Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

ENHANCEMENT OF VENDOR-MANAGED INVENTORY PLANNING THROUGH DEEP REINFORCEMENT LEARNING

Marco Ratusny[1], Jee Hyung Kim[1], Hajime Sekiya[2], Maximilian Schiffer[3], and Hans Ehm[1]

[1]Infineon Technologies AG, Neubiberg, BAVARIA, GERMANY
[2]Department of Mathematics and Computer Science, University of Hagen, Hagen, NORTH RHINE-WESTPHALIA, GERMANY
[3]School of Management, Technical University Munich, Munich, BAVARIA, GERMANY

ABSTRACT

We explore the application of Twin Delayed Deep Deterministic Policy Gradient (TD3), a Deep Reinforcement Learning (DRL) algorithm, for optimizing Vendor-Managed Inventory (VMI) systems in the semiconductor industry. We introduce a novel multi-scenario DRL algorithm with a continuous action space, designed to effectively manage diverse product/customer combinations, thereby improving VMI performance. We evaluate our algorithm's efficacy on three distinct products as well as 100 product/customer combinations for the multi-scenario approach. A sensitivity analysis examines the effects of varying shipment penalties on the percentage of no violations (PNV) and shipments. Our findings indicate that our DRL-based VMI model significantly surpasses existing policies used in the semiconductor industry by five percentage points.

1 INTRODUCTION

The global economy is currently witnessing a surge in demand for innovative supply chain models, driven by the rapid growth of information technology and the continued globalization of industries. Concurrently, supply chains are becoming more susceptible to interruptions due to factors originating from the demand side, supply side, and catastrophic occurrences (Monostori 2018). The semiconductor industry, in particular, is experiencing significant challenges due to its complex internal supply chains, which involve more than 1,000 manufacturing steps for Wafer Manufacturing (Lee et al. 2019; Hsu et al. 2020). This intricate production process and the need to manage long lead times further complicate the industry's ability to react swiftly to demand fluctuations. Therefore, predicting demand for semiconductors is difficult due to consumer markets' fluctuating and cyclic nature, leading to the bullwhip effect (BWE) and the need for maintaining safety stocks. However, holding excessive inventory poses financial risks, especially for the semiconductor industry, where short product life cycles can cause high scrap costs.

In this challenging backdrop, collaborative approaches among supply chain partners have gained utmost importance, establishing concepts such as Vendor-Managed Inventory (VMI), a supply chain collaboration strategy focused on information sharing and inventory management (Lotfi et al. 2022). It has recently gained significant attention in academic research and industry practice. In the general form of VMI, suppliers are responsible for monitoring and replenishing inventory based on customer demand forecasts rather than explicit orders. This approach allows suppliers to make informed decisions and maintain inventory levels within the mutually agreed minimum and maximum levels (Fry et al. 2001). VMI offers flexibility to both suppliers and customers, ensuring inventory availability and timely inventory consumption.

The semiconductor industry exhibits inherently long production lead times, such that VMI applied in this contest exhibits a unique structure as shown in Figure 1. Customers share current stock levels and, essentially, demand forecasts. This demand forecast information enables suppliers to plan and dispatch

replenishment to designated inventory locations. Customers can then draw from the inventory as required, maintaining a balance between contractually predefined minimum and maximum levels.

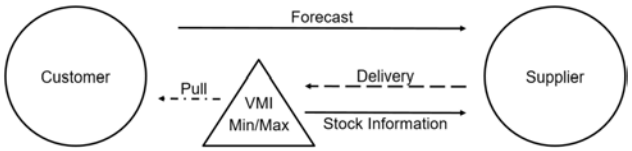

Figure 1: VMI configuration in the semiconductor industry (Afridi et al. 2020).

Nevertheless, VMI does not always perform as desired in the semiconductor industry due to the lack of common understanding between suppliers and customers: suppliers rely heavily on customer demand forecasts, which can be inaccurate, leading to suboptimal planning. Inaccurate forecasts could result in suppliers being unable to respond to sudden fluctuations in demand, exacerbated by the industry's long production lead times. Thus, more advanced methodologies capable of handling demand forecast fluctuations are necessary to enable more accurate and efficient inventory replenishment and address these challenges. Additionally, evaluating VMI performance requires specific metrics differing from traditional inventory management methods.

Related Work While various research studies have explored analytical, simulation, system dynamics, and metaheuristic techniques for VMI planning, practical implementation often relies on comprehensible and straightforward methods like heuristics and the newsvendor model (Sui et al. 2010). Recent methodologies for addressing VMI challenges involve the application of artificial intelligence (AI) techniques, framing VMI as a Markov decision process (MDP) (Mohamadi et al. 2024). Little research today utilizes deep reinforcement learning (DRL) algorithms to solve problems under VMI settings. Sui et al. (2010) aim to find an optimal replenishment policy that minimizes total costs in a two-echelon supply chain under a VMI consignment setting. Boute et al. (2022) details the essential design elements of DRL algorithms and their strategic implementation within the domain of inventory management. Oroojlooyjadid et al. (2022) propose a DRL algorithm to play the beer game and obtain near-optimal order quantities when teammates follow a base-stock policy. Mohamadi et al. (2024) uses DRL, specifically an advanced Actor-Critic algorithm, to solve a perishable inventory problem, outperforming the current implementation. Moreover, many studies focused on discrete action spaces, which might not sufficiently represent the intricate decision-making processes necessary in current supply chain management. (Kara and Dogan 2018; Sun et al. 2019) Afridi et al. (2020) and Ahmad et al. (2022) provide the most related work compared to ours and propose an approach based on DRL algorithms to determine a VMI replenishment plan. Their research shows that DRL algorithms are a potential approach for VMI since they lead to significant improvement in its performance as compared to the commonly used VMI planning methods. Nevertheless, their work is based on only one or three products, which does not fully validate the viability of DRL algorithms for VMI. Furthermore, both studies run into the issue of increased shipments, which are undesirable among customers. However, our approach addresses the issues of increased shipments and the ability to handle multiple product/customer combinations simultaneously.

