, the agent usually has to explore the environment, take action, and observe the ensuing consequences. Complementary to this exploration, the agent also engages in exploitation by using its experience to improve its performance.

The learning algorithm consists of a value function that quantifies the performance of the policy, and a control strategy—a method for discovering a better policy based on experience with the value function. The continuous advance of machine learning techniques has reshaped the RL landscape, enabling more and more demanding tasks to be implemented successfully. This section will provide a brief overview of the most supportive algorithms in this landscape.

1. Q-Learning

The Q-learning is the most well-known Reinforcement Learning algorithm. It was conceived as a solution for the discrete case, where the state-action space is composed of only a few states and a few actions. Q-learning permits learning in a model-free manner the optimal action-value function for a given controlled stochastic process that is represented by a Markov Decision Process given under a well-defined notion of optimality called the Bellman Optimality Operator. This is given by an iterative equation relating the value of the current state to that of the neighboring states and actions. The Q-learning algorithm differs in that the actual iterative process does not necessarily converge to the solution dictated by this Bellman Operator.

4.1. Q-Learning

In future works, we support the use of deeper networks trained with more diverse training data, which could serve as a useful initialization step for specific problems. Approximate dynamic programming methods are typically used to address problems with large state spaces where the state-action value function can be approximated through a parametric function. One such popular algorithm is Q-Learning, in its off-policy version, which estimates an increasingly accurate so-called action-value function.

At each iteration, Q-Learning effectively learns to predict the long-term utility of each action available in that state. To derive Q-Learning's update criterion, we first need to introduce some additional notation. Q-learning can be understood as a kind of temporal difference algorithm. The algorithm learns a mapping from state-action pairs to expected returns that can be used to derive an optimal policy. Q-learning iteratively refines its Q-values based on each observed action until the Q-values converge. Initially, they might not reflect the true returns.

However, instances of the anticipated returns do appear in the observed data.

Over time, the average value of these sampled returns approaches their expected value, so that Q-Learning converges. The Q-Learning update rule attempts to shift the predicted Q-value towards the observed return on every timestep, and it does so at a rate determined by the learning rate. This rule is valid because, according to the law of total expectations, the expectation of the Q-value is equal to the average of the observed returns. More precisely, the algorithm acts greedily concerning the current Q-values. When the current policy is refined, the update shifts the Q-values of the action taken in the observed state closer to the realized return.

4.2. Deep Reinforcement Learning

Reinforcement Learning (RL) has become successful in different applications, such as robotics or game playing. These early algorithms were based on Deep Q-Learning (DQL), where experience replay and experience sampling was used to successfully train deep neural networks stably and reliably. Although DQL applied several optimizations to Q-Learning to make it feasible with deep networks, it also has a significant flaw: it is an off-policy method that learns from old data. Off-policy cuts the learning speed of the agent and enlarges the convergence time, especially when using DQL to train Deep Networks in complicated and poorly sampled environments. Moreover, note that the large majority of work utilizing DQL architectures are training agents to perform better in their respective environments than the current minimum performance required to successfully finish. However, for online optimization problems, such as the supply chain inventory management problem addressed in this thesis, we want the RL agent to provide correct answers to the problem while still exploring the environment.

An alternative solution is to use a model-free policy optimization method. These methods directly learn a parameterized policy by optimization methods. The most straightforward approach is to need the parameters to derive the best expected performance of the agent given the current policy defined by these parameters, and that such parameters would be the agent's optimal set of parameters. This method is called Policy Gradient (PG) and comes in several variations. Policy optimization methods directly derive the optimal action at each step, behaving differently than value function methods. At the end of the day, all methods ultimately serve to approximate the optimal policy considering both the exploration and exploitation dilemma.

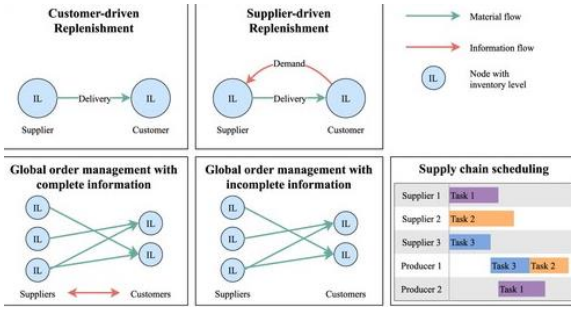

Fig 3 : Reinforcement Learning Algorithms

4.3. Policy Gradient Methods

Policy Gradient methods form a class of Reinforcement Learning (RL) algorithms that explicitly optimize for policies defined as distributions over actions conditioned on states. Unlike other classical RL algorithms that rely on a hand-crafted function approximator to determine actions, these methods rely on a general function approximator, which is updated directly to improve performance. They have several advantages compared to other classical algorithms. First, these methods directly address the challenging problem of selecting actions from the very large action spaces, which often arise in RL problems, particularly in robotics and other continuous environments. Second, they do not suffer from the instability known as the “High-Variance Problem” of Q-Learning, which often requires an expensive bootstrapping process to stabilize training. It is also unnecessary for Policy Gradient methods to give a high-quality policy. Third, these methods allow continuous action spaces; generally, for such

problems, it is much easier to parameterize the full action distribution than the Q function. In particular, the action distribution is typically tightly coupled to the cost function, which itself is easily parameterized, whereas the Q function and the dynamics of a system are not.

