Omnichannel Supply Chains Amid Demand Shocks: A Centralized Hierarchical Reinforcement Learning Framework

Panagiotis G. Giannopoulos and Thomas K. Dasaklis *

Posted Date: 17 March 2026

doi: 10.20944/preprints202603.1244.v1

Keywords: omnichannel supply chain; demand shocks; hierarchical reinforcement learning; proximal policy optimization; lateral transshipment; resilience; capacity constraints; jump-diffusion demand

 doi:10.20944/preprints202603.1244.v1

Article

Omnichannel Supply Chains Amid Demand Shocks: A Centralized Hierarchical Reinforcement Learning Framework

Panagiotis G. Giannopoulos [1] **and Thomas K. Dasaklis** [1,*]

School of Social Sciences, Hellenic Open University
* Correspondence: dasaklis@eap.gr

Abstract

Background: The rapid evolution of omnichannel retailing has reshaped retail supply chains (SCs) by tightly coupling replenishment, fulfillment, and service decisions across multiple demand channels under inventory, lead-time, and capacity constraints. These interdependencies create complex coordination challenges, particularly when demand shocks interact with limited operational capacity. **Methods**: To address these challenges, this study develops a centralized Hierarchical Reinforcement Learning (HRL) control framework that makes decision timing explicit: replenishment and allocation are optimized weekly, while fulfillment and lateral inventory rebalancing are controlled daily. Policies are learned using Proximal Policy Optimization (PPO) in an actor–critic architecture with bounded stochastic policies suitable for constrained action spaces. To mitigate the curse of dimensionality often encountered in HRL, we introduce a capacity-aware state–action encoding mechanism that compresses the control interface into structured summary signals. Demand shocks are modeled using two specifications: a mixed profiling where half the products follow uniform demand and half follow a Merton-type jump-diffusion process, and a fully shock-driven region. **Results**: The framework is evaluated against forecast-driven base-stock and greedy fulfillment heuristics, as well as a perfect-information oracle. Results show that the proposed encoding improves learning efficiency and scalability, achieving higher profit and service performance than the full-observation alternative. **Conclusions**: Overall, hierarchically timed control outperforms heuristic baselines while remaining below the oracle bound, with the largest gains observed when demand shocks coincide with binding fulfillment and transfer capacities.

Keywords: omnichannel supply chain; demand shocks; hierarchical reinforcement learning; proximal policy optimization; lateral transshipment; resilience; capacity constraints; jump-diffusion demand

1. Introduction

The modern retail supply chain (SC) has experienced a fundamental shift, especially due to the rise of electronic retailing and digitally enabled fulfillment networks. Etailers (electronic retailers) serve a key role in the online shopping landscape today by providing digital platforms through which consumers can easily search, compare and purchase products online. In particular, an etailer is an online retail platform that sells products directly to consumers while potentially hosting third-party sellers on the same platform, thus combining retailing and marketplace functions [43]. In this landscape, traditional brick-and-mortar retailers and pure e-tailers increasingly function within interconnected distribution ecosystems that require tight coordination of inventory, logistics and customer service decisions across spatially dispersed nodes. Early empirical evidence indicates that retailers have been progressively restructuring their physical distribution processes to support this new environment, for example, by integrating store and distribution center inventories and leveraging retail stores as forward fulfillment nodes to enhance last-mile responsiveness [17]. These developments have significantly

 doi:10.20944/preprints202603.1244.v1

increased the operational interdependencies within retail SCs and have created new challenges for inventory positioning, demand fulfillment, and network coordination.

Building on this evolution, omnichannel retailing has emerged as a dominant paradigm in which firms simultaneously manage physical and digital channels within a unified customer experience and operational framework [4]. The rapid growth of omnichannel systems during the last decade has introduced substantial complexity due to cross-channel demand substitution, multi-location inventory coupling, and capacity-constrained fulfillment decisions. Recent research shows that commonly adopted strategies such as ship-from-store and buy-online-pick-up-in-store (BOPS) can create significant value, but their effectiveness depends critically on demand structure, cost parameters, and inventory allocation capabilities [10]. At the same time, studies examining BOPS adoption, channel coordination, and demand interactions highlight that omnichannel performance is highly sensitive to substitution effects, encroachment dynamics, and nonlinear demand behavior [14,30]. These findings underscore the need for more sophisticated operational decision frameworks capable of managing the tightly coupled and stochastic nature of omnichannel environments.

Another critical aspect of omnichannel SC systems is the temporal hierarchy of decision-making. In SC and production planning research, this idea is closely related to the notion of *temporal integration*, namely the coordination of decisions across different timescales and decision-making levels, such as strategic, tactical, and operational ones [1]. A similar temporal structuring also appears in other supply-chain functions, including forecasting, where demand estimates support inventory- and production-oriented planning, service-level management, and broader decision-support processes across the different hierarchies commonly identified in SCs [7]. These particularities are especially prevalent in multi-echelon systems that are not only spatially or organizationally distributed, but also temporally structured. Higher-level decisions are usually more aggregate and slower-moving, whereas lower-level decisions are more detailed, more reactive, and more closely tied to real-time operating conditions. Recent work on multi-layer planning similarly emphasizes that monthly, weekly, and daily decision layers serve different planning purposes and must remain aligned in order to support effective execution under uncertainty [25]. This distinction is particularly important in omnichannel SCs, where upstream decisions such as replenishment planning, inventory positioning, and allocation are subject to lead times and capacity restrictions, while downstream decisions such as fulfillment, order routing, and local transshipment must respond more quickly to realized demand.

Despite the increasing operational importance of omnichannel systems, the corresponding analytical and data-driven decision literature remains fragmented. A growing body of work has examined coordination and inventory decisions using game-theoretic models, bilevel optimization, simulation–optimization, and nonlinear programming approaches [5,11,15,18,19,24,28]. While these studies provide valuable structural insights, they mainly rely on static analytical formulations or offline optimization procedures, which are less suitable for capturing the multi-period, stochastic, and dynamically evolving nature of modern omnichannel fulfillment systems. At the same time, recent studies have begun to explore reinforcement learning (RL) and other adaptive control schemes in omnichannel settings; however, these contributions remain relatively limited in terms of network structure, multi-echelon inventory interactions, lateral rebalancing, capacity-coupled fulfillment dynamics, and the explicit representation of temporally differentiated decision layers [23,27,36,38]. This gap is particularly critical in capacity-constrained omnichannel environments, where decisions interact across multiple time scales and high-dimensional state spaces make integrated control increasingly challenging.

Motivated by these limitations, the present study develops a hierarchical RL framework to support coordinated replenishment and fulfillment decisions in capacity-constrained omnichannel retail networks. More specifically, the main contribution of the study is the development of an HRL decision framework that explicitly captures the multi-timescale nature of omnichannel operations by decomposing weekly replenishment planning and daily fulfillment control into coordinated managerial layers. Building on this core contribution, the paper also introduces an integrated omnichannel

 doi:10.20944/preprints202603.1244.v1

modeling environment that jointly considers physical stores, a centralized fulfillment center (FC), and multiple demand channels under explicit inventory and processing capacity constraints, while also incorporating lateral inter-store transshipment as a dynamic inventory rebalancing mechanism. In addition, the study evaluates the proposed framework under shock-prone demand conditions through a Merton-type process and benchmarks its performance against flat PPO, business-relevant heuristics, and a perfect-information oracle. Collectively, these elements show how hierarchical control can improve service performance and profitability in capacity-constrained omnichannel systems and, therefore, advance the methodological toolkit available for data-driven retail operations.

The remainder of the paper is organized as follows. Section 2 provides the background information and literature review, first discussing the distinction between RL and HRL through the lens of temporal abstraction and multi-timescale decision-making and then reviewing the relevant RL-centric literature on omnichannel SCs. Section 3 formulates the studied omnichannel problem, presents the network structure, decision variables, objective function and sources of uncertainty and develops the corresponding MDP representation. Section 4 introduces the proposed HRL framework, detailing the state and action spaces, the architectural design, the PPO-based implementation and the benchmarking protocol. Section 5 reports the experimental evaluation and analyzes the numerical results obtained under different capacity configurations, demand regimes and store scales, while also examining how node-specific capacities affect network resilience. Section 6 discusses the main findings, managerial implications, study limitations and future research directions. Section 7 concludes the paper.

2. Background Information and Literature Review

Our work builds on the premise that Hierarchical Reinforcement Learning (HRL) may be a well-suited solution approach for SCs, where decision hierarchies naturally arise from the underlying business model and organizational structure. In omnichannel SCs, such hierarchical policies also emerge operationally, as product flows and fulfillment responsibilities are coordinated across multiple echelons and time scales. For this reason, explicitly accounting for the temporal hierarchy of decision-making provides a stronger justification for HRL in omnichannel SCs, as HRL can align the control architecture with the multi-layer temporal logic of the system by assigning slower, higher-level policies to aggregate planning decisions and faster, lower-level policies to operational execution. Building on this perspective, this section first highlights the key differences between RL and HRL agentic schemes, through the lens of how decision timing is structured across levels. It then reviews the existing RL-centric literature relevant to the omnichannel SCs to illustrate ways forward to the existing state-of-the-art.

2.1. HRL vs RL: Temporal Hierarchy of Decision-Making in Supply Chains

RL represents one of the most-studied paradigms in the recent ML literature, designed to capture how an intelligent agent acquires decision-making competence through repeated interaction with an environment and feedback on the consequences of its actions. Most RL formulations are grounded in the Markov Decision Process (MDP) framework, which models sequential decision-making under uncertainty through the tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ the action space, $P(s' \mid s, a)$ the transition dynamics, $R(s, a)$ the reward function, and $\gamma \in (0, 1]$ a discount factor that determines the relative importance of future outcomes. At each time step t, the agent observes $s_t \in \mathcal{S}$, selects $a_t \in \mathcal{A}$ according to a policy $\pi(a \mid s)$, receives a scalar reward r_t, and transitions to s_{t+1}. The learning objective is to identify a policy that maximizes the expected long-term return, typically defined as the discounted cumulative reward $R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$, thereby balancing short-term gains and longer-term performance [41]. This functionality has found to be particularly relevant to SC management, where many problems preserve a sequential and progressive nature; accordingly. In this regard, RL schemes have been applied and shown significant potential in several dynamic problems relating to SC coordination, pricing, inventory management, and production control, among others [8,35].

 doi:10.20944/preprints202603.1244.v1

When developing an RL scheme, generally three main decisions have to be made on the design side. The first one corresponds to the selection of a single- or a multi-agent approach. This is primarily a consideration reflecting the information symmetries and asymmetries applied in SCs, since it determines how many agents interact with each other and with the environment. The second decision concerns the choice of the RL solver family, namely whether the learning scheme will be value-based, policy-based, or actor–critic. This choice is largely driven by the nature of the simulated decisions and the action representation: value-based schemes are typically more convenient when decisions are discrete, whereas policy-based schemes naturally support continuous and constrained controls (e.g., ratios, fractions, allocations). Actor–critic schemes combine the two perspectives by learning a value estimator to stabilize policy optimization [2]. The third decision relates to the architectural scheme used to organize control with respect to the environment, specifically by designing how the agent–environment interaction is structured and coordinated across components and time scales. In most of the existing cases, this typically translates to choosing between centralized versus decentralized control.