Contribution We propose a novel multi-scenario DRL algorithm that can handle various product/customer combinations to enhance VMI performance. More specifically, our contribution is threefold. First, we create an extension of the work by Afridi et al. (2020) and Ahmad et al. (2022) by utilizing the Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithm, requiring a continuous action space compared to the Deep Q-Network (DQN) algorithms with a discrete action space. Second, we develop a multi-scenario DRL algorithm capable of handling multiple product/customer combinations. Third, we provide a numerical study showing that our multi-scenario approach outperforms the current solution of a large European semiconductor company and provides results comparable to an algorithm trained explicitly on one product/customer combination. Moreover, we provide a sensitivity analysis of the impact of the shipment penalties on the Key Performance Indicator (KPI) evaluation.

Our multi-scenario analysis yields significant improvements for central KPIs. In particular, we observe a significant improvement in the percentage no-violation (PNV) and the α-service level (SL) by more than five percentage points. In addition, our TD3 algorithm allows to maintain the total number of shipments at the level of the existing strategies, even with a small penalty rate. Notably, we saw a 17.7% increase in PNV under TD3, which is an undesired significant increase compared to current operational benchmarks.

Organization The remainder of the paper is structured as follows: Section 2 delineates the research methodology, illustrating how VMI is formulated as a MDP and detailing the application of the TD3 algorithm to enhance VMI replenishment strategies. It also introduces a multi-scenario approach. Section 3 outlines the experimental design. Section 4 presents a numerical study analyzing the performance of deployed DRL algorithms on select products and benchmarking against the current implementation of the semiconductor manufacturer. Section 5 summarizes the main conclusions from the research and outlines potential areas for future research.

2 METHODOLOGY

In the following, we will describe our methodology by first defining our MDP and its main components, detailing our multi-scenario approach, the resulting action space, and our DRL agent architecture.

2.1 VMI as a Markov Decision Process

A MDP is a mathematical framework that allows the modeling of multi-stage decision-making in a stochastic environment (Sutton and Barto 2018). It is especially suitable for modeling sequential decision-making scenarios like those in a VMI system because the next inventory condition (next state) is determined only by the current inventory condition (current state) and the chosen number of product shipments (current action). Given the present state and action, the previous inventory condition and action do not affect the current decision, i.e., it satisfies the Markov property. Fundamental to an MDP is its state space, which includes all possible conditions the VMI system can reach. Each state provides a complete system representation at a particular moment in the time horizon. Building on this setup, the MDP further consists of actions, which depict decisions made in a state, and a transition function, which defines the probability of moving from one state to another state given a specific action. A reward function quantifies the benefits of each action. In the following, we will describe our VMI planning problem as an MDP.

Decision Epochs The decision epochs correspond to the discrete time steps at which decisions are made during the planning horizon. In our context, decisions are taken daily, i.e., every single day represents a separate decision step. Each day, the supplier outlines the quantities for replenishment, and within a day, customers can withdraw items from the stock.

State Space The state space S is a set of all possible states where we denote a state on day t as $s_t \in S$. The state s_t contains a complete system representation on day t, which allows the agent to decide on the number of replenishments on day $d := t + OLT$, where OLT is the order lead time. Accordingly, the state includes the stock level on day t (CSP_t), the planned replenishment (FR_t) and the demand prediction (FC_t) for the next $(OLT-1)$ days, the minimum and maximum stock level on day d (zF_d, ZF_d), the maximum number of available shipment on day d (a^d_{max}), and the accumulated customer forecast behavior until day t (FcB_t).

We define three-dimensional pre-processed input variables for the agent based on the state space. This enables the agent to process meaningful information rather than raw data. Specifically, we define three variables: the deviation of the predicted inventory level from the expected mean level ($DTMF$), the customer forecast bias (FcB), and the naive action state (NAS). Each state is normalized in a range of $[-1,1]$, which ensures that the number of state-action pairs remains finite as this is crucial for efficient exploration and evaluation within the MDP.

The $DTMF_t$ is defined as the normalized difference between the predicted inventory level on the day d (FSP_d) and the expected mean level on the day d (MF_d). The forecasted stock position is estimated based

on the current stock position (CSP_t), planned replenishments (FR_x), and expected demand (FC_x) as shown in Equation (1). MF_d is the forecasted mean inventory level computed as the average of the forecasted minimum and maximum stock levels (zF_d and ZF_d), i.e., $MF_d = \frac{zF_d + ZF_d}{2}$. Finally, the $DTMF_t$ is calculated as their normalized difference, as shown in Equation (2).

$$FSP_d = CSP_t + \sum_{x=t+1}^{d} FR_x - \sum_{x=t+1}^{d} FC_x \tag{1}$$

$$DTMF_t = \begin{cases} +1 & \text{for } FSP_d > 2 \cdot MF_d \\ \dfrac{FSP_d - MF_d}{MF_d} & \text{for } 0 \leq FSP_d \leq 2 \cdot MF_d. \\ -1 & \text{for } FSP_d < 0 \end{cases} \tag{2}$$

The FcB captures the prediction behavior of customers, whether they tend to predict more or less than they consume. FcB on the day t is represented as FcB_t and defined as the average of the realized difference between the customer's demand prediction (FC_t) and real demand (C_t), as shown in Equation (3).

$$FcB_t := \frac{\sum_{t'<t}(FC_{t'} - C_{t'})}{\sum_{t'<t}|FC_{t'} - C_{t'}|}. \tag{3}$$