For this class of modifications, the policy is parameterized as a neural network and trained using the following stochastic policy gradient. To calculate an approximate gradient of the expected return directly from a sampled trajectory, PG algorithms typically use an auxiliary function: the advantage function, which quantifies how much better an action is relative to the expected cost of taking that action conditioned on the same state. Reinforcement Learning discussions often begin with a discussion of the state-action value function, which maps every state-action pair to the expected discounted return accrued by following that policy. In the control example, we take advantage of its local linearity to construct a recognition of the cost typically tightly coupled to the cost function, which itself is easily parameterized.

5. Model Development

In this section, we present how we develop the DQN model and its components to address the main research question. As the automotive supply chain is characterized by high volatility and uncertainty, we use a simulation-based approach to develop the core model of the DQN model. A simulation-based approach is based on the idea that there exists some form of optimization, defined by a mix of the set of parameters and functions that govern the simulation model. A simulation model is run repeatedly while changing, either in a random fashion, the decision variables that govern the simulation generator, or systematically, the parameters of the distribution that govern any random events in the simulation model. The simulation model generates some output measures, as temporary values. After some run length, the temporary values are averaged and used as the basis for the model update. The simulation is run to convergence, producing estimated output values for the parameter set being considered. The process is usually recursive and model run lengths are determined usually by considerations such as expected accuracy of estimated output. The recursive nature of the problem suggests that the simulation runs, essentially a basic step in a basic algorithm, could be done sequentially. Therefore, since we want the DQN model to have the capability of acting in a realistic period that is long enough to reproduce accurately the automotive supply chain dynamics, we implement a simulation-based approach to allow for wide-ranging exploration of the automotive supply chain dynamics during a realistic time frame.

1. Simulation Environment

The automotive industry is characterized by high demand volatility with lower and higher cycles, driven by abrupt local or global events. Therefore, it is plausible to use a simulation-based approach to model the just-in-time function of inventory management for the automotive supply chains because the dynamic aspect of the automotive supply chain incorporates high volatility and a more or less deterministic process. In addition, DQN and any reinforcement learning model, in general, are characterized by disjoint exploration which is useful in the context of unpredictable situations or sudden demand changes.

5.1. Simulation Environment

For this study, we develop a simulation agent utilizing a toolkit, where we utilize both the reinforcement learning agent, as well as the training environment simulations. With this setup, we implement our training environment within the software environment for our training and evaluation phase. Our model consists of four components, namely a supplier, a customer, and a reinforcement learning agent to control the order and receiving quantities of the customer and supplier respectively. The customer, supplier, and agent are interconnected through an interface. Thus, our setup fulfills the requirements set out for

research using demand and supply agent-based simulations.

In our agent-based simulation, the environment keeps track of time. The underlying market simulation runs sequentially in a time-stepped fashion. Each simulation step represents a discrete period. All decisions for the current period are made before the market is not cleared yet and all consequences are stored. The simulation then advances to the next period whereupon the same procedure applies until the end of the complete market simulation is reached. The objective of the customer or agent is to maximize their profit during the market simulation by reducing costs, or in other words, to minimize the cost during a specific period. As the simulation environment requires a clear reflective interface unlike a self-playing simulation, the action selected by the agent to control the inventory policy of its customer is not fed back in a transactionwise manner. It is stored until the end of the simulation and executes feedback.

5.2. State and Action Space Definition

An RL model learns by observing a system's statement and feedback. Various representations of the observations are commonly known as a state space. It plays an important role in the training of the model and might be time-discounted in some problems. An RL agent possibly acts in several different ways at any point in time. Such actions are also defined as an action space. The collection of state-action pairs as well as the expected reward define the underlying MDP. When appropriately designed, the MDP can help an agent tackle potentially complex decision-making problems through the agent's experience in the environment. The state and action space, along with the reward function, describes what is learned by the RL agent. These functionally should explicitly capture the differences in operational objective and availability, especially when working towards a system-wide optimized solution.

The state variable describes the complete information of the system at an instant when the system generates an incentive for an action. The relevant state variable should have a direct and indirect correlation with the incentive and must provide enough information about the expected dynamics of the state variable over time starting from that state. This enables the model to predict future outcomes better and generate optimal incentive strategies. The major concern while selecting the relevant state variables is to reduce redundancy in variation accrued to each state variable. This helps in effectively representing the stock variant of the reward function. The reward function can be modeled at a stage stochastic discrete nature. The action space defines how much the Letdown is optimistic about the product that is available for sale up to the point of an immediate sale. The space is discrete in the determination of the book for sale.