These specifications do not depend on whether the RL controller is implemented in a simple or hierarchical manner, since they are required under any RL-based formulation and mainly concern the problem definition and the solver-family choice, rather than the temporal organization of decision-making [33]. The need for elaborating on HRL schemes is primarily related to the temporal hierarchy of decision-making that naturally emerges in SC operations. In particular, it reflects a direct consequence of the so-called hierarchies in SCs, where decisions are structured across layers with different responsibilities and time scales: strategic and tactical controls are typically slow and periodic (e.g., replenishment cycles and inventory positioning), whereas operational controls are fast and reactive (e.g., daily fulfillment, transshipment adjustments, and channel assignment). In simple RL schemes, decision-making is commonly modeled on a single time grid, which forces all controls to be represented and optimized at the same cadence, despite their inherently different rhythms. In contrast, HRL explicitly accounts for this sense of timing by separating control across levels, leveraging temporal abstraction whereby a high-level policy operates on a slower clock and issues temporally extended directives that persist over multiple lower-level steps, while a low-level policy operates on a faster clock and executes operational actions conditioned on these directives [33].

Figure 1 provides a graphical illustration of a single-agent HRL scheme. At the higher level (Level $i = 1$), a manager agent acts at coarse decision epochs. Its policy $\pi^{(1)}(\cdot)$ issues a directive $g_k^{(1)}$, which remains active for $\Delta^{(1)}$ lower-level steps. At the lower level (Level $i = 2$), a worker agent acts at each primitive step. Its policy $\pi^{(2)}(\cdot)$ selects the operational action $a_t^{(2)}$. This selection is typically conditioned on the fine-grained state and the active directive, i.e., $\pi^{(2)}(a_t^{(2)} \mid s_t^{(2)}, g_k^{(1)})$. The two levels differ not only in timing. They also differ in state representation. The manager agent usually observes a coarser state space. It includes aggregated and longer-horizon summaries, as well as global context. The worker agent observes a fine operational state. This state is augmented with $g_k^{(1)}$ and progress variables (e.g., remaining budget). Rewards are generated at the primitive scale (e.g., r_t). They can be accumulated over $\Delta^{(1)}$ steps to form a macro-return for Level 1. The environment also transitions from s_t to $s_{t+\Delta^{(1)}}$ over the same window. The same structure generalizes to L levels. We index layers as $i \in \{1, \ldots, L\}$. Each agent is parameterized by a policy $\pi^{(i)}(\cdot)$. Each level emits directives $g^{(i)}$ that are consumed by the next lower level. This enables adjustable *leveling* when more than two decision layers exist in a corresponding configuration.

 doi:10.20944/preprints202603.1244.v1

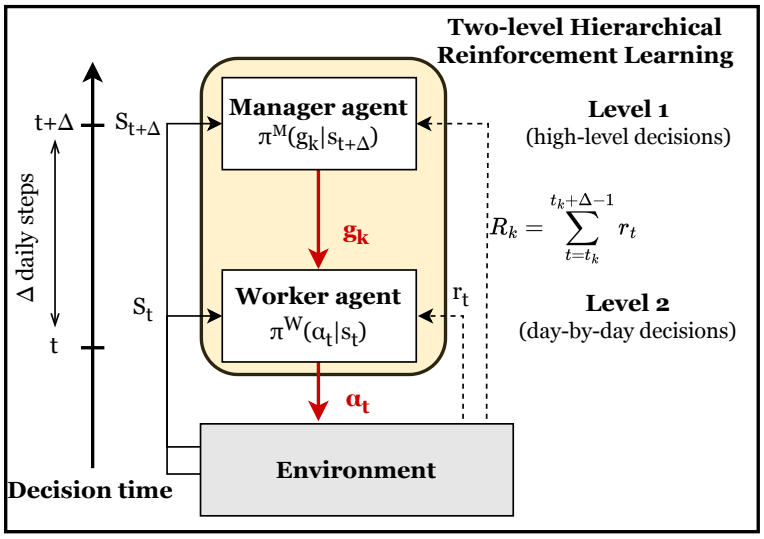

Figure 1. A conceptual representation of the interaction of Hierarchical Reinforcement Learning agents (Manager, Worker) with the environment.

2.2. RL-Centric Applications in Omnichannel SCs

A growing body of research has begun to examine RL as a decision-support approach for omnichannel retailing, particularly in problems related to replenishment, fulfillment, pricing, and inventory control. Within this stream, one line of work focuses on inventory-centric operational decisions under uncertainty, showing that learning-based methods can perform well in high-dimensional settings and can improve profitability or service-related outcomes relative to more conventional benchmarks. This is illustrated by studies on replenishment and rationing under product returns, integrated replenishment and fulfillment, and joint pricing, replenishment, and rationing, which collectively suggest that RL may offer a useful methodological direction for dynamic omnichannel control [9,23,27]. At the same time, the existing evidence should perhaps be interpreted with some caution, since these frameworks are often developed in stylized single-retailer settings and do not always capture the broader structural complexity of modern omnichannel systems.

A second emerging theme concerns the extension of learning-based methods beyond core inventory decisions toward more specialized operational and analytics-oriented applications. On the operational side, RL has been used to improve real-time store execution, for example in dynamic in-store picker routing, thereby highlighting its potential for local omnichannel fulfillment tasks [31]. On the analytics and behavioral side, recent studies have incorporated RL into models of customer behavioral uncertainty, loyalty prediction, multi-objective retail optimization, and AI-driven marketing adaptation, suggesting that learning methods are increasingly being viewed as flexible tools for data-rich omnichannel environments [36–39]. Nevertheless, these contributions appear to address rather different decision layers and objectives, and therefore do not yet amount to a fully unified stream of operational omnichannel SC control.

The reviewed studies indicate both progress and remaining fragmentation in the RL-centric omnichannel literature. While the field is clearly moving toward more adaptive and data-driven formulations, many existing contributions still provide only limited representation of multi-echelon inventory interactions, lateral stock rebalancing, capacity-coupled fulfillment coordination, and richer shock-sensitive demand environments. Thus, a reasonable reading of the current literature is that RL has shown encouraging potential across several omnichannel decision domains, but that more hierarchical, network-aware, and operationally integrated formulations are still needed in order to better reflect the tightly coupled structure of large-scale omnichannel supply networks. Moreover, although several studies adopt sequential decision formulations, the explicit representation of tempo-

 doi:10.20944/preprints202603.1244.v1

rally differentiated decision layers remains comparatively underdeveloped in the existing omnichannel RL literature.

3. Problem Formulation and Modeling scheme

As previously mentioned, the scope of our study is oriented towards developing a hierarchical RL framework as a decision-making tool in the context of mitigating the out-of-stock risks related to stochastic demand arrivals in the case of omnichannel SCs. This section first delves into the formulation and the specifics of the considered problem, which are further analyzed in the second part for defining the corresponding decision variables and building the overall objective function governing the formulated system.

3.1. Problem Formulation

The problem considered in this study, encompasses a retailing SC which operates under an omnichannel structure. Under this specification, the main decisions modeled reflect on the replenishment and fulfillment decisions that the participants should be made towards both maximizing their profitability while also keeping their customers satisfied, which in simple terms translates to the minimization of stock-out risks and the achievement of high service levels across all the supported sales channels. Similar to the most of the recent studies in this field, our formulation builds on the primitive that many stores (n) could operate downwards in the SC, all of them selling a specific number of products (m).

Beyond the stores, the rest of the actors incorporated in our scenario include a centralized FC responsible for serving the demand emerging in the physical stores, while also having the capability of directly shipping online orders to customers. Both the FC and the number of stores are assumed to be operated from a retailer, without the incorporation of an external physical entity, which in simple terms translate that not information asymmetry restrictions apply to our case. Except for these two type of actors an external warehouse is also used. This actor facilitates upstream replenishment by acting as the supplier-facing node that injects inventory into the system under non-zero lead times, thereby buffering supply variability and supporting the timely availability of stock at the FC. Both principal participants—the FC and the stores—are modeled as capacitated entities, meaning that they operate under explicit upper bounds on key operational resources. In this study, capacitation primarily refers to finite inventory holding capacity (maximum on-hand stock per product at each node) and finite processing/dispatch capacity (limits on how much inventory can be shipped or transferred within a period, e.g., daily FC-to-store shipments and lateral transshipment). These constraints are critical because they restrict feasible replenishment and fulfillment actions, forcing policies to prioritize products and channels; and manage trade off short-term service improvements against longer-term inventory positioning and cost efficiency.

Each order, in our case, is mapped to a customer. This safeguards a consistent representation of demand ownership and fulfillment responsibility. It also supports a geographical zoning of the served channels, since each customer is linked to a specific service region. Each zone represents exactly one store. Hence, for n potential stores, we consider n corresponding zones. This one-to-one mapping is deliberate. In omnichannel SCs, stores are not only selling points; they are also used as local picking and handover points for customers, especially for pickup-oriented services. Therefore, anchoring demand to store-specific zones provides a simple and interpretable way to capture spatial structure without introducing additional routing complexity. Under this specification, demand is realized through three retail channels: (i) walk-in sales (also mentioned in the literature as in-store/offline demand), (ii) click-and-collect orders (also mentioned in the literature as BOPIS—buy online, pick up in store), and (iii) online home-delivery orders.