The NAS provides the DRL algorithm's output before scaling such that $scaling(NAS_t) = max(0, ZF_d - FSP_d)$. We defined this state because the continuous action scaling explained in Section 2.2 makes the meaning of the DRL algorithm's output before scaling different for each product. The NAS is defined in Equation (4).

$$NAS_t := \left(\frac{max(0, ZF_d - FSP_d)}{PackingSize} - \frac{a^d_{max} + a^d_{min}}{2} \right) \times \frac{2}{a^d_{max} - a^d_{min}}. \tag{4}$$

Transition Function Given a state s_t, the agent defines an action on the day d, denoted as $a_d \in [0, 1, \ldots, a^d_{max}]$. The action a_d is the integer number of packages sent on day d and is bounded by the maximum package number a^d_{max}. The transition function $P_{a_d}(s_t, s_{t+1})$ defines the probability that the state s_t moves into the next state s_{t+1} when taking action a_d. By transitioning from state s_t to s_{t+1}, the stock position on day $t+1$ is updated as $CSP_{t+1} = CSP_t + FR_{t+1} - C_{t+1}$, where C_{t+1} is the real demand instead of demand prediction FC_{t+1}, introducing a randomness to the transition. The future replenishment and the demand predictions for the next $OLT - 1$ days are shifted by one day from s_t to s_{t+1}. The future replenishment on the day d is set to be the chosen action a_d. The environment dynamics gives the demand forecast on day d, the minimum and maximum level on day $d+1$, the maximum number of available shipments on day $d+1$, and the accumulated customer forecast behavior until day $t+1$. Note that the explicit distribution of the transition function is unknown from the DRL agent.

Reward Function Given the transition from state s_t to s_{t+1} by action a_d, the agent receives the reward. While the state represents the predicted VMI condition at decision time t, the reward is based on the realized inventory on the replenishment date d. The reward function r_d is defined such that it is maximum if the stock level on the day d, CSP_d, is at the mean inventory level M_d. Here, we compute M_d as the average of minimum and maximum stock levels, z_d, and Z_d, which we derive by using forecast-based equations without incorporating OLT. We further designed the reward function to achieve a maximum value, modulated by parameters α and β (both between 0 and 1), with $Mz_d = \alpha z_d + (1-\alpha)M_d$ setting minimum stock levels and $MZ_d = \beta Z_d + (1-\beta)M_d$ setting maximum stock levels. This structure is purposefully devised to pinpoint inventory positions relative to the established thresholds. Lastly, we normalize the reward, ensuring a maximal reward when inventories align with M_w and penalize deviations from the mean level (DTM_d), particularly below the minimum stock level. The reward function is shown in Equation (5).

$$r_d = \begin{cases} -1 + \frac{2Z_d - CSP_d}{2Z_d - Z_d} & \text{if } Z_d \leq CSP_d \leq 2Z_d \\ \frac{Z_d - CSP_d}{Z_d - MZ_d} & \text{if } MZ_d \leq CSP_d \leq Z_d \\ 1 & \text{if } Mz_d \leq CSP_d \leq MZ_d \\ \frac{z_d - CSP_d}{MZ_d - z_d} & \text{if } z_d \leq CSP_d \leq Mz_d \\ -1 + \frac{CSP_d}{z_d} & \text{if } 0 \leq CSP_d \leq z_d \\ -1 & \text{elsewhere} \end{cases} \tag{5}$$

In the reward function above, service levels were solely considered for determining the reward. However, in a practical scenario, the number of shipments also plays a crucial role in inventory management. Too many shipments could be problematic and unrealistic, leading to increased transportation costs, warehouse management complexity, and a higher carbon footprint. To account for this aspect, we introduced a penalty factor for a positive number of shipments. A non-negative constant parameter $p \in [0,2]$ determines the strength of the penalty and modifies rewards to $r_d \leftarrow \max(r_d - p, -1)$ if the model opts to ship products to the customer. The higher the value of this parameter, the more heavily the model penalizes frequent shipments.

2.2 Multi-Scenario Approach & Action Space

Employing a multi-scenario approach in VMI systems is advantageous to adapt to real-world settings as it can capture multiple product/customer combinations within a single framework, leading to reduced maintenance and a more efficient use of resources. The semiconductor industry, characterized by diverse products, benefits significantly from a multi-scenario approach. The primary aim of this model is to sidestep the computational burden and time-intensive process associated with training individual models for each product-customer combination. Opting for a multi-scenario approach that can adjust to various environments provides a scalable and efficient strategy for optimizing inventory management practices. While the multi-scenario approach is designed for a broad application, carefully considering the trade-off between scalability and the level of performance specific to each scenario is necessary. While it may not attain the same high performance as several individual models, fine-tuned for particular situations, the multi-scenario approach offers a significant advantage by reducing the complexity and resource demands of the system's operation. Evaluating this trade-off is crucial for assessing the model's practicality in managing VMI. For the multi-scenario approach, we utilize the methodology described in Section 2.1, with adjustments as described in the following two subsections.