Equation 1 : Inventory Level Dynamics:

$$I_{t+1} = I_t + O_t - D_t$$

where:

- I_t = Inventory at time t
- O_t = Order quantity at time t
- D_t = Demand at time t

5.3. Reward Function Design

While several reward functions are conceivable to guide an agent toward its objective in reinforcement learning, our selection attempts to map performance metrics often used in automotive supply chains to reward functions that can be utilized within the framework described. As discussed, incentives typically affect the parameters of different units. However, given the nature of our problem and possible implementation issues when directly defining a reward function based on the parameters, we devise the reward function based on the concept of a unit-based incentive system. Thus, we aim to calculate a per-hour reward function that can also be mapped to monetary incentives following the

concept of a unit-based incentive system, mapping it to an agent-environment interaction representing a unit of time passing. A further motivation for our design choice stems from the consideration of incentive parameters, which have been shown to have a meaningful impact on performance. Thus, we take care to reflect as much of the incentive systems proposed in the literature.

State-of-the-art supply chain incentive systems typically base reward function design on cash flow or profit metrics. Consequently, we opt for a reward structure reflecting these parameters while also including inventory levels. This choice becomes necessary as we have shown that optimal cash flow depends heavily on the inventory levels at hand. By both the concept of a unit-based incentive system as well as further literature, we define the monetary reward function at each timestep by the following formula. Hereby, the calculation of cash flow is based on formulas. A special emphasis lies on incorporating other chain members in the definition of the reward function. This way, the inventory of the suppliers can be steadily controlled, and a coordinated replenishment decision during simulations is achieved.

6. Implementation in Automotive Supply Chains

Implementing our model in a practical supply chain environment is one of the main challenges to be addressed. Our research not only suggests a way of applying RL tooling in practice but also specifically for the automotive supply chain observed in the presented case study. In this section, we elaborate on the how-to of RL-based decentralized decision-making on JIT inventory control in selected local and global-specific automotive supply chain use cases. This includes how the realistic case study was identified, the relation of our approach to existing systems in place, and lessons learned concerning the operational challenges we faced.

Case Study Selection

The considered RL approach mirrors realistic supplier-buyer relationships in an operational scenario. The simulation weeks of the operational scenario were carefully selected accordingly. The selected automotive supply chain case study consists of three first-tier suppliers that produce sub-components of a complex product. The sub-components are assembled at an original equipment manufacturer location. The original equipment manufacturer is the only customer of the first-tier suppliers. The overall control of the supply chain is thereby in the hands of the original equipment manufacturer. All suppliers produce in an MTO manner, with varying levels of supply uncertainties during the production periods observed. The original equipment manufacturer has to rely on his suppliers' delivery quantities during the production period to have a streamlined production of the complex sub-component product. In case of insufficient delivery quantities, other MTO manufacturing operations are getting affected leading to financial penalties that are billed to the original equipment manufacturer by the end-customer. Translating this overall functioning of the automotive supply chain case study to the specific setup of the RL model requires careful parameterization.

The supply behavior divergence inherent to the selected use case aligns with the RL paradigms. The original equipment manufacturer has to rely on JIT deliveries from the suppliers while at the same time, the suppliers have to rely on the original equipment manufacturer purchase orders to configure their production plans. Since our RL model provides a scenario-inherent environment model, it is possible to specify the corresponding environment for each original equipment manufacturer-supplier-mto-order combination in the input files. Customer demands are defined in external files that can be edited without re-compiling the model system. The order combinations for the MTO operations of the original equipment manufacturer can be specified together with initial log file implementations to allow for performance assessment.

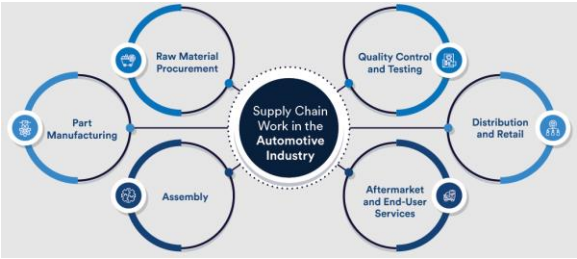

Fig 4 : Automotive Supply Chain

6.1. Case Study Selection

In this chapter, we present the implementation of the systems with data from the automotive parts industry. When designing these experiments, we want to provide insights into the main challenges the proposed methods would encounter when integrated into the daily routines of the companies. Therefore, we selected specific scenarios with levels of uncertainty and complexity that are representative of the daily supply chain decision-making of these companies, but where the material parameters affect the behavior.

The studied companies are Tier 1 and Tier 2 companies from the automotive industry of Mexico, specializing in the manufacture of metallic parts. Company data are deposits from clients and suppliers; finished product and raw material inventories; and sales and material requisitions bills from the last years. These data are the input for the supply chain simulation model used to evaluate the proposed dashboards. This simulation model is integrated with a Reinforcement Learning system, to endogenously obtain optimized predicting and replenishment decision dashboards. The model computes the optimal order policy based on the minimization of the total cost function of the whole supply chain, considering the uncertainties of demands and transports. All the internal and external cost parameters can be modified to simulate multiple scenarios.