Alongside the above channels, our model also incorporates an additional transshipment channel, designed to enable the inter-node movement of inventory across downstream locations. This channel operates under the premise that inventory sharing and re-balancing between stores can mitigate localized shortages, reduce the risk of stock-outs, and support higher service performance across

 doi:10.20944/preprints202603.1244.v1

zones. Despite its operational relevance, such inter-seller transshipment mechanisms are seldom modeled in the omnichannel literature, particularly in studies that employ RL as the primary solution approach. In this regard, our work advances the current body of knowledge by assessing the value of lateral inventory re-balancing within an RL-based omnichannel control framework, and by quantifying its impact on both profitability and service-related outcomes under realistic capacity and lead-time constraints. Figure 2 illustrates the modeling scheme designed for this study.

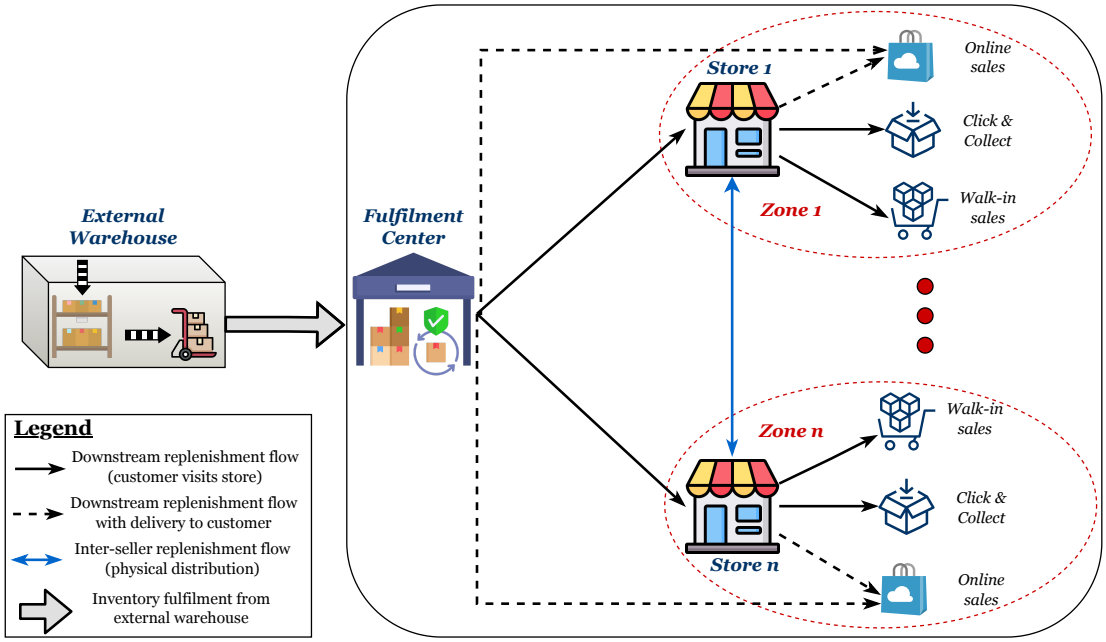

Figure 2. Illustration of the omnichannel network considered in this study, developed as an extension of the market mock-up presented in [11].

Another worth-mentioning aspect concerns the demand profiles and their implications for network performance, particularly with respect to lead-time exposure and stock-out risk. While a substantial part of the related literature relies on simplified demand assumptions, such as uniform or perfectly periodic patterns, real retail demand is often characterized by irregular surges and intermittent realizations. To reflect these empirically relevant stressors, this study considers two demand profiles, illustrated in Figure 3. The first corresponds to a uniform demand process fluctuating around a constant mean level μ. The second follows a Merton-type demand specification, in which demand shifts from a pre-shock mean level μ_1 to a higher shock-period mean μ_2, and subsequently returns to a post-shock mean μ_3, with $\mu_1 = \mu_3$ and $\mu_2 > \mu_1$. This Merton-type jump structure is particularly relevant here because it captures abrupt departures from baseline demand in a parsimonious way, thereby allowing the analysis to examine disruption-and-recovery conditions under volatile demand realizations. This modeling choice is also consistent with Liu et al. [29], who use a Merton jump-diffusion process to represent non-stationary customer demand in volatile environments. Demand is generated at the channel–product level, so that in each period, zone-specific demand is sampled separately for each retail channel and each product. This induces heterogeneous demand streams that compete for shared inventories and capacities across the FC and stores, making it possible to observe how shocks in specific products and channels propagate through the omnichannel fulfillment structure, amplify lead-time effects, and increase the likelihood of localized stock-outs.

 doi:10.20944/preprints202603.1244.v1

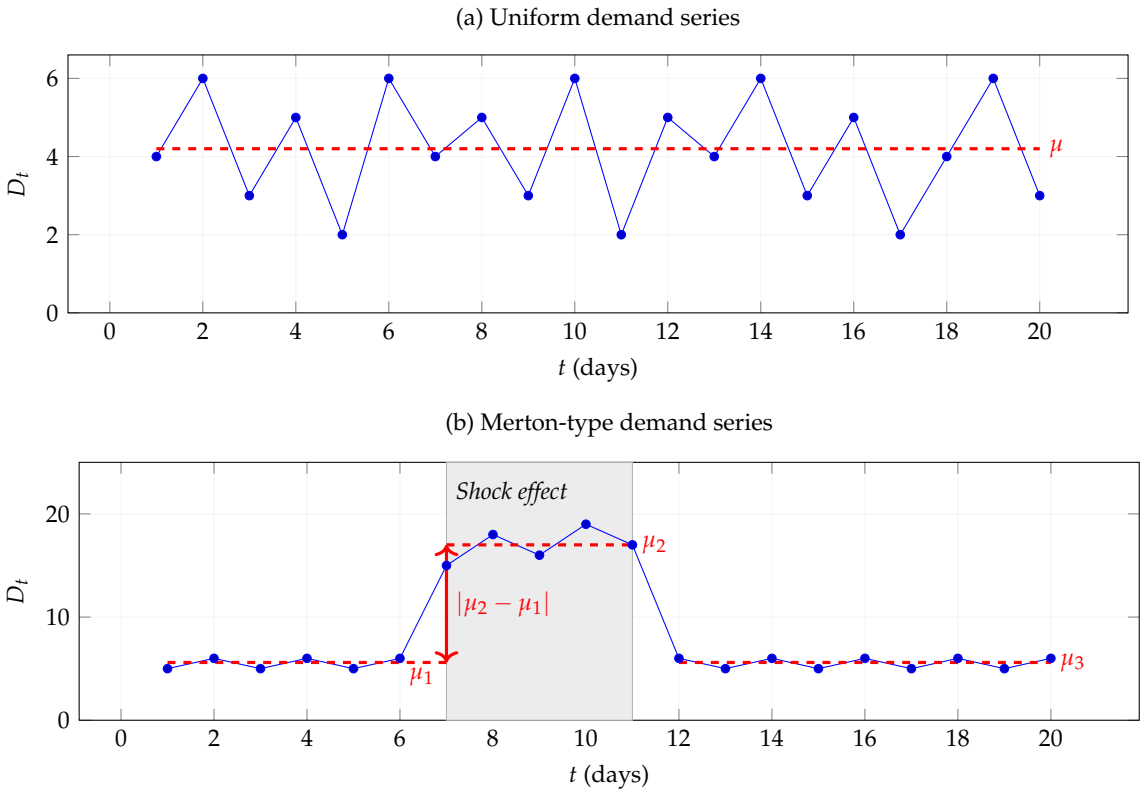

Figure 3. An illustration of the two demand profiles analyzed in this study.
Notation: Red dashed lines denote profile-specific mean demand levels. In the uniform series, demand varies around a constant mean μ. In the Merton-type series, the shaded interval marks the shock period, where the mean rises from μ_1 to μ_2 and subsequently returns to μ_3, with $\mu_1 = \mu_3$. The quantity $|\mu_2 - \mu_1|$ captures the magnitude of the shock effect.

Regarding the sequence of events and the corresponding controls on the fulfillment and replenishment cycles our model makes the following assumptions:

- Split fulfillment is not permitted; each customer order must be entirely fulfilled by a single seller (store or FC), i.e., no partial shipments or multi-origin fulfillment.
- Replenished products from the supplier are consolidated into batches, reflecting common practice where orders are placed in standardized batch sizes to speed up the retailer's operations [3]. Accordingly, cycle-level replenishment actions produced by the controller are translated by the simulator into executable batch-feasible quantities before being implemented in the environment.
- Sales are immediately lost when the inventory required to satisfy demand is unavailable at the selected fulfillment node (lost sales; no backlogging).
- Replenishment decisions are made cyclically at a coarser time scale (weekly), whereas fulfillment decisions are made in each time period of the selling horizon (daily).
- Within each day, the event order is fixed: pipeline arrivals are received first, then (if applicable) weekly controls are executed, followed by daily operational controls (e.g., transshipments and online allocation), and finally demand is realized and fulfilled.

3.2. Definition of Decision Variables, Objective Function and Sources of Uncertainty

For solving the above-discussed problem, we rely on an RL-centric framework. Since such frameworks are grounded in MDPs, this section defines the main state-transition ingredients, the operational control variables, and the objective function governing the learning process. Building on the notion of demand shocks in the network, we model an omnichannel retailing system over a finite selling horizon, where fulfillment decisions are made at every primitive time step, whereas replenishment-type decisions are activated only at designated cycle epochs. Following the background

 doi:10.20944/preprints202603.1244.v1

specification regarding the execution of HRL as presented in section 2, the same formulation can be interpreted either under a flat RL controller or under a multi-level HRL controller by appropriately defining action timing and level-specific information sets.

We consider a discrete-time horizon with index set $\mathcal{T} = \{1, \dots, T\}$, product set $\mathcal{P}$, store set $\mathcal{S}$, a centralized FC denoted by f, demand-zone set $\mathcal{Z}$, and channel set $\mathcal{C} = \{w, c, o\}$ for walk-in, click-and-collect, and online demand, respectively, while $s : \mathcal{Z} \to \mathcal{S}$ maps each zone to its serving store. Uncertainty enters through stochastic channel-specific demand, forecast error, and the induced stochastic transitions. Specifically, if $\tilde{D}^X_{tpz}$ denotes random demand and D^X_{tpz} its realization for period t, product p, zone z, and channel X, then demand evolves as $D^X_{tpz} \sim \tilde{D}^X_{tpz}$ over $\mathcal{T} \times \mathcal{P} \times \mathcal{Z} \times \mathcal{C}$.