Continuous Action Space Adopting a continuous action space within the multi-scenario approach for VMI is essential when addressing multiple product-customer scenarios. Unlike a discrete action space, which can lead to computational challenges and instable convergence when growing in size, a continuous action space allows for flexible product granularity and more fluid optimization. In a discrete setting, the softmax activation function is commonly employed in the DRL algorithm's output layer, but as the action space expands, this may impede convergence. A continuous action space overcomes this hurdle by facilitating incremental adjustments and smoother transitions between decisions. To map the continuous output of the model into a discrete set of actions, we utilize the *tanh* function for activation. This mapping is crucial as it aligns the continuous model's output with the practical discrete actions required for VMI. The mapping is described by Equation (6):

$$a = \begin{cases} \tanh(a_{\text{model}}) \times \frac{a_{\max} - 1}{2} + \frac{a_{\max} + 1}{2} & \text{if } \tanh(a_{\text{model}}) \geq \alpha \\ 0 & \text{otherwise} \end{cases}. \tag{6}$$

Here, a_{model} is the continuous output from the model subjected to the *tanh* function. The result is then scaled to the desired action range between 0 and $a_{\max}$. Notably, $a_{\max}$ is dynamic and adjusted by the maximum stock levels, denoted by $Z_w F$, relevant at the moment of decision-making. This ensures that the model's output is appropriately calibrated to the practical constraints of inventory levels. Since a is lastly rounded to an integer and the higher $a_{\max}$ value decreases the range of $\tanh(a_{\text{model}})$ mapped to less than 0.5 by scaling, it leads many non-zero shipments. Thus, the mapping includes a constant $\tanh(a_{\text{model}})$ domain mapped to 0. The α value is set to -0.75 in our numerical experiments as $\tanh(a_{\text{model}}) \in (-1, 1)$.

Integrating a continuous action space enables the multi-scenario approach to more effectively manage the complexities of diverse product-customer combinations in VMI. Such a space provides the model with the opportunity for more precise and nuanced decision-making tailored to the varying inventory requirements across different scenarios. Additionally, by configuring specific action spaces reflecting each product-customer environment, the model can accommodate each scenario's unique characteristics and constraints. Combining the continuous action space with environment-specific considerations empowers the multi-scenario approach to generalize its decision-making process across various settings, optimizing its effectiveness in VMI systems.

Learning Process of the Multi-Scenario Approach The multi-scenario approach adapts to various product-customer environments using the TD3 algorithm, which is adept at managing continuous action spaces. Throughout the training, the model undergoes iterative learning episodes that simulate the full cycle of interactions within a given product-customer scenario. Through repeated cycles, the model learns to generalize its decision-making to manage inventory across diverse settings efficiently. The learning process involves the model engaging with both training and validation environments. In training, it refines its policy by observing the consequences of its actions — the rewards and state transitions. During validation, the model's exploration mechanisms, such as epsilon-greedy, are temporarily suspended, and the neural network's output layers are fixed to evaluate its decision-making. This evaluation phase is crucial as it provides insights into the model's performance without further learning interference. After each validation phase, the exploration strategy is re-engaged, and the model continues to learn from the training environment. This cyclical process between active learning and performance validation ensures that the model consistently improves its ability to navigate and optimize within various scenarios. Hyperparameter optimization is a critical aspect of fine-tuning the model's learning algorithm. This is achieved through Bayesian Optimization (BO), utilizing the Tree-Structured Parzen Estimator (TPE) method, which systematically explores the hyperparameter space to identify the most effective combinations for training the model. To gauge the model's efficacy in managing VMI systems, the training process is monitored by tracking KPIs. These include metrics like accumulated rewards, shipment volumes, and service level achievements. Losses encountered during training are also recorded. By plotting and scrutinizing these KPIs, the model's proficiency in inventory management is continuously assessed, allowing for ongoing refinements that enhance overall performance. The diligent observation of KPIs is instrumental in the iterative improvement of the model throughout the learning phase.

2.3 DRL Algorithms

VMI involves complex, multi-faceted decision environments where the supplier must consider various factors such as inventory levels, demand forecasts, lead times, and service levels. DRL is adept at handling such complexity. Thus, our methodology hinges on the utilization of DRL starting with the implementation of the DQN algorithm, as introduced by Mnih et al. (2015). DQN utilizes a neural network to approximate the optimal action-value function, crucial for determining the most advantageous action in a given state. The main improvements of DQN are the employment of experience replay and fixed Q-targets, which are techniques developed to stabilize the learning process.

TD3, building on the Deep Deterministic Policy Gradient (DDPG) algorithm is an actor-critic method optimized for continuous action domains (Fujimoto et al. 2018). TD3 refines this approach by correcting function approximation errors that impact policy and value updates. This is done by introducing twin

critic networks that work to reduce overestimation bias and by delayed policy updates, which enhance the stability of the learning process. The update mechanism for the critic networks in TD3 is given by Equation (7):

$$Q(s,a) \leftarrow Q(s,a) + \alpha \cdot \big(r + \gamma \cdot \min(Q_1(s', \pi(s';\theta_\pi)), Q_2(s', \pi(s';\theta_\pi))) - Q(s,a)\big) \tag{7}$$

While the DQN provides the foundation for DRL, we use the TD3 algorithm as the the focal point of the study in this work. For VMI planning, TD3 stands out because its sophisticated mechanisms ensure more stable and accurate policy learning Fujimoto et al. (2018).

The DQN model developed by Ahmad et al. (2022) proposes a heuristic for consolidating replenishments. Yet the total shipment was much higher than the current policy. Therefore, we add a shipment penalty to our model and reduce the resulting reward by pre-determined hyperparameter $p \in [0,2]$ whenever the DRL algorithm takes a positive action. If the reward becomes less than -1, then the reward is clipped to -1 to keep the range between $[-1,1]$.

The reward is calculated based on the minimum and maximum stock levels, which is determined by the customer forecast on that day. If there is a sudden forecast change, it is impossible to maintain the stock level to be inside the min/max level, which makes it difficult for the DRL algorithm to learn the optimal policy due to the negative reward regardless of the chosen action. Thus, we stabilize the reward function by defining the min/max level for the reward function as the average around the day (±2 days in our numerical experiment), making the model possible to get positive rewards by appropriate action.