The optimum replenishment deadlines are obtained for each integration level and different values of the cost parameters. These replenishment dashboards are defined as a four-dimensional matrix, defining the replenishment lead times considering the levels of integration and costs, and the time windows of deposits and demands. This four-dimensional matrix is the key tool of the proposed model, as without it we would not be able to eventually obtain optimized replenishment policies to evaluate the proposed dashboards.

6.2. Integration with Existing Systems

Tighter integration with existing systems enhances material flow visibility in complex inter-organizational environments such as automotive supply chains. Bulk data are necessary for an RL algorithm to derive a near-optimal replenishment strategy. Time delays in shipping due to transportation blockages or supplier management can cause high material inventories. Factories with insufficient demand information can order excessive material as a buffer against demand uncertainty. Establishing long-term relationships between partners can help rectify asymmetric relationships. Supply chain visibility can be advanced through increased information flows of forecasts, capacities, demand, and production plans. Suppliers can utilize inventory levels, material flow forecasts, product and demand volatility, and shipping performance history as additional criteria for scheduling.

Scheduling determines the timing of shipments and ultimately the extent of time delays. Such a complex managerial problem has attracted significant research effort and attention since the early 1980s when semiconductor manufacturing plants began to implement methods. So-called 'pull' scheduling policies allow only small orders to be placed and limit the range of variability in manufacturing. We expect an RL replenishment policy to also operate in smaller time horizons. Researchers have worked on policies in the hope of deriving heuristics for real-time applications. Just-in-time approaches to scheduling are based either on cycles or on fixed production sequences. Can be considered a sequence-control problem in which normally executed cycles are disrupted by emergencies or unpredicted demand when inventory consumption is sluggish.

6.3. Operational Challenges

To successfully operate a just-in-time inventory system while utilizing a reinforcement learning algorithm, unique challenges and considerations arise in this operational context. These challenges differ in various ways from the challenges associated with operating traditional forecast-driven inventory systems. The first challenge arises from the uncertain initial conditions associated with the model-free reinforcement learning agent. Without any training data to suggest what allocation quantities minimize observed inventory holding and stock-out costs, the chosen quantities need to be spread with a sufficient mixing ratio across the discrete action variable to allow the critic's action-value function to be properly explored, estimated, and optimized to find the optimal quantities. Thus, the chosen allocation quantities will imply a certain mixing ratio across the action variable; this ratio should be chosen based on the expected opportunity costs associated with the available allocation quantities. If the mixing ratio across the action values is constant, choosing more extreme allocation quantities, either vastly favoring one supplier over the others or drastically under-allocating to all suppliers, will allow the learning process to converge faster; however, due to the expected opportunity costs, this strategy will result in worse performance if the initial portfolio allocation is far away from optimal, necessitating that the extreme allocation quantity strategy only be used in the very early learning phase.

Inventory and stock-out costs are observable, but predictable structural or exchange rates or the probabilities underlying demand estimates for each supplier are unknown. For this reason, demand estimates used in the action-value function cannot be predicted from historical data predictions. Other types of machine learning methods can handle this prediction, and in general, are better suited for environments where accurate prediction of the feedback signals is easier to learn. Because of this need for availability predictions in the initial biasing of the action-value function, operating inventory systems that rely on other forecasting algorithms may not be suitable candidates for reinforcement learning modification.

7. Results and Discussion

To evaluate the performance of the different agents explored during the training processes, we used order service rate and order average shipment delay time. Order service rate is measured as the number of orders that are delivered on time over the total number of orders and the average order shipment delay is measured as the average delay of all orders. Given that the two performance metrics above have different scales, we used a normalization method, which we call z-score transformation, to transform the performance metrics into a common scale. To calculate the z-score transformation, we first subtracted the mean from the original metric and divided it by the standard deviation, which is the square root of the variance. In this section, we evaluated the normalized performance metrics, zservice−rate, and zaverage−delay to assess the performance of agents.

After the evaluation of the different agents, we will discuss the results for both metrics and then elaborately present the best policy. Following that, we also performed a comparison of the learned optimal policy against two traditional policies, including Just-in-Time and base-stock policies. Finally, a sensitivity test was conducted to provide better insights into using DRL in automotive SCs. In traditional automotive supply chain systems, quick response and high order fulfillment rates are often needed. Thus, if too many orders can't be delivered on time, it tends to incur more costs in the long run.