To support control under uncertainty, the system maintains demand forecasts through a moving-average-type updating rule, instantiated here as an exponentially weighted update. In particular, the one-step-ahead forecast evolves according to Equation 1:

$$F^X_{(t+1)pz} = \alpha D^X_{tpz} + (1 - \alpha) F^X_{tpz}, \qquad \alpha \in (0, 1). \tag{1}$$

The exact role of this forecasting component in the benchmarking and learning procedures is discussed later in the paper.

The overall operations problem is formulated as a constrained profit-maximization problem in which the controller selects operational variables determining replenishment and fulfillment actions. At each time t, the controls include fulfilled quantities per channel and zone, lost-sales variables, online-routing quantities split into store-fulfilled and FC-fulfilled portions, lateral transshipment quantities between store nodes, and replenishment/allocation quantities activated only at replenishment epochs. We collect these variables into the control bundle shown in Equation 2:

$$u_t = \left(x^X_{tpz}, x^{X,\mathrm{los}}_{tpz}, x^{o,f}_{tpz}, x^{o,s}_{tpz}, \tau^{s \to s'}_{tp}, y_{tpj} \right)_{p,z,s,s',j,X}. \tag{2}$$

In this formulation, u_t represents the business-level control bundle that must ultimately satisfy the operational rules of the system.

The controls are constrained by demand accounting, inventory feasibility, non-backlogging logic, and capacity limitations. First, realized demand is either fulfilled or lost, as stated in Equation 3:

$$x^X_{tpz} + x^{X,\mathrm{los}}_{tpz} = D^X_{tpz} \qquad \forall (t, p, z, X). \tag{3}$$

Second, store-side fulfillment must remain feasible with respect to available inventory, which yields Equation 4:

$$x^w_{tpz} + x^c_{tpz} + x^{o,s}_{tpz} \le I^{s(z)}_{tp} \qquad \forall (t, p, z). \tag{4}$$

In addition, split fulfillment for online orders is not permitted, so online demand for a given (t, p, z) is assigned to at most one seller by means of a binary selector $k_{tpz} \in \{0, 1\}$, with $x^{o,f}_{tpz} \le k_{tpz} M$, $x^{o,s}_{tpz} \le (1 - k_{tpz}) M$, and $x^{o,f}_{tpz} + x^{o,s}_{tpz} = x^o_{tpz}$. In implementation terms, this selector is not treated as an independently emitted primitive decision, but as the binary execution outcome induced when the corresponding routing signal is converted into a single admissible seller assignment so as to preserve the no-split fulfillment rule. Inventories are also bounded by storage capacities through $0 \le I^j_{tp} \le U^j_p$ for all relevant nodes $j \in \mathcal{S} \cup \{f\}$. Finally, transport and operational limits on online shipments and inter-store transshipments are summarized in Equation 5:

$$0 \le x^{o,f}_{tpz} \le \overline{M}^f_z, \quad 0 \le x^{o,s}_{tpz} \le \overline{M}^{s(z)}_z, \quad 0 \le \tau^{s \to s'}_{tp} \le \overline{\tau}^{s \to s'}_p. \tag{5}$$

 doi:10.20944/preprints202603.1244.v1

System dynamics are induced by these controls. Inventories evolve according to lead-time arrivals, replenishment injections, transshipment activity, and fulfillment outflows. This is captured in Equation 6:

$$I^j_{(t+1)p} = I^j_{tp} + A^j_{tp}(u_t) - \mathrm{Out}^j_{tp}(u_t) \qquad \forall (t,p,j). \tag{6}$$

Accordingly, the transition kernel of the MDP is induced jointly by Equation 1, Equation 3, Equation 4, Equation 5, and Equation 6.

Under the above dynamics and feasibility conditions, the objective is to maximize expected total profit over $\mathcal{T}$. Let ρ_X denote the unit revenue for channel X, $c^{o,s}$ and $c^{o,f}$ the online-fulfillment costs from store and FC, c^{tr} the transshipment cost, h_j the holding cost at node j, and π_X the lost-sales penalty for channel X. The resulting per-period profit contribution is defined in Equation 7:

$$\Pi_t(u_t) = \sum_{p,z,X} \rho_X \, x^X_{tpz} - \sum_{p,z} \left(c^{o,s} x^{o,s}_{tpz} + c^{o,f} x^{o,f}_{tpz}\right) - \sum_{p,s\neq s'} c^{\mathrm{tr}} \tau^{s\to s'}_{tp} - \sum_{p,j} h_j I^j_{tp} - \sum_{p,z,X} \pi_X \, x^{X,\mathrm{los}}_{tpz}. \tag{7}$$

The constrained operations problem can therefore be written as in Equation 8:

$$\max_{u_{1:T}} \mathbb{E}\left[\sum_{t\in\mathcal{T}} \Pi_t(u_t)\right] \quad \text{s.t. Equation 3, Equation 4, Equation 5, Equation 6.} \tag{8}$$

In this form, profit maximization remains inherently coupled with lost-sales minimization, because the feasibility restrictions limit service decisions while unmet demand is absorbed by the lost-sales terms in Equation 3 and penalized directly in Equation 7.

We align the above formulation with an MDP $\mathcal{M} = (\mathbb{S}, \mathbb{A}, P, R)$ over the same horizon. The reward is defined as an affine transformation of the profit contribution in Equation 7, namely Equation 9:

$$r_t = R(s_t, a_t) = \Pi_t(u_t) + \kappa. \tag{9}$$

Hence, the RL objective is to maximize the expected return, that is, $\max_\pi J(\pi) = \mathbb{E}_\pi[\sum_{t\in\mathcal{T}} r_t]$, while its alignment with the original profit-maximization objective follows from $J(\pi) = \mathbb{E}[\sum_{t\in\mathcal{T}} \Pi_t(u_t)] + |\mathcal{T}|\kappa$. In this regard, the rewarding mechanism associated with each state transition implements the same profit-driven criterion as the constrained optimization problem in Equation 8, while the feasibility structure induced by Equation 3, Equation 4, Equation 5, and Equation 6 ensures that profit maximization remains directly linked to lost-sales minimization under capacity limitations.

4. The Proposed HRL Framework: Methods, Implementation Techniques and Benchmarking Protocol

4.1. Details on the State and Action Spaces

Following the analysis presented in the previous section regarding the developing of the MDP relevant to the environment dynamics, we now make explicit the state–action tuples used in the implementation and, in particular, the goal signal through which the manager conditions the worker in the HRL variant. Let $t \in \mathcal{T}$ denote the primitive decision periods and let $k \in \{1, \ldots, T^{\mathrm{cyc}}\}$ denote replenishment cycles of fixed length L (weekly in our implementation), where cycle k corresponds to the set of primitive periods

$$\mathcal{T}_k = \{(k-1)L + 1, \ldots, kL\}. \tag{10}$$

Under a flat controller, the environment state at time t is $s_t \in \mathbb{S}$ and includes on-hand inventories, pipeline inventories induced by lead times, demand-forecast features, and simple time features, namely

$$s_t = \left(I^f_{tp},\ (I^s_{tp})_{s\in\mathcal{S}},\ (\mathcal{A}^{f,\ell}_{tp})^{\ell_f}_{\ell=1},\ (\mathcal{A}^{s,\ell}_{tps})^{\ell_s}_{\ell=1},\ (F^X_{tps})_{X\in\mathcal{C},s\in\mathcal{S}},\ \eta_t\right)_{p\in\mathcal{P}}. \tag{11}$$

The flat action $a_t \in \mathbb{A}$ is a continuous vector that parameterizes the operational controls u_t in Equation 2 by inducing (i) store target positions $\widehat{I}^s_{tp}$, which the environment uses to execute feasible lateral

 doi:10.20944/preprints202603.1244.v1

transshipments, and (ii) online routing fractions $\alpha^s_{tp} \in [0,1]$ (store share of online demand); additionally, at replenishment epochs $t \in \mathcal{T}^{\text{cyc}} \subset \mathcal{T}$, the action also induces the cycle-level replenishment/allocation quantities (y_{tpf}, y_{tps}). In implementation terms, the flat action vector is normalized on $[0,1]$ and decoded component-wise: target-position coordinates are scaled to store capacity, online-routing coordinates are interpreted directly as bounded store-fulfillment shares, supplier-order coordinates are scaled to the weekly supplier cap, and FC-to-store shipment coordinates are scaled to the weekly FC shipment cap. This mapping is summarized as

$$a_t \longmapsto \left(\widehat{I}^s_{tp},\ \alpha^s_{tp},\ y_{tpf},\ y_{tps}\right)_{s\in\mathcal{S},\,p\in\mathcal{P}} \equiv u_t, \tag{12}$$

with (y_{tpf}, y_{tps}) active only when $t \in \mathcal{T}^{\text{cyc}}$. More specifically, the induced store targets are obtained by scaling normalized action coordinates to store capacity, $\widehat{I}^s_{tp} \in [0, C_s]$, whereas online-routing coordinates remain in $[0,1]$. At replenishment epochs, supplier-order requests are scaled to the weekly cap $Q^{\max}_{sup} = 250$ units per product, and FC-to-store shipment requests are scaled to the weekly cap $Q^{\max}_{fc\to s} = 120$ units per product and store. State transitions are induced by the inventory dynamics in Equation 6 together with stochastic demand estimated by applying Equation 1, and the reward is profit-centric as in Equation 9.