The DRL algorithm training is time-consuming, especially for the multi-scenario approach. Pre-training can help to overcome this issue by providing a better initial neural network. The pre-training data should be carefully chosen as it will determine the behavior of the pre-trained model. When considering for which conditions the pre-trained model should work well, the inventory error is the supplier's responsibility only if the forecast accuracy and bias-corrected forecast accuracy of customer prediction are high enough. Therefore, we generated customer forecast data from real pull data with some randomness so that the data has a high and bias-corrected forecast accuracy, resulting in the trained model performing well in such conditions. The randomness is sampled from truncated standard normal distribution so that the expectation matches the real demand.

3 EXPERIMENTAL DESIGN

This paper focuses on the 28-day OLT, which offers a more realistic assessment of TD3, considering backend production. We use a planning horizon of 847 days (121 weeks) for the training phase with weekly forecasts and daily pulls, while the test data contains 280 days (40 weeks), for products A, B, and C. We add a warm-up period of 4 weeks at the start of the testing horizon, during which the DRL algorithms plan replenishments but are not evaluated. The initial inventory is set to zero at the first replenishment date. We evaluate performance via three KPIs: PNV, α-service level (α-SL), and β-service level (β-SL), based on Danese (2004) and Afridi et al. (2020). The three KPIs measure percentages of the days within the overall forecast horizon that VMI satisfies certain conditions. PNV refers to the percentage of inventory level inside the predefined minimum and maximum inventory levels (NV state); α-SL represents the ratio of the inventory level not being stock-out (SO); and β-SL measures the proportion of demand met from stock, reflecting the system's effectiveness in fulfilling customer orders without delays. It is important to note that while α-SL and β-SL primarily focus on violations of the minimum level (understock or stockout), they do not consider violations of the maximum level. More precisely, they are defined as follows:

$$PNV = \frac{\sum_d \mathbb{1}_{s_d = NV}}{T}, \tag{8}$$

$$\alpha\text{-SL} = \frac{\sum_d \mathbb{1}_{s_d \neq SO}}{T}, \tag{9}$$

$$\beta\text{-SL} = \frac{\sum_d \min(\frac{CSP_d}{D_d}, 1)}{T}. \tag{10}$$

In Equations (8)–(10), T represents the total number of forecasting days, s_d is the state of VMI on the day d if the inventory is within the minimum and maximum inventory levels (NO) or stock-out (SO), D_d is the demand on day d, CSP_d is the inventory level on day d. In addition to these KPIs, the number of shipments within the entire period is considered in this study to maintain a realistic level of control.

We split the evaluation into two parts. First, we use the same three products as in Ahmad et al. (2022) for performance comparison. Product A represents the same product and customer combination as in Afridi et al. (2020), product B represents a product with the same customer, and product C is the same product as product A but with a different customer. The demand patterns for these products vary, with product A having the most stable demand, product B showing demand decrease over time, and product C having a significant increase in demand after the first few weeks. For the multi-scenario approach, we select 100 different product-customer combinations without missing values in the customer forecast and pull data. This data has various demand patterns and differs from products A, B, and C. Each combination's time horizon is divided into training and validation sets, maintaining an 80%-20% ratio. The training horizon spans again 847 days (121 weeks), and the test data contains 280 days (40 weeks), keeping a warm-up period of 4 weeks at the start of the testing horizon.

4 RESULTS

4.1 Product / Customer specific results

Each product-customer-specific model was trained with 2000 iterations, with the same network architecture and hyperparameters. Tables 1–3 show the resulting KPIs of the current policy, DQN results from Ahmad et al. (2022), and the TD3 with different penalties, which show the mean and standard deviation of ten different random seeds. For each product, the TD3 model without a penalty shows better PNV than the current policy and the DQN. Larger penalty values generally lead to fewer shipments and lower PNV. The α-SL and β-SL of the TD3 models are almost 100%, or close to it. The number of shipments is close to the current policy if the penalty value is higher. Product A with TD3 and a penalty of 0.75 achieves 20% higher PNV than the current policy while the number of shipments is at the same level. Additionally, our approach can follow the increase and decrease of the customer's demand, as shown in products B and C.

Table 1: Results of product A.

KPI	Current policy	DQN	TD3 $p = 0.00$	TD3 $p = 0.25$	TD3 $p = 0.5$	TD3 $p = 0.75$
PNV	53%	60%	$82.9\% \pm 3.3$	$80.8\% \pm 3.1$	$78.9\% \pm 3.1$	$73.6\% \pm 9.3$
α-SL	100%	100%	$100\% \pm 0$	$100\% \pm 0$	$100\% \pm 0$	$100\% \pm 0$
β-SL	97.1%	99.6%	$99.6\% \pm 0$	$99.6\% \pm 0$	$99.6\% \pm 0$	$99.6\% \pm 0$
Total shipments	29	122	70.5 ± 11.0	55.3 ± 19.8	45.4 ± 15.7	29.4 ± 9.2

Table 2: Results of product B.

KPI	Current policy	DQN	TD3 $p = 0.00$	TD3 $p = 0.25$	TD3 $p = 0.5$	TD3 $p = 0.75$
PNV	54%	55%	$56.3\% \pm 1.3$	$59.1\% \pm 2.9$	$53.5\% \pm 4.6$	$56.6\% \pm 2.3$
α-SL	100%	100%	$99.9\% \pm 0.2$	$99.3\% \pm 1.6$	$98.0\% \pm 1.9$	$98.7\% \pm 2.7$
β-SL	99.3%	99.6%	$99.6\% \pm 0.15$	$99.2\% \pm 0.9$	$98.3\% \pm 1.1$	$98.9\% \pm 1.5$
Total shipments	35	68	65.7 ± 13.4	53.9 ± 21.4	44.2 ± 14.8	46.4 ± 16.8

The product-specific model results indicate a trend that large penalties lead to lower PNV and fewer shipments. In this context, we conduct a sensitivity analysis for product A concerning its penalty, where ten different TD3 models with the same hyperparameters but with different random seeds are trained for

Table 3: Results of product C.