7.1. Performance Metrics

To evaluate the performance of our coordination algorithm, we need to establish effective metrics to quantify the improvements in specific performance indicators of the supply chain. We have chosen four performance metrics, which reflect key aspects of

supply chain performance. The first performance metric is the total order-to-delivery time, which indicates the average delay incurred by the end customers in receiving an order after the order has been placed. It reflects the responsiveness of the supply chain to customer demands. The second metric is the total average inventory level, which denotes the sum of the inventories of all supply chain members. It indicates the overall capability of the supply chain to protect against demand uncertainty. A higher inventory level results in a higher capital investment and operational cost, so there arises a trade-off relationship between order-to-delivery time and inventory level, and a lower inventory level is needed to reduce the cost. The third performance metric is the total ordering cost, which is incurred by all supply chain members due to placing replenishment orders to the upstream members to deliver items for customers. It reflects the cost incurred from demand uncertainty, which drives the ordering process. The final performance metric is the total operations cost, which is incurred by all members due to holding and ordering items. A higher total operation cost implies that the supply chain has a higher operating inefficiency.

These four performance metrics are interrelated to each other and are essential indicators of the collaborative order fulfillment process and the replenishment strategies within the automotive supply chain. Therefore, they are the most critical factors when designing and developing a multi-agent system to implement just-in-time inventory management in the automotive industry. By comparing these performance metrics when applying different algorithms, we can illustrate the advantages and disadvantages of each model. Subsequently, we will demonstrate that the proposed multi-agent algorithm has exceptionally improved the performance of the collaborative systems quite significantly, especially concerning the total cost incurred when coordinating the systems.

7.2. Comparison with Traditional Methods

One way to validate the proposed method is to compare it with traditional MSA methods. In this study, we chose the RMM3, a classical heuristic based on a simplified maternal model of a cell that uses RSOs with predetermined Part Baseline Schedules and MSA-RP, a heuristic based on a balancing algorithm to reduce the inadequacies of FJSP, as a benchmark. To make such a comparison possible and fair, we trained the RMM3 to generate the same number of deliveries over the JIT horizon as DQLS and performed the simulation for JIT days, using the same simulation algorithm as DQLS. As the automotive industry seasons, we used a week of each season for the MSA data preparation, with data partitioned per quarter.

As a second validation alternative, we used a scenario based on free-of-charge interfamily shipments introduced as a service differentiated by a sequencing approach. The theory behind the concept is that the manufacturer can achieve economies of scale by sequencing the model families and producing them in large quantities. This can be beneficial only for suppliers whose supplies are too small for economic shipping. Hence, there are two ways to cauterize the aftermath of this thinning process: charge fees; and accept heavier embargoes. These alternative strategies give the supplier the choice of how to address the black hole that shrinks its market. In this study, the similarity with the approach used as validation is that the DQLS-generated schedules also sequence for a week. In the present case, it is for a soft embargo, that allows for the somewhat irregular periodization of the interfamily production sequence. We used a two-model, two-supplier case to validate the methods.

The reason for testing against MSA is that both algorithms follow the same distributive approach: build material supply first, then synchronize with changing production requirements and production resource availability. Indeed, a possible first step is to assure adequate MSA performance when setting up CS-MSA for bulk supplies, blocking for a composed set of production targets that considers

initial scheduling and the irreversibility of the congestion process.

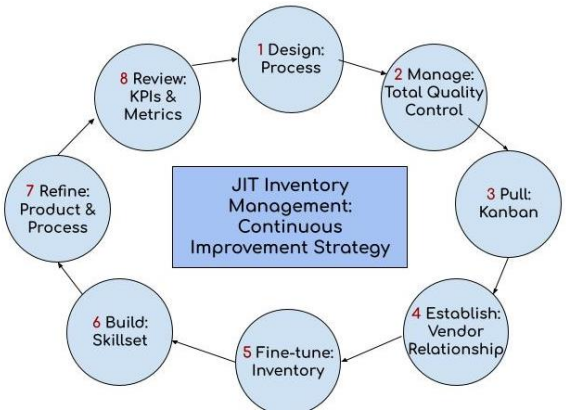

Fig 5 : Just-in-Time Inventory Management

7.3. Sensitivity Analysis

Sensitivity analysis can reveal the use of different parameter settings as well as issues relating to the decision-making for the initial condition of the learning algorithms. For the DDPG network, we vary the number of neurons in the hidden layers to analyze the model sensitivity. The learning rate and discount factors are also varied to study how varying their values affects the learning curve of the algorithm. The number of hidden neurons is one of the most important factors affecting RL convergence. A DDPG with one hidden layer and eight and sixteen neurons in the hidden layer was used. The case with sixteen neurons did not pause at the turning points for several episodes and therefore does not satisfy the condition of episode pausing for tuning; the case with eight did. Thus, the decision for the final architecture corresponds to 8 neurons in 1 hidden layer. The initial Q-value for a state-action pair is set at 0.5. When Q-values are used without regarding the scaling, negative and positive Q-values can be set equal to 0.5, but the presented results suggest that this practice is dangerous. The fact that the algorithm procedure can learn is related to how the DDPG handles the initial Q-values. The results shown below lead to that negative and positive Q-values should not be set equal to 0.5.