Under the hierarchical realization, we define a manager operating on the cycle index k and a goal-conditioned worker operating on the primitive index t. The manager observes at the beginning of cycle k a state $s^{(m)}_k \in \mathbb{S}^{(m)}$ defined as the environment snapshot at the first primitive period of the cycle,

$$s^{(m)}_k = s_{(k-1)L+1}, \tag{13}$$

and selects a cycle action that consists of (i) replenishment/allocation decisions and (ii) a goal signal for the worker. In the implementation, the goal signal is the pair of store targets and transfer budgets,

$$g_k = \left(\widehat{I}^s_{kp},\ B^s_{kp}\right)_{s\in\mathcal{S},\,p\in\mathcal{P}} \in \mathbb{G}, \tag{14}$$

and the manager action can be written compactly as

$$a^{(m)}_k = \left(y_{kpf},\ y_{kps},\ g_k\right) \in \mathbb{A}^{(m)}. \tag{15}$$

Given g_k, the worker observes an augmented state and chooses primitive actions throughout $\mathcal{T}_k$. Specifically, for each $t \in \mathcal{T}_k$ the worker state is

$$s^{(w)}_t = (s_t,\ g_k) \in \mathbb{S}^{(w)} := \mathbb{S} \times \mathbb{G}, \tag{16}$$

and the worker action $a^{(w)}_t \in \mathbb{A}^{(w)}$ parameterizes the per-period components of u_t by selecting online routing fractions α^s_{tp} and by driving the system towards the target positions $\widehat{I}^s_{kp}$ using feasible transshipments subject to the budgets B^s_{kp} and the capacity constraints (cf. Equation 4–Equation 5). In implementation terms, manager targets are again scaled to store capacity, while transfer-budget coordinates are scaled to the weekly transshipment allowance $B^s_{kp} \in [0, 6\,\overline{\tau}_p]$, where $\overline{\tau}_p = 8$ units per product and day in the experiments, yielding a weekly limit of 48 units per product and store. The manager receives the cycle return defined as the sum of primitive rewards,

$$R^{(m)}_k = \sum_{t\in\mathcal{T}_k} r_t, \tag{17}$$

where r_t is given by Equation 9. Hence, both the flat policy and the hierarchical pair optimize the same profit-driven objective (equivalently coupling profit maximization with lost-sales minimization via Equation 3), while differing only in temporal abstraction and in the explicit goal-conditioning mechanism g_k in Equation 14 that mediates manager–worker coordination.

 doi:10.20944/preprints202603.1244.v1

Regarding the implementation of both actors and critics employed in our HRL approach, we note that they are built on feed-forward NNs. In the case of the actors, action generation is based on a Beta policy, so that each action component is modeled on the bounded interval $[0, 1]$. Concretely, for each action coordinate i, the actor outputs two strictly positive shape parameters (α_i, β_i) through separate output heads followed by a Softplus transformation and a positive offset (in our case, $+1$), and the corresponding action component is sampled as $a_i \sim \text{Beta}(\alpha_i, \beta_i)$. During deterministic evaluation, the mean action $a_i = \alpha_i/(\alpha_i + \beta_i)$ is used. This choice is appropriate in our setting because the action space is continuous and normalized, so the support of the Beta distribution is directly aligned with the support of the control variables, unlike a Gaussian policy that would require additional squashing or clipping. Importantly, these action components do not directly execute business decisions in raw form; rather, they provide normalized control signals that are decoded by the simulator into feasible realized controls under the business rules introduced in Section 3. For instance, routing-related outputs parameterize bounded online-fulfillment shares, whereas replenishment-related outputs parameterize shipment or allocation requests that are subsequently translated into batch-feasible, capacity-feasible, and lead-time-consistent quantities. Hence, the Beta policy is used to parameterize a constrained decision interface rather than to imply that all business actions are intrinsically continuous at execution level.

It is also worth noting that, for safeguarding the tractability of the learning problem and aligning the control logic with the underlying business context, a state-dependent pruning mechanism is imposed on the raw action space. In particular, although the original action space formally contains all admissible control coordinates, several of them become operationally irrelevant at specific decision points, e.g., replenishment-related components outside cycle epochs or flow-allocation components rendered inactive by zero inventory, exhausted capacity, or lead-time restrictions. To account for this, we define a binary relevance mask $m(s) \in \{0, 1\}^{\dim(\mathbb{A})}$ as a deterministic function of the current state, and the corresponding pruned action set as

$$\mathbb{A}^{\text{Pr}}(s) = \{a \in \mathbb{A} : a_i = 0 \text{ whenever } m_i(s) = 0\}, \tag{18}$$

while the effective action applied by the controller is

$$\tilde{a}(s) = m(s) \odot a. \tag{19}$$

Hence, coordinates with $m_i(s) = 0$ remain part of the formal action representation but are treated as irrelevant for optimization, since they cannot induce meaningful state transitions under the prevailing operating conditions. The masked action $\tilde{a}(s)$ is then passed to the simulator, where it is decoded into feasible realized controls. At this stage, inventory feasibility is enforced before the operational transition is finalized, residual infeasibilities are handled through a deterministic feasibility mapping, and routing-related outputs are converted into a single admissible seller assignment so as to preserve the no-split fulfillment rule. This pruning mechanism is relevant both for flat RL and HRL: in the former, it reduces the effective dimensionality of the direct control vector, whereas in the latter it restricts both manager- and worker-level decisions to business-consistent subspaces, thereby mitigating the curse of dimensionality. All experiments and results reported in this study were obtained under this pruned action-space realization, and policy updates were computed with respect to the masked action interface actually exposed to the simulator.

4.2. Architectural Paradigm and Implementation Details

Following the specification of the modeled environment and the corresponding state and action spaces, this section presents the architectural paradigm adopted for the development of the HRL framework. Our approach is policy-based, meaning that at each iteration the policy is estimated directly from trajectory data generated through interaction with the environment. In continuous-control environments, actor-critic and policy-based methods have generally shown more stable optimization

 doi:10.20944/preprints202603.1244.v1

behavior and more favorable empirical convergence characteristics than value-based alternatives that rely on action discretization [40]. From an implementation perspective, multiple alternatives could in principle be considered for developing an effective policy-based solution; however, a growing body of recent work has focused on actor-critic variants because they combine direct policy learning with value-based guidance during training [20]. In the HRL setting, evidence drawn mainly from robotics and recommender systems further suggests that actor-critic formulations are particularly well-suited to hierarchical decision structures, since they support learning across multiple temporal scales while preserving stable policy improvement [13,26]. Aligned with this rationale, our work adopts the hierarchical actor-critic scheme illustrated in Figure 4.

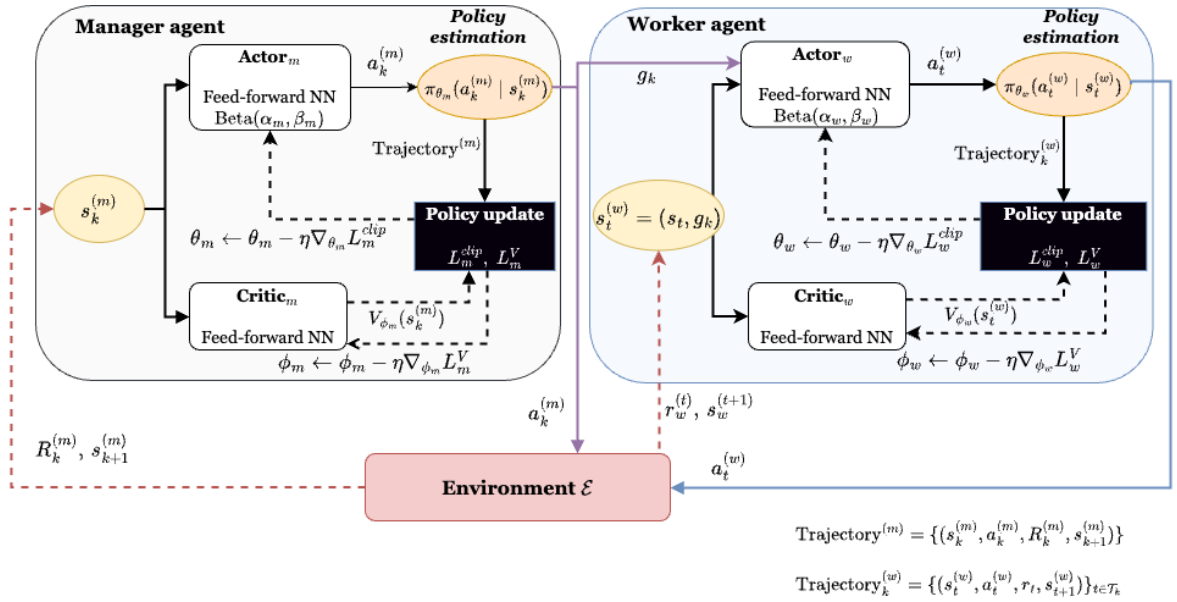

Figure 4. A graphical representation of the implemented HRL-PPO scheme drawn for this study.

The presented approach follows the on-policy paradigm. This means that policy updates are performed using trajectory data generated by the current policy through direct interaction with the environment. At each training iteration, the actor networks estimate the current decision rules at the two hierarchical levels, namely the manager policy $\pi_{\theta_m}(a_k^{(m)} \mid s_k^{(m)})$ and the worker policy $\pi_{\theta_w}(a_t^{(w)} \mid s_t^{(w)})$. On the basis of these policies, trajectories are sampled from the environment and subsequently used to compute the objective functions that guide the update of both the actor and critic parameters.

Within this scheme, the critic provides value estimates that are used to construct the advantage signal, while the actor is updated through the PPO objective. In particular, the value loss is defined as $L^V(\phi) = \mathbb{E}_t[(V_\phi(s_t) - R_t)^2]$, where R_t denotes the return target. Accordingly, this quantity measures the discrepancy between the critic prediction and the return induced by the sampled trajectory, and therefore determines the critic-side learning signal. The actor-side learning signal is instead based on the estimated advantage, given in Equation 20, where the temporal-difference residual is defined as $\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$.

$$\hat{A}_t = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l \, \delta_{t+l}. \tag{20}$$

According to Equation 20, the advantage estimate captures whether the sampled action performed better or worse than expected relative to the critic baseline, and is therefore the quantity through which the direction of policy improvement is determined.

The probability ratio is introduced in order to compare the policy currently being optimized against the policy that generated the trajectory data. More precisely, for each sampled state–action

 doi:10.20944/preprints202603.1244.v1

pair (s_t, a_t), the ratio is defined as $\rho_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta^{\text{old}}}(a_t|s_t)}$. In this operator, the numerator corresponds to the probability assigned to the sampled action by the updated policy, whereas the denominator corresponds to the probability assigned to the same action by the previous policy under which the trajectory was collected. Hence, the ratio quantifies the relative change in action likelihood induced by the policy update.