KPI	Current policy	DQN	TD3 $p = 0.00$	TD3 $p = 0.25$	TD3 $p = 0.5$	TD3 $p = 0.75$
PNV	45%	60%	$63.8\% \pm 3.5$	$69.9\% \pm 3.3$	$67.2\% \pm 5.1$	$66.2\% \pm 6.9$
α-SL	100%	100%	$99.8\% \pm 0.3$	$99.3\% \pm 1.3$	$99.5\% \pm 0.3$	$99.5\% \pm 0.3$
β-SL	100%	99.6%	$99.4\% \pm 0.3$	$98.8\% \pm 0.7$	$99.0\% \pm 0.3$	$99.0\% \pm 0.5$
Total shipments	34	280	64.6 ± 48.6	48 ± 11.5	$61.8\% \pm 31.4$	$54.4\% \pm 36.6$

each penalty, and their average KPIs are calculated. The penalties are tested from 0 to 2 for each 0.05 step as the reward range ($[-1, 1]$) leads to the maximum penalty of 2. Figure 2 shows the respective trade-off between the number of shipments and the PNV. For penalties larger than $\sim$1.0, the number of shipments converges around the current policy levels, while the PNV is much larger than the current policy. Note that we do not report the α-SL and β-SL since they are 100 %, except for a penalty of 1.8.

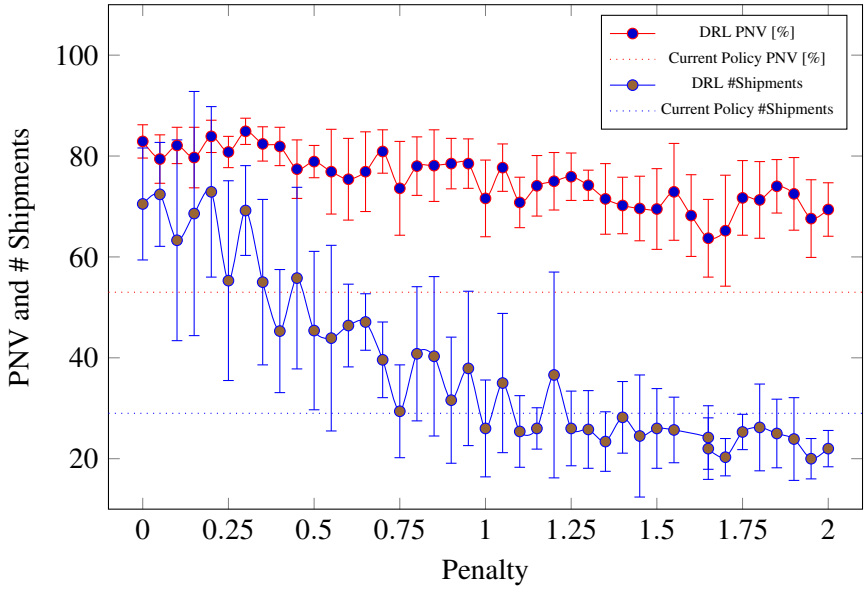

Figure 2: Sensitivity analysis of product A, focusing on the penalty term with standard error included.

4.2 Multi-Scenario Approach

The multi-scenario approach was trained with 20 iterations, where one hundred product/customer combinations were sequentially trained within each iteration. The realized experiences are saved in one replay memory during the training and sampled to update the network. Table 4 shows the resulting KPIs of the current policy and the TD3 algorithm with different penalty values trained for multiple scenarios. Similar to the product-specific models' results, the larger penalty leads to lower PNV and fewer total shipments. TD3 with a penalty of 0.75 has a lower number of total shipments than the current policy, while the PNV is 7% higher.

Takeaway 1: The multi-scenario approach leads to reduced maintenance efforts in operations Implementing a multi-scenario approach simplifies operational maintenance by reducing the need for multiple individual models. This "one-size-fits-all" approach ensures that a single model is sufficient, unlike the traditional method of training different models for each product/customer combination. This leads to a more efficient use of resources and streamlines the model management process.

Table 4: Results of the multi-scenario approach.

KPI	Current policy	DQN	TD3 $p = 0.00$	TD3 $p = 0.25$	TD3 $p = 0.5$	TD3 $p = 0.75$
PNV	28.2%	-%	$35.1\% \pm 1.3$	$36.2\% \pm 0.8$	$36.1\% \pm 0.3$	$35.15\% \pm 0.5$
α-SL	91.05%	-%	$96.1\% \pm 0.6$	$96.8\% \pm 0.8$	$96.6\% \pm 1.0$	$96.2\% \pm 0.4$
β-SL	98.60%	-%	$99.3\% \pm 0.2$	$99.4\% \pm 0.1$	$99.3\% \pm 0.2$	$99.3\% \pm 0.1$
Total shipments	57.9	-	89.3 ± 7.8	89.2 ± 17.4	72.6 ± 14.9	44.1 ± 5.7

Takeaway 2: Achieving higher customer satisfaction by achieving higher service levels Increasing SLs directly contributes to enhanced customer satisfaction, as consistently demonstrated in Table 4. In the multi-scenario approach, PNV and α-SL showed marked improvements. Notably, we achieve an increase of α-SL and PNV of at least 5%-points.

Takeaway 3: Multi-scenario approach outperforms the current implementation The multi-scenario approach exhibits superior performance over the current implementation, demonstrating enhanced effectiveness. While a customized approach for each product/customer pair remains optimal, it conflicts with "Takeaway2", which favors a more generalizable model application, still outperforming the current implementation.

There are product/customer combinations where neither the current policy nor the multi-scenario approach can perform well. Still, the multi-scenario approach outperforms the current policy. Upon careful examination, it was discovered that the complete dataset contained several data anomalies, which proved challenging in achieving optimal PNV scores. Three distinct categories of these anomalous data conditions were identified.