If this information is used on the RL software, the same Q-value learning problem may arise. The stability of the RL architecture also needs to be addressed further. The convergence speed of the reward function in a ratio is similar, but learning seems slower because of the difficulties in the learning procedure path. Because this network has symmetric Q-values in the DDPG framework, asymmetric Q-value architectures should be introduced in future work for faster convergence.

8. Implications for Practice

1. Strategic Recommendations

Based on the previous results, for those managers who operate a JIT policy and focus on the minimization of costs related to shortage penalties, for example, expensive lost sales or expensive downtime of the assembly line, our recommendation is to gradually raise order intervals and to adapt stock levels as demand variability rises. Moreover, a decision-maker should prefer to apply a JIT policy to locations with small stock or ordering costs and prefer the NIT policy to locations with large stock or ordering costs. Conversely, if shortage costs related to JIT policies are not significant, a manager could focus on demand uncertainty alone to set stock levels. Specifically, he or she should increase stock levels along the entire supply chain as uncertainty rises.

However, this can lead to a huge overstock situation. In this regard, an analysis of stock levels for different locations can improve coordination within the network and provide upper levels with a clearer view concerning the critical location steps. The methodology we present allows for examining the supply chain cost structure in terms of product characteristics, stock and ordering costs, ordered service level, installed capacity costs, demand uncertainty, and shortage penalties. Such an analysis is a valuable basis for refining the supply chain design process and deriving a general supply chain management guideline.

2. Policy Implications

As stated above, further coordination between different hierarchies of the supply chain is required to avoid too many JIT demands at the next location downstream under a JIT policy since any request for

a pickup from excellent demand predictability adds a non-negligible upper bound to the stock level. Some of the principles for demand control could be the following. In the decision on whether or not to outsource supply chain functions, a further test concerning the essential features of the various products, including suitability for each of the service strategies, is indispensable.

8.1. Strategic Recommendations

As the presented results have shown, using insights from reinforcement learning may enhance the regular policy of automotive companies. Such insights might be used to fine-tune safety stock policies and were proven to support long-term supply chain planning while lowering inventory holding costs when compared with a regular policy. These supportive use cases are not only positive developments for companies relying on safety stock leveling. The presented research brings additional insights into how companies might model their dependencies in a different way that reduces their reliance on safety stock policies. They might for example try to explore and exploit the dynamics of their dependencies with other partners to allow for quicker replenishment cycles and to look for joint ventures into special mission supply chains with few partners that allow for more sensitive demand models. Therefore, by decreasing supply chain complexity across all levels, automotive companies may enable the development of supply chains in which data sharing may lead to innovative demand-supply considerations and creative new business models.

However, for each of these recommendations come multiple challenges. Forming a unique three-tier-demand model for the whole supply chain is not easy as it relies on many assumptions and the modeling efforts are different for tier one, two, or three suppliers with different amounts of produced inventory. Setting up a special mission supply chain with few links may also not be easy and takes time, especially for a branch that is used to many partners at each level. Furthermore, building trust in suppliers' data processes so that levels of information sharing allow for enhancing models might take a long time, especially across different cultural branches. Therefore, it is imperative for the research as well as the teaching that we do to create a better understanding of the opportunities and challenges across the whole automotive chain to enable companies to further broaden their range of possible business models whilst supporting an educational understanding of these dynamics.

8.2. Policy Implications

This research provides implications for both industrial companies and support institutions. The model can help find the relationship between regulatory policy and JIT production. It can demonstrate the JIT path for local automotive suppliers. Local governments need to consider whether to provide financial support to help develop JIT for local automotive suppliers. Rules and restrictions on entering JIT mode should also be considered. If the size and value of local automotive suppliers can divide JIT manufacturers into two or three parts, the policies should be differentiated. Differentiated local government subsidies can meet the demand of local automotive suppliers. Restrictions on AR and the required inventory for entry into JIT mode can be set according to the different values of local suppliers. Support institutions should strengthen the investment intensity of local major automotive manufacturers to enter the research and development links. Suppliers from suitable countries could participate in these details. These support institutions can also promote the central automotive manufacturer to share the research and development network and the research and development product with the local major automotive manufacturer, which can apply for more investment in the short term. The government can help local suppliers share the investment rewards.

The decisions for the two parties fade away. The model can also display how the setup of prohibitive expenses affects obstacles to coordinating policy.

When policy parameters decrease, the penalty points will gradually become more attractive or reflect the spatial differences of local automotive suppliers. The incentive policies can vary according to the current location. If preferred suppliers can be divided into a few types in terms of size and value, then the corresponding incentive policies should be considered.

Equation 3 : Reward Function for Inventory Policy Optimization:

$$R_t = -h \cdot \max(I_t, 0) - p \cdot \max(-I_t, 0)$$

where:

- R_t = Reward at time t
- h = Holding cost per unit
- p = Penalty for stockout per unit

8.3. Future Research Directions

Towards Just-in-Time Inventory Management in Real Systems. As important as it is to test our approach on historical data from a real-world existing automotive supply chain, it is even more crucial to implement and test the system in a real supply chain where such resources would be constantly flowing, or influencing their most likely future destinations. Therefore, more research and collaboration among factories in supply chains is essential. This must necessarily expand on further Practical Use Case Scenarios, utilizing new AI techniques integrating the entire supply chain to develop policy models sensitive to the specific characteristics of each sector in the same supply chain.