The interpretation of this ratio is immediate. When $\rho_t(\theta) = 1$, the updated and previous policies assign exactly the same probability to action a_t under state s_t. When $\rho_t(\theta) > 1$, the updated policy assigns greater probability mass to that action, whereas when $\rho_t(\theta) < 1$, the updated policy assigns lower probability mass. Therefore, $\rho_t(\theta)$ provides a local measure of how strongly the policy shifts on the basis of the same experience sample. This ratio is then combined with the estimated advantage $\hat{A}_t$, so that the direction of policy improvement depends on whether the sampled action proved better or worse than expected. In particular, the unclipped surrogate term is written as $\rho_t(\theta)\hat{A}_t$.

Based on this specification, if $\hat{A}_t > 0$, the optimization encourages an increase in the probability assigned to the sampled action, whereas if $\hat{A}_t < 0$, it encourages a decrease. Nevertheless, updating the policy solely on the basis of the probability ratio may result in overly large policy shifts, since substantial deviations between π_θ and $\pi_{\theta^{\text{old}}}$ could still be favored whenever they appear to improve the objective. To address this issue, PPO introduces a clipping operator that constrains the ratio within a bounded neighborhood around unity, namely $\text{clip}(\rho_t(\theta), 1-\epsilon, 1+\epsilon)$, where $\epsilon > 0$ denotes the clipping threshold. This mechanism is intended to prevent the updated policy from moving excessively far from the previous one during a single optimization step, thereby reducing instability and limiting the noise that may arise during policy estimation. In line with this rationale, the final optimization target is given in Equation 21.

$$L^{\text{clip}}(\theta) = \mathbb{E}_t\left[\min\left(\rho_t(\theta)\hat{A}_t,\ \text{clip}\left(\rho_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_t\right)\right]. \tag{21}$$

As shown in Equation 21, the probability ratio becomes the core mechanism through which PPO regulates the scale of policy updates and supports stable policy improvement. These two losses are linked directly to parameter renewal through gradient-based optimization. The actor parameters are updated according to Equation 22, whereas the critic parameters are updated according to Equation 23. The same logic applies at both hierarchical levels, that is, for the manager-level pair (θ_m, ϕ_m) and for the worker-level pair (θ_w, ϕ_w).

$$\theta \leftarrow \theta - \eta \nabla_\theta L^{\text{clip}}(\theta), \tag{22}$$

$$\phi \leftarrow \phi - \eta \nabla_\phi L^{V}(\phi). \tag{23}$$

Regarding the implementation of both actors and critics employed in our HRL approach, we note that they are built on feed-forward NNs. In the case of the actors, action generation is based on a Beta policy, so that each action component is modeled through a Beta-distributed random variable on the bounded interval $[0, 1]$. This choice is particularly appropriate in our setting because the action space is continuous and normalized, and therefore the support of the Beta distribution is directly aligned with the support of the control variables. An alternative would be to employ a Gaussian policy, whose support extends over $(-\infty, +\infty)$; however, such a choice would require an additional squashing or clipping mechanism in order to enforce bounded actions, whereas the Beta formulation provides a direct bounded representation. Importantly, these action components do not directly execute business decisions in raw form. Rather, they provide normalized control signals that are decoded by the simulator into feasible realized controls under the business rules introduced in Section 3. For instance, routing-related outputs parameterize bounded online-fulfillment shares, which are then mapped to a single admissible seller so as to enforce the no-split fulfillment rule, whereas replenishment-related outputs parameterize shipment or allocation requests that are subsequently translated into batch-feasible, capacity-feasible, and lead-time-consistent quantities. Hence, the Beta policy is used

 doi:10.20944/preprints202603.1244.v1

to parameterize a constrained decision interface rather than to imply that all business actions are intrinsically continuous at execution level. After experimentation, the hyper-parameter configuration retained for the implementation is reported in Table 1; the same configuration was used for both the flat RL benchmark and the HRL scheme.

Table 1. Hyper-parameter configuration used for the development of the HRL-PPO scheme.

Hyper-parameter	**Value**
Learning rate	2×10^{-4}
Discount factor γ	0.99
GAE parameter λ	0.95
PPO clipping parameter ϵ	0.20
Value loss coefficient	0.50
Entropy coefficient	0.001
Training epochs per update	4
Batch size	256
Number of hidden layers	2
Roll-out exposure (days)	2048
Hidden-layer architecture	$(64, 64)$
Activation function	tanh

5. Experimental Evaluation

This section presents the experimental protocol adopted in the study and the corresponding numerical results. In this regard, it aims at illustrating the potential of the proposed HRL–PPO framework to support the resilience of omnichannel SCs under varying demand patterns.

5.1. Benchmarking Protocol

The evaluation protocol designed for this study is threefold. First, we examine whether the proposed HRL formulation offers advantages over a standard PPO-based RL scheme under the same simulation environment and demand-generation setting. Second, we benchmark the proposed scheme against two business-relevant heuristics, namely a base-stock/order-up-to rule and a greedy fulfillment/re-balancing rule, both of which reflect simple yet operationally meaningful inventory management policies for stock positioning and inventory allocation. In both the heuristic and learning-based settings, future demand is not directly observed, but instead estimated through the forecasting component embedded in the simulator, implemented via simple exponential smoothing ($F_{t+1} = \alpha D_t + (1-\alpha)F_t$). This choice was made due to its favorable trade-off between forecasting accuracy, robustness, and implementation simplicity, which explains its longstanding use as a practical benchmark in the forecasting literature [6,16]. Moreover, the smoothing parameter α was calibrated rather than fixed a-priori. Specifically, five candidate values, $\alpha \in \{0.2, 0.4, 0.5, 0.6, 0.7\}$, were evaluated separately for each demand profile using one-step-ahead RMSE (Root Mean Squared Error). The best result for the blend-demand profile was obtained at $\alpha = 0.4$ with RMSE equal to 6.124, whereas the Merton-only profile performed best at $\alpha = 0.6$ with RMSE equal to 9.126; the remaining α values deviated by up to approximately 20% from the best-performing specification in each case. This differentiation is also consistent with the demand structures considered, since the blend profile favors a more moderate smoothing weight, whereas the more shock-prone Merton-only profile is better served by a more reactive update parameter, in line with the broader literature on smoothing operators and erratic demand behavior [42].

As a last step, we evaluate the performance gap between the obtained policies and a perfect-information benchmark constructed under the same simulator rules. Specifically, this benchmark is implemented as a policy that has direct access to the realized demand tape and uses this information to determine weekly replenishment and FC-to-store shipment decisions, as well as daily transshipment targets and online-fulfillment splits. Accordingly, unlike the heuristic and learning-based approaches, this benchmark does not rely on the forecasting mechanism when allocating inventory across the

 doi:10.20944/preprints202603.1244.v1

network. Strictly speaking, this benchmark should not be interpreted as a mathematically optimal upper bound, but rather as a strong reference policy under privileged demand information. Although this setting constitutes an over-simplification of real operating conditions, since future demand is rarely known with certainty in practice, it nevertheless provides, in line with [23], a strong reference point for evaluating the performance of the proposed HRL–PPO scheme under perfect demand information. An analysis regarding the exact encoding of the two heuristics and the perfect-information benchmark is provided in Appendix A.

To ensure that the comparison between the proposed HRL–PPO scheme and the flat PPO benchmark remains methodologically fair, both learning-based controllers are assessed under the same simulator, reward basis, scenario family, and training–testing protocol, while also being trained under the same overall interaction budget and evaluated over the same horizon. Moreover, the same operational feasibility and action-filtering rules are enforced in both cases. Hence, the comparison is intended to isolate differences in control organization rather than differences in environmental assumptions or privileged information access. More specifically, the flat PPO policy acts directly on the available system state, whereas the HRL–PPO controller introduces temporal decomposition through manager-level coordination and worker-level execution. The additional coordination signals used within the hierarchical scheme are generated internally from the same decision context and should therefore be interpreted as part of the architecture itself, rather than as an external informational enhancement.

The evaluation protocol is implemented across three distinct business scenarios. This design choice is intentional, as it allows us to examine how alternative capacity configurations across the nodes of the omnichannel network influence overall system performance and resilience to demand shocks. Table 2 summarizes the capacity restrictions specified for each scenario. All three scenarios are developed under the premise that inventory positioning plays a compensatory role within the network, since greater inventory concentration at one echelon may partially offset tighter capacity constraints or longer lead times at another [12,22]. Scenario 1 serves as the baseline configuration, reflecting a relatively balanced capacity allocation between the FC and store echelons. Scenario 2 represents a more upstream-oriented operating structure, in which the store-side inventory position is weakened relative to the FC (e.g., $C_s = \frac{2}{3}C_s^{(1)}$). By contrast, Scenario 3 reflects a more downstream-oriented arrangement, in which inventory and order-fulfillment capacities are shifted closer to demand points (e.g., $C_s = \frac{4}{3}C_s^{(1)}$ and $C^{so} = \frac{6}{5}C^{so,(1)}$), while the FC-to-store replenishment link becomes slower. This setting approximates a more locally responsive operating scheme, in which store-level autonomy is strengthened, particularly because stores may also function as intermediate delivery points under the inter-seller structure incorporated in our model.

As Table 2 illustrates, all business scenarios were evaluated assuming multiple stores at the last echelon, with the number of stores varying from 2 to 30, while the product assortment was fixed at 6 items. Given that our work is oriented towards analyzing the impact of the developed HRL-PPO on safeguarding the resilience of omnichannel networks, the experimental protocol is applied to two different demand types.

 doi:10.20944/preprints202603.1244.v1

Table 2. Details on the capacity factors used in each of the studied business scenarios.