1. unreliable prediction: customer prediction is too unreliable.
2. zero max: if the maximum inventory level falls below the packing size, any replenishment inevitably results in an inventory level exceeding the prescribed minimum and maximum levels, negatively impacting the PNV.
3. high deviation: the minimum and maximum levels change too frequently. Maintaining inventory levels within the minimum and maximum range for two consecutive days, it becomes particularly challenging to stay within the boundaries if the minimum level of the first day is significantly higher than the maximum level of the following day and vice versa.

We trained a multi-scenario approach for products A, B, and C only to check the multi-scenario approach's performance without data anomalies. The results for the product-specific DQN model are derived by calculating the average values of the KPIs presented in Tables 1, 2, and 3. The TD3 algorithm was trained on 500 iterations, where the data of products A, B, and C were sequentially trained within each iteration. The model's experiences were saved in a (shared) replay memory and sampled for network updates as part of the training process. Finally, our KPIs were determined by averaging the results of the three datasets. Table 5 shows the resulting KPIs of the multi-scenario approach for products A, B, and C. As with the product-specific results and the results of the multi-scenario approach, the choice of a larger penalty leads to lower total shipments. However, the PNV is still higher than the current policy.

Takeaway 4: Shipment penalties lead to comparable shipments of the current approach Higher penalties for shipments lead to a comparable number of shipments to those of the existing approach. Additionally, we see a decrease in the number of shipments of up to 23% compared to the mean value (see Table 4). For products A, B, and C, there is an increase of 48% compared to the mean value (see Table 5).

5 CONCLUSION

In this work, we propose a novel multi-scenario DRL algorithm with a continuous action space, developed to efficiently handle diverse product/customer combinations and enhance VMI performance. A penalty for

Table 5: Results of multi-scenario approach for product A, B, and C.

KPI	Current policy	DQN	TD3 $p = 0.00$	TD3 $p = 0.25$	TD3 $p = 0.5$	TD3 $p = 0.75$
PNV	50.6%	58.33%	$63.4\% \pm 11.5$	$64.8\% \pm 12.1$	$63.8\% \pm 12.0$	$61.3\% \pm 11.7$
α-SL	100%	100%	$99.7\% \pm 0.5$	$99.8\% \pm 0.3$	$99.5\% \pm 0.9$	$99.8\% \pm 0.6$
β-SL	99%	100%	$99.3\% \pm 0.0$	$99.5\% \pm 0.9$	$99.3\% \pm 0.5$	$99.4\% \pm 0.4$
Total shipments	32.6	156.7	71.3 ± 32.6	62.8 ± 20.9	50.0 ± 23.3	48.0 ± 26.9

the shipment frequency is introduced for TD3 to address a possible increased number of shipments. TD3 is evaluated, first, on 100 product/customer combinations, and second on three product/customer combinations from the semiconductor industry partner firm. The proposed method is compared to the current policy and a product-specific DQN model at a semiconductor manufacturer and outperforms it, demonstrating significant improvements in maintaining inventory levels.

A few comments on limitations of our work are in order, including the limited data quality from customers. Using VMI in the semiconductor industry, the semiconductor manufacturer heavily relies on the forecast quality of their customers due to the high lead times. Therefore, if the forecast accuracy is low, even advanced DRL algorithms cannot improve VMI performance. Future research should develop a strategy that defines under which circumstances DRL is beneficial and when it is preferable to keep the existing methods, especially in scenarios characterized by poor forecast accuracy or a low number of data points. Additionally, the utilization of a reach-based replenishment strategy, as implemented in VMI, has been shown to enhance the BWE, particularly during periods of disruptions, as evidenced by Ehm et al. (2023). Therefore, conducting a comparative analysis of our algorithm's performance during disruptions against the suggested absolute value replenishment method by Ehm et al. (2023) would provide valuable insights into the stability of our algorithm under similar challenging conditions.

Despite these limitations, our research contributes valuable insights into supply chain management, particularly in the semiconductor industry. We show that by employing advanced DRL techniques and the multi-scenario approach for VMI, companies can enhance their operational efficiency, responsiveness to market demand, and overall supply chain performance. The proposed multi-scenario approach has the potential to streamline inventory management across multiple products, increasing competitive advantages for businesses.

REFERENCES

Afridi, M. T., S. Nieto-Isaza, H. Ehm, T. Ponsignon and A. Hamed. 2020. "A deep reinforcement learning approach for optimal replenishment policy in a vendor managed inventory setting for semiconductors". In *2020 Winter Simulation Conference (WSC)*, 1753–1764 https://doi.org/10.1109/WSC48552.2020.9384048.

Ahmad, F., S. Nieto-Isaza, M. Ratusny, and E. Hans. 2022. "Application of Deep Reinforcement Learning for Planning of Vendor-Managed Inventory for Semiconductors". https://informs-sim.org/wsc22papers/by_auth.html.

Boute, R. N., J. Gijsbrechts, W. Van Jaarsveld, and N. Vanvuchelen. 2022. "Deep reinforcement learning for inventory control: A roadmap". *European Journal of Operational Research* 298(2):401–412 https://doi.org/https://doi.org/10.1016/j.ejor.2021.07.016.

Danese, P. 2004. "Beyond Vendor Managed Inventory: the Glaxosmithkline Case". *Supply Chain Forum: An International Journal* 5(2):32–40 https://doi.org/10.1080/16258312.2004.11517131.

Ehm, H., C. H. Chung, S. Kar Chowdhury, M. Ratusny and A. Ismail. 2023. "The Bullwhip Effect in End-To-End Supply Chains: The Impact of Reach-Based Replenishment Policies with a Long Cycle Time Supplier". In *2023 Winter Simulation Conference (WSC)*, 2194–2205 https://doi.org/10.1109/WSC60868.2023.10407267.