Importance of a Replay Buffer. Random data does not make up an experience replay buffer, only predicting and executing decisions based on accurate models of those predicted decisions can create that buffer. Then using it to train the other main networks of our framework, we can efficiently bootstrap the decision policy through off-policy reinforcement learning. We, therefore, encourage researchers of the entire area to focus on how to create optimal Replay Buffers that work for their problems, maximizing their capacity for Transfer Learning.

Focus on Model Research. Cell-based manipulation designing and learning better model family priors is paramount to be able for Supply Chains and other applications to move to Multi-Agent Reconstruction by Reinforcement Learning like the ones that dominate these markets. We think that the introduction of similar but simpler models in the supply chains should be the main focus of related future research. Because we all know the adage. Do not optimize parameters and complex architecture of a model, when the job can be done with a simpler one less subjected to overfitting.

9. Conclusion

After gaining insights on just-in-time inventory management in automotive supply chains using reinforcement learning, we make several comments in this conclusion section. It appears that although original equipment manufacturers recently invested much more in in-house component production capabilities, they still desire to reduce their supplier base. However, this strength in local final assembly—vertical control—is not balanced by controlled domestic supplier capacity levels. As a result, supply delays can emerge. It should also be noted that supply disruptions adversely remove the advantages of flexible sourcing policies in automotive supply chains. An increasing demand volatility undermines the cost advantages of the multistage just-in-sequence concept. However, it is still unclear to which extent manufacturers are willing to absorb additional risks through lower sourcing costs paid to omnichannel suppliers.

Research opportunities exist about the integration of real-time data analytics and cloud-based information sharing via blockchain-like networks. Proactive and joint risk mitigation measures may lead to more favorable conditions than reactively increasing component prices at higher speed or volume levels. The point is that controlled realignment of assembly plant resource conflict strategy will lead to a corresponding increase in

supplier deliveries and a decrease in costs compared to working alone. Furthermore, joint component inventory management by automobile manufacturers and suppliers requires a certain culture. Suppliers will only join such an endeavor if they can trust OEMs to share their willingness to create long-run benefits.

It is important to point out that modeling predictive analytics is a compressive task itself. However, machine learning-based predictive analytics cannot resolve the complex synergy effect. Finally, the driving role of the OEM in predicting demand, influencing component requirements, and sharing available lower welding costs are of utmost importance in automotive supply chains. In this context, it is essential to balance the use of controllable key performance indicators—for both involved parties—against additional change management efforts. Finally, we would like to emphasize that the focus of our research is automotive supply chains but that principles learned can also be applied in related industries.

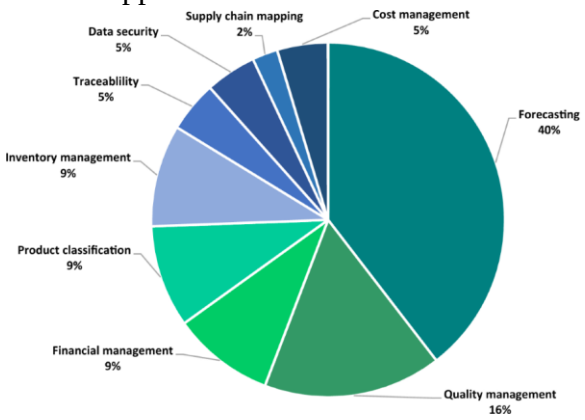

Fig 6 : Deep Learning Into Supply Chain Management

9.1. Final Thoughts and Future Outlook

The growth of the automotive industry has resulted in a high level of complexity in the automotive supply chain with the high cost of inventory. To tackle the high cost and risk of inventories, we proposed to trigger the shipment of the supplier's components in response to demands from the customer or subsequent stages of the supply chain. Additionally, as manufacturers produce and sell more products, they do not keep inventories of finished goods until they are asked by their customers. The development of techniques enabling a Just-in-Time inventory system for an automotive supply chain has been investigated extensively, however, the research has not been formulated to get a clear avenue toward practical implementation. This research aimed to plug the gap by exploring the application of Reinforcement Learning to solve an automotive supply chain problem for Just-in-Time inventory.

To summarize, to get a clearer understanding of using the Reinforcement Learning algorithm to optimize inventory management, we focused on the practical aspect of the system. We showed that using discrete actions with penalties for either stockout or excess inventory had the best effect on the model. We have also identified that using a two-layered neural network was the best architecture to use according to the Hyperparameter tuning experiments that we have done. Furthermore, our findings presented here contribute to the goal of creating a practical inventory management system utilizing deep reinforcement learning for not only automotive but also other supply chains. While we achieved our goals and produced a working model, further work is still required to construct a more generalized model that could be used in an actual working environment. Nevertheless, we hope that our findings will serve as a guideline for other researchers and serve as a base for further research. With that said, the use of Deep Learning, Natural Language Processing, and Reinforcement Learning to construct a more practical model could be considered in response to the current trends in deep learning and AI development.