Field	Scenario 1	Scenario 2	Scenario 3
Evaluated scale			
Stores S (evaluated)	$\{2, 7, 12, 18, 24, 30\}$	$\{2, 7, 12, 18, 24, 30\}$	$\{2, 7, 12, 18, 24, 30\}$
Products P	6	6	6
Capacities & lead times (explicit per experiment)			
Store cap/product C_s	60	40	80
FC cap/product C_f	300	320	320
Transship cap/day/product C^{tr}	8	6	10
FC→store ship cap/day/product C^{fc}	60	70	50
Store→online ship cap/day/product C^{so}	30	30	36
Supplier→FC lead time L^{sup} (days)	2	1	2
FC→store lead time L^{fc} (days)	1	1	2
Store→store lead time L^{tr} (days)	0	2	1
Max supplier order/week $Q^{sup}_{\max}$	250	200	300
Max FC→store ship/week $Q^{fc\to s}_{\max}$	120	80	100

Table 3 specifies the product-level demand configurations and the corresponding parameter values used to represent heterogeneous demand behavior across the assortment, and these configurations are examined under all business scenarios. More specifically, the first configuration adopts a mixed demand structure, in which three products follow a uniform demand pattern and the remaining three are modeled through Merton-type shocks, whereas the second assumes a fully shock-driven setting in which all six products are subject to Merton-type demand behavior. The adopted calibration is a deliberate choice intended to reflect the expected cross-channel structure of omnichannel demand, namely a stronger baseline for walk-in demand, a more limited click-and-collect stream, and a relatively more shock-prone online channel; this is consistent with the literature showing that disruptive events tend to induce sharper reallocations toward digital channels while store traffic often remains the dominant reference flow in retail systems [32]. For facilitating the reproducibility of our work, we also note that each instance was evaluated over a horizon of $T = 48$ daily periods, corresponding to 8 selling weeks of 6 days each, with replenishment decisions activated every 6 days. The flat PPO benchmark was trained for 5000 episodes per instance, while the hierarchical scheme used 1500 worker warm-up episodes, 2500 worker full-training episodes, and 5000 manager episodes. Training was stopped at 5000 episodes because both the smoothed training-return trajectories and the fixed-seed evaluation reward curves were observed to stabilize by that point, indicating convergence without a meaningful gain from longer runs. A fixed seed-pool protocol was adopted, using 42 training seeds and 10 disjoint evaluation seeds; no separate validation split or early stopping rule was employed, and all experiments were implemented in PyTorch and Gymnasium, with execution on GPU when CUDA was available and otherwise on CPU.

 doi:10.20944/preprints202603.1244.v1

Table 3. Experimental design: demand parameter values considered across the three business scenarios.

Demand parameters	Values
Blend of uniform and Merton-type demand shocks	
Uniform products $\mathcal{P}_U$	$\{0, 1, 2\}$
Jump products $\mathcal{P}_J$	$\{3, 4, 5\}$
Uniform ranges (w/cc/onl)	$[2, 6]/[0, 2]/[1, 5]$
Jump params walk-in $(\mu, \sigma, \lambda, m, \sigma_j)$	$(3.0, 1.1, 0.22, 6.5, 1.8)$
Jump params cc $(\mu, \sigma, \lambda, m, \sigma_j)$	$(1.0, 0.7, 0.10, 3.2, 1.1)$
Jump params online $(\mu, \sigma, \lambda, m, \sigma_j)$	$(2.0, 1.0, 0.30, 7.5, 2.0)$
All products facing Merton-type shocks in different timings	
Jump products $\mathcal{P}_J$	$\{0, 1, 2, 3, 4, 5\}$
Jump params walk-in $(\mu, \sigma, \lambda, m, \sigma_j)$	$(3.0, 1.1, 0.22, 6.5, 1.8)$
Jump params cc $(\mu, \sigma, \lambda, m, \sigma_j)$	$(1.0, 0.7, 0.10, 3.2, 1.1)$
Jump params online $(\mu, \sigma, \lambda, m, \sigma_j)$	$(2.0, 1.0, 0.30, 7.5, 2.0)$

5.2. Results

This subsection reports the results obtained from the three-fold evaluation protocol described above. It begins with a comparative analysis of the objective function, namely profit maximization, between the baseline PPO and the proposed HRL-PPO framework. This comparison is conducted under both demand configurations considered in the experimental design, namely the mixed setting with uniform and shock-affected products and the fully shock-driven setting. The second level of analysis benchmarks the proposed approach against the perfect-information oracle and the selected problem-specific heuristics. This comparison is performed across all instances by synthesizing the reward into its main cost- and service-related dimensions.

5.2.1. Blend of Uniform and Merton-Type Demands Under Different Operating Scenarios

Following the specifications of Table 3 regarding the mixture of demand patterns across products, this subsection comparatively assesses the progress achieved towards maximizing the overall objective function of the studied problem. For illustration purposes, we refer to Figure 5, which presents the rewarding obtained after 5,000 training episodes for the first business scenario analyzed in this study. Since the reward is defined as an affine transformation of the underlying profit-based objective, it serves as a direct proxy for the convergence of the learned policy and, therefore, as an estimate of how effectively each method improves system-level decision-making over time. To enhance interpretability, the results are reported in moving-average form. Specifically, we average performance over every 60 consecutive episodes and across the multiple training seeds used (i.e., 42) during the learning phase, so as to attenuate the noise induced by random initialization, stochastic demand realizations, and exploration effects. This presentation practice is standardized in the RL literature, as it facilitates a more stable view of convergence behavior and a more reliable assessment of robustness and generalization [34]. The corresponding rewarding trajectories for the remaining two business scenarios followed a highly similar pattern and are therefore omitted for reasons of concise presentation, without affecting the interpretation of the convergence behavior discussed in this subsection.

Based on Figure 5, several conclusions can be drawn regarding the behavior and the convergence level of the two approaches compared. Specifically, HRL-PPO seems to converge to a consistently higher reward level than the flat RL-PPO benchmark, in all the cases analyzed. Also, in most of the cases, the progressive rewarding presents weaker oscillation around its mean value and persistent drops once training passes the initial adaptation stage, which could be regarded as illustrative evidence of stability. For a cleaner comparison, the post–warm-up phase is the most informative. This phase can be identified as the point after which the reward curves begin to stabilize and display a clearer upward direction. In our experiments, this appears to occur after approximately 1,500 episodes in most cases. In addition, for several store-scale settings, HRL-PPO starts from, or very quickly reaches, a clearly higher reward region. This suggests that the hierarchical structure provides a better timing of decisions from the early stages of learning. This finding reflects the stronger capacity of the hierarchical scheme

 doi:10.20944/preprints202603.1244.v1

to coordinate cost- and service-related decisions in a temporally consistent manner, thereby preserving the network's resilience under stochastic demand conditions.

Following the above analysis, and by decomposing the rewarding into its elements, several dimensions regarding the problem solution can be drawn. Given that our research is oriented towards assessing the capacity of the HRL-PPO scheme to converge to solutions that yield minimized lost sales while also maintaining operationally meaningful inventory behavior, Table 4 reports the results obtained regarding the holding cost, lost sales rate, and inter-seller node transshipments. The latter is particularly relevant ot our modeling scheme since it reflects the extent to which the policy exploits lateral inventory re-balancing across sellers in support of omnichannel demand fulfillment; an extension brought by this study to the existing body of research. The results reported in the table correspond to mean values and standard deviations over the last 500 episodes, for safeguarding the convergence of the RL schemes. For the two heuristic benchmarks, namely the base-stock/order-up-to rule and the greedy fulfillment/re-balancing rule, the evaluation was conducted under the same simulation environment and demand-generation setting, based on repeated independent simulation runs under identical experimental conditions, and was terminated once the relative improvement in the running mean objective value between two successive batches of runs, i.e., $\Delta^{(k)} = \frac{\bar{J}^{(k-1)} - \bar{J}^{(k)}}{\bar{J}^{(k-1)}} \times 100$, fell below 2%, indicating that further runs did not lead to materially different results.

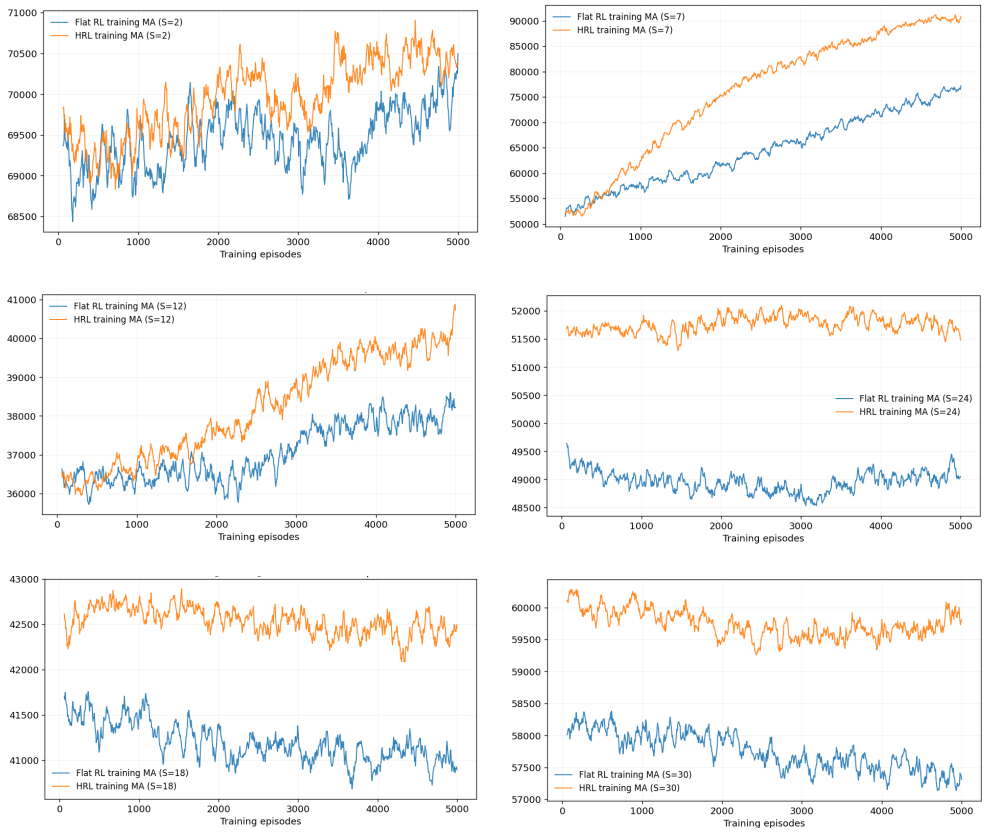

Figure 5. Results for the reward progression in Scenario 1 under the first demand configuration (half of the products face Merton-type demand).