Fry, M. J., R. Kapuscinski, and T. L. Olsen. 2001. "Coordinating production and delivery under a (z, Z)-type vendor-managed inventory contract". *Manufacturing & Service Operations Management* 3(2):151–173 https://doi.org/10.1287/msom.3.2.151.9989.

Fujimoto, S., H. Hoof, and D. Meger. 2018. "Addressing function approximation error in actor-critic methods". In *International conference on machine learning*, 1587–1596. PMLR.

Fujimoto, S., H. van Hoof, and D. Meger. 2018, 10–15 Jul. "Addressing Function Approximation Error in Actor-Critic Methods". In *Proceedings of the 35th International Conference on Machine Learning*, edited by J. Dy and A. Krause, Volume 80 of *Proceedings of Machine Learning Research*, 1587–1596: PMLR.

Hsu, C.-Y., W.-J. Chen, and J.-C. Chien. 2020. "Similarity matching of wafer bin maps for manufacturing intelligence to empower industry 3.5 for semiconductor manufacturing". *Computers & Industrial Engineering* 142:106358 https://doi.org/https://doi.org/10.1016/j.cie.2020.106358.

Kara, A. and I. Dogan. 2018. "Reinforcement learning approaches for specifying ordering policies of perishable inventory systems". *Expert Systems with Applications* 91:150–158 https://doi.org/https://doi.org/10.1016/j.eswa.2017.08.046.

Lee, D.-H., J.-K. Yang, C.-H. Lee, and K.-J. Kim. 2019. "A data-driven approach to selection of critical process steps in the semiconductor manufacturing process considering missing and imbalanced data". *Journal of Manufacturing Systems* 52:146–156 https://doi.org/https://doi.org/10.1016/j.jmsy.2019.07.001.

Lotfi, R., M. Rajabzadeh, A. Zamani, and M. S. Rajabi. 2022. "Viable supply chain with vendor-managed inventory approach by considering blockchain, risk and robustness". *Annals of Operations Research*:1–20 https://doi.org/https://doi.org/10.1007/s10479-022-05119-y.

Mnih, V., K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare *et al*. 2015. "Human-level control through deep reinforcement learning". *nature* 518(7540):529–533 https://doi.org/https://doi.org/10.1038/nature14236.

Mohamadi, N., S. T. A. Niaki, M. Taher, and A. Shavandi. 2024. "An application of deep reinforcement learning and vendor-managed inventory in perishable supply chain management". *Engineering Applications of Artificial Intelligence* 127:107403 https://doi.org/https://doi.org/10.1016/j.engappai.2023.107403.

Monostori, J. 2018. "Supply chains robustness: Challenges and opportunities". *Procedia CIRP* 67:110–115 https://doi.org/https://doi.org/10.1016/j.procir.2017.12.185. 11th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 19-21 July 2017, Gulf of Naples, Italy.

Oroojlooyjadid, A., M. Nazari, L. V. Snyder, and M. Takáč. 2022. "A Deep Q-Network for the Beer Game: Deep Reinforcement Learning for Inventory Optimization". *Manufacturing & Service Operations Management* 24(1):285–304 https://doi.org/10.1287/msom.2020.0939.

Sui, Z., A. Gosavi, and L. Lin. 2010. "A Reinforcement Learning Approach for Inventory Replenishment in Vendor-Managed Inventory Systems With Consignment Inventory". *Engineering Management Journal* 22(4):44–53 https://doi.org/10.1080/10429247.2010.11431878.

Sun, R., P. Sun, J. Li, and G. Zhao. 2019. "Inventory Cost Control Model for Fresh Product Retailers Based on DQN". In *2019 IEEE International Conference on Big Data (Big Data)*, 5321–5325 https://doi.org/10.1109/BigData47090.2019.9006424.

Sutton, R. S. and A. G. Barto. 2018. *Reinforcement Learning: An Introduction*. MIT press.

AUTHOR BIOGRAPHIES

MARCO RATUSNY is a Ph.D. Candidate in the Corporate Supply Chain organization of Infineon Technologies AG. He received his B. Sc. (2018) in Information Systems and Management and his M. Sc. (2020) in Stochastic Engineering in Business and Finance at the Munich University of Applied Sciences, Germany. His research interest is in demand planning. His email address is marco.ratusny@infineon.com.

JEE HYUNG KIM is a digital supply chain specialist in the Automotive Division of Infineon Technologies AG. He received his B. A. (2020) in Global Business Administration at the Sungkyunkwan University, Korea, and his M. Sc. (2024) in Management and Technology at the Technical University of Munich, Germany. His research interest is in demand forecasting and inventory management. His email address is jeehyung.kim@infineon.com.

HAJIME SEKIYA was a working student in the Corporate Supply Chain organization of Infineon Technologies AG. He received his B. Ed. (2021) in Mathematics at the Tokyo Gakugei University, Japan, and his M. Sc. (2024) in Applied Mathematics at the Technical University of Munich, Germany. He is currently a Ph.D. candidate at the University of Hagen. His research interest is in planning and optimization. His email address is hajime.sekiya@fernuni-hagen.de.

MAXIMILIAN SCHIFFER is a tenured Professor for Business Analytics & Intelligent Systems at the Technical University of Munich and a core member of the Munich Data Science Institute. His research interests range from data-driven optimization to deep learning for various applications. His email address is schiffer@tum.de.

HANS EHM is a Senior Principal Engineer Supply Chain of Infineon Technologies AG. For four decades, he has been in the semiconductor industry in multiple positions in frontend, backend, and supply chain. For one decade, he has been heading the Supply Chain Innovation department of Infineon Technologies AG. His email address is hans.ehm@infineon.com.