11. References:

1. Chava, K., Chakilam, C., Suura, S. R., & Recharla, M. (2021). Advancing Healthcare Innovation in 2021: Integrating AI, Digital Health Technologies, and Precision Medicine for Improved Patient Outcomes. Global Journal of Medical Case Reports, 1(1), 29–41.

Retrieved from https://www.scipublications.com/journal/index.php/gjmcr/article/view/1294

2. Nuka, S. T., Annapareddy, V. N., Koppolu, H. K. R., & Kannan, S. (2021). Advancements in Smart Medical and Industrial Devices: Enhancing Efficiency and Connectivity with High-Speed Telecom Networks. Open Journal of Medical Sciences, 1(1), 55–72. Retrieved from https://www.scipublications.com/journal/index.php/ojms/article/view/1295
3. Avinash Pamisetty. (2021). A comparative study of cloud platforms for scalable infrastructure in food distribution supply chains. Journal of International Crisis and Risk Communication Research , 68–86. Retrieved from https://jicrcr.com/index.php/jicrcr/article/view/2980
4. Anil Lokesh Gadi. (2021). The Future of Automotive Mobility: Integrating Cloud-Based Connected Services for Sustainable and Autonomous Transportation. International Journal on Recent and Innovation Trends in Computing and Communication, 9(12), 179–187. Retrieved from https://ijritcc.org/index.php/ijritcc/article/view/11557
5. Balaji Adusupalli. (2021). Multi-Agent Advisory Networks: Redefining Insurance Consulting with Collaborative Agentic AI Systems. Journal of International Crisis and Risk Communication Research , 45–67. Retrieved from https://jicrcr.com/index.php/jicrcr/article/view/2969
6. [6] Singireddy, J., Dodda, A., Burugulla, J. K. R., Paleti, S., & Challa, K. (2021). Innovative Financial Technologies: Strengthening Compliance, Secure Transactions, and Intelligent Advisory Systems Through AI-Driven Automation and Scalable Data Architectures. Universal Journal of Finance and Economics, 1(1), 123–143. Retrieved from https://www.scipublications.com/journal/index.php/ujfe/article/view/1298
7. Adusupalli, B., Singireddy, S., Sriram, H. K., Kaulwar, P. K., & Malempati, M. (2021). Revolutionizing Risk Assessment and Financial Ecosystems with Smart Automation, Secure Digital Solutions, and Advanced Analytical Frameworks. Universal Journal of Finance and Economics, 1(1), 101–122. Retrieved from https://www.scipublications.com/journal/index.php/ujfe/article/view/1297
8. Gadi, A. L., Kannan, S., Nandan, B. P., Komaragiri, V. B., & Singireddy, S. (2021). Advanced Computational Technologies in Vehicle Production, Digital Connectivity, and Sustainable Transportation: Innovations in Intelligent Systems, Eco-Friendly Manufacturing, and Financial Optimization. Universal Journal of Finance and Economics, 1(1), 87–100. Retrieved from https://www.scipublications.com/journal/index.php/ujfe/article/view/1296
9. Cloud Native Architecture for Scalable Fintech Applications with Real Time Payments. (2021). International Journal of Engineering and Computer Science, 10(12), 25501-25515. https://doi.org/10.18535/ijecs.v10i12.4654
10. Pallav Kumar Kaulwar. (2021). From Code to Counsel: Deep Learning and Data Engineering Synergy for Intelligent Tax Strategy Generation. Journal of International Crisis and Risk Communication Research , 1–20. Retrieved from https://jicrcr.com/index.php/jicrcr/article/view/2967
11. Chinta, P. C. R., & Katnapally, N. (2021). Neural Network-Based Risk Assessment for Cybersecurity in Big Data-Oriented ERP Infrastructures. Neural Network-Based Risk

Assessment for Cybersecurity in Big Data-Oriented ERP Infrastructures.
12. Katnapally, N., Chinta, P. C. R., Routhu, K. K., Velaga, V., Bodepudi, V., & Karaka, L. M. (2021). Leveraging Big Data Analytics and Machine Learning Techniques for Sentiment Analysis of Amazon Product Reviews in Business Insights. American Journal of Computing and Engineering, 4(2), 35-51.
13. Routhu, K., Bodepudi, V., Jha, K. M., & Chinta, P. C. R. (2020). A Deep Learning Architectures for Enhancing Cyber Security Protocols in Big Data Integrated ERP Systems. Available at SSRN 5102662.
14. Chinta, P. C. R., & Karaka, L. M.(2020). AGENTIC AI AND REINFORCEMENT LEARNING: TOWARDS MORE AUTONOMOUS AND ADAPTIVE AI SYSTEMS.