The results in Table 4 suggest that the proposed HRL-PPO framework achieves the most balanced optimization across the examined performance dimensions, yielding, on average, a reduction in lost sales of about 11.2% relative to PPO, 23.5% relative to the base-stock/order-up-to rule, and 22.4% relative to the greedy fulfillment/re-balancing rule. If the most resilient network under demand disturbances is the one with the lowest unmet demand, then lost sales provide a direct reading of resilience. Under this rationale, the best value reached by HRL-PPO is 0.0864, observed in Scenario 2 at the smallest store scale. More broadly, Scenario 2 remains the best-performing setting, with

 doi:10.20944/preprints202603.1244.v1

mean HRL-PPO lost sales about 18.6% lower than Scenario 1 and 25.5% lower than Scenario 3. The results also show a clear scale effect. As the number of stores increases, lost sales rise in all cases, by about 66.7% in Scenario 1, 62.5% in Scenario 2, and 54.5% in Scenario 3 from the smallest to the largest network. This indicates that larger networks create a stronger coordination burden, even under hierarchical control. At the same time, the improvement in resilience is not achieved at the expense of inventory efficiency, since HRL-PPO also reduces holding costs by about 8.4% relative to PPO, 17.8% relative to the base-stock/order-up-to rule, and 17.6% relative to the greedy fulfillment/re-balancing rule. This important aspect showcases that the framework does not simply protect service levels through excessive stock accumulation, but rather through better coordination of inventory positioning and replenishment decisions. The findings therefore suggest that the most influential capacity elements for resilience are those that support timely replenishment propagation across the network, while local storage and lateral transfers act as a complementary factor affecting lost-sales minimization.

doi:10.20944/preprints202603.1244.v1

Table 4. Comparative results for the first demand configuration, across the business scenarios studied.

Scenario	Stores S	Base-stock / order-up-to rule			Greedy fulfillment / re-balancing rule			PPO			HRL-PPO		
		Holding cost	Inter-seller node transshipment	Lost sales rate	Holding cost	Inter-seller node transshipment	Lost sales rate	Holding cost	Inter-seller node transshipment	Lost sales rate	Holding cost	Inter-seller node transshipment	Lost sales rate
Scenario 1	2	214.60 ± 6.89	78.42 ± 4.08	0.131 ± 0.009	201.73 ± 6.55	83.11 ± 4.49	0.126 ± 0.009	190.84 ± 5.12	73.26 ± 3.05	0.115 ± 0.007	181.42 ± 4.37	68.95 ± 2.62	0.1026 ± 0.0062
	7	495.18 ± 15.85	241.86 ± 12.34	0.145 ± 0.010	523.92 ± 17.29	227.34 ± 12.28	0.149 ± 0.010	463.01 ± 12.27	208.90 ± 8.98	0.127 ± 0.008	434.22 ± 10.42	191.54 ± 7.47	0.1140 ± 0.0071
	12	792.44 ± 25.36	488.73 ± 25.90	0.170 ± 0.011	746.36 ± 24.26	518.06 ± 28.49	0.164 ± 0.011	703.82 ± 18.30	452.11 ± 19.44	0.151 ± 0.009	654.95 ± 15.72	412.77 ± 15.68	0.1368 ± 0.0080
	18	1154.80 ± 37.53	744.61 ± 40.20	0.174 ± 0.012	1221.38 ± 40.31	699.93 ± 39.20	0.171 ± 0.012	1073.66 ± 28.99	648.74 ± 28.54	0.154 ± 0.010	996.15 ± 24.41	594.27 ± 23.77	0.1368 ± 0.0081
	24	1527.92 ± 50.42	1012.58 ± 56.70	0.188 ± 0.013	1439.30 ± 48.94	1073.33 ± 61.18	0.183 ± 0.013	1353.03 ± 37.21	934.02 ± 42.03	0.167 ± 0.011	1241.08 ± 31.65	848.06 ± 33.92	0.1482 ± 0.0090
	30	1908.55 ± 64.89	1287.90 ± 73.41	0.214 ± 0.015	2017.70 ± 70.62	1210.63 ± 70.22	0.218 ± 0.015	1770.02 ± 49.56	1113.78 ± 50.12	0.192 ± 0.012	1618.79 ± 40.47	1000.77 ± 38.03	0.1710 ± 0.0102
Scenario 2	2	228.74 ± 7.43	91.35 ± 4.84	0.118 ± 0.008	242.12 ± 8.11	85.87 ± 4.72	0.122 ± 0.008	213.56 ± 5.87	78.64 ± 3.38	0.099 ± 0.006	198.33 ± 4.96	72.41 ± 2.90	0.0864 ± 0.0054
	7	536.92 ± 17.72	280.44 ± 14.86	0.131 ± 0.009	505.78 ± 17.20	297.27 ± 16.35	0.128 ± 0.009	470.39 ± 12.94	259.48 ± 11.16	0.110 ± 0.007	431.52 ± 10.57	234.96 ± 9.16	0.0972 ± 0.0062
	12	860.15 ± 28.39	565.70 ± 31.11	0.146 ± 0.010	911.21 ± 30.98	531.76 ± 30.31	0.149 ± 0.010	803.58 ± 22.10	485.34 ± 21.84	0.123 ± 0.008	734.87 ± 18.37	437.59 ± 17.94	0.1080 ± 0.0070
	18	1254.94 ± 42.67	861.36 ± 47.38	0.150 ± 0.011	1180.62 ± 41.32	912.12 ± 51.98	0.147 ± 0.011	1098.23 ± 30.75	777.84 ± 35.00	0.126 ± 0.009	1003.53 ± 25.09	702.12 ± 28.08	0.1080 ± 0.0071
	24	1658.87 ± 57.23	1169.84 ± 66.68	0.164 ± 0.012	1757.04 ± 62.11	1099.65 ± 64.88	0.160 ± 0.012	1540.13 ± 43.12	1011.68 ± 46.54	0.139 ± 0.010	1397.21 ± 35.63	903.11 ± 36.12	0.1188 ± 0.0081
	30	2075.42 ± 72.64	1486.53 ± 86.20	0.190 ± 0.014	1951.61 ± 70.26	1575.72 ± 94.54	0.186 ± 0.014	1805.24 ± 52.35	1340.76 ± 61.67	0.162 ± 0.011	1638.95 ± 42.61	1195.43 ± 47.82	0.1404 ± 0.0092
Scenario 3	2	245.88 ± 8.24	106.26 ± 5.74	0.147 ± 0.009	231.67 ± 7.99	112.64 ± 6.31	0.151 ± 0.009	215.43 ± 6.25	96.70 ± 4.16	0.128 ± 0.007	199.15 ± 5.18	88.84 ± 3.55	0.1144 ± 0.0063
	7	580.34 ± 19.73	327.18 ± 17.67	0.170 ± 0.010	613.88 ± 21.49	307.55 ± 17.55	0.166 ± 0.010	525.30 ± 14.97	276.80 ± 11.90	0.149 ± 0.008	477.82 ± 12.42	248.14 ± 9.93	0.1352 ± 0.0073
	12	932.66 ± 32.64	659.42 ± 36.93	0.173 ± 0.011	878.42 ± 31.62	699.10 ± 40.55	0.170 ± 0.011	800.11 ± 23.20	598.43 ± 26.33	0.151 ± 0.009	723.70 ± 18.82	536.87 ± 21.47	0.1352 ± 0.0074
	18	1363.54 ± 49.09	1002.87 ± 57.16	0.196 ± 0.012	1442.11 ± 53.36	942.70 ± 55.62	0.201 ± 0.012	1237.63 ± 37.13	848.43 ± 39.87	0.173 ± 0.010	1115.19 ± 29.00	754.16 ± 30.92	0.1560 ± 0.0086
	24	1806.28 ± 66.83	1361.84 ± 80.35	0.208 ± 0.013	1701.49 ± 64.66	1443.55 ± 87.28	0.205 ± 0.013	1541.72 ± 47.79	1218.10 ± 57.25	0.184 ± 0.011	1370.70 ± 37.01	1083.15 ± 44.41	0.1664 ± 0.0097
	30	2260.94 ± 85.92	1730.56 ± 105.56	0.262 ± 0.015	2391.16 ± 93.26	1626.73 ± 101.86	0.227 ± 0.015	2012.24 ± 62.38	1458.35 ± 70.00	0.196 ± 0.012	1787.10 ± 48.25	1285.12 ± 52.69	0.1768 ± 0.0110

 doi:10.20944/preprints202603.1244.v1

Interestingly, the comparative analysis with the perfect-information benchmark showed that the proposed HRL–PPO scheme was able to approach this reference performance rather closely at small network scales, reaching up to approximately 85% of the benchmark value in the two-store case. As the number of stores increased, this proximity gradually declined, indicating that the performance gap widened with network size as coordination complexity became more pronounced; in the largest store configuration, the corresponding ratio dropped to approximately 72%. Nevertheless, the HRL–PPO policy remained consistently competitive across all tested scales, preserving a substantial share of the value attained under privileged demand information. From a computational perspective, the comparison also revealed a measurable time-related gap, with the mean execution-time difference between the perfect-information benchmark and the HRL–PPO scheme amounting to approximately 16% across the examined configurations, which further highlights the practical value of future-demand visibility as a strong informational reference.

5.2.2. Merton-type demands under different operating scenarios

Figure 6 illustrates the rewarding obtained in the case where all products across all channels are subject to demand shocks, based on the settings relevant to the first scenario orchestrated in this study. As a counterpart to the rewarding illustration under the mixed-demand setting, the results in this case suggest that the overall learning behavior remains qualitatively consistent with that observed in the blended-demand environment. In both settings, the two approaches preserve similar convergence tendencies, while the hierarchical formulation continues to exhibit a clearer long-run advantage in terms of robustness and reward formation. This indicates that the transition from a mixed demand structure to a fully jump-driven one does not fundamentally alter the comparative learning profile of the policies, although a more disturbance-sensitive pattern becomes evident in Table 5. More specifically, three differences stand out in the Merton-only case. First, the rewardings exhibit sharper local peaks and more pronounced short-term corrections, particularly at small and medium store scales, which is consistent with the abrupt demand shocks induced by the Merton process. Second, temporary crossovers and brief reversals between Flat RL and HRL appear more frequently than in the blended-demand setting, where the separation between the two curves is generally smoother. Third, the plateau phase is less uniform and presents higher local variability across store configurations, indicating that convergence is still achieved in a broad sense, but under stronger stochastic perturbations and less regular stabilization dynamics.

 doi:10.20944/preprints202603.1244.v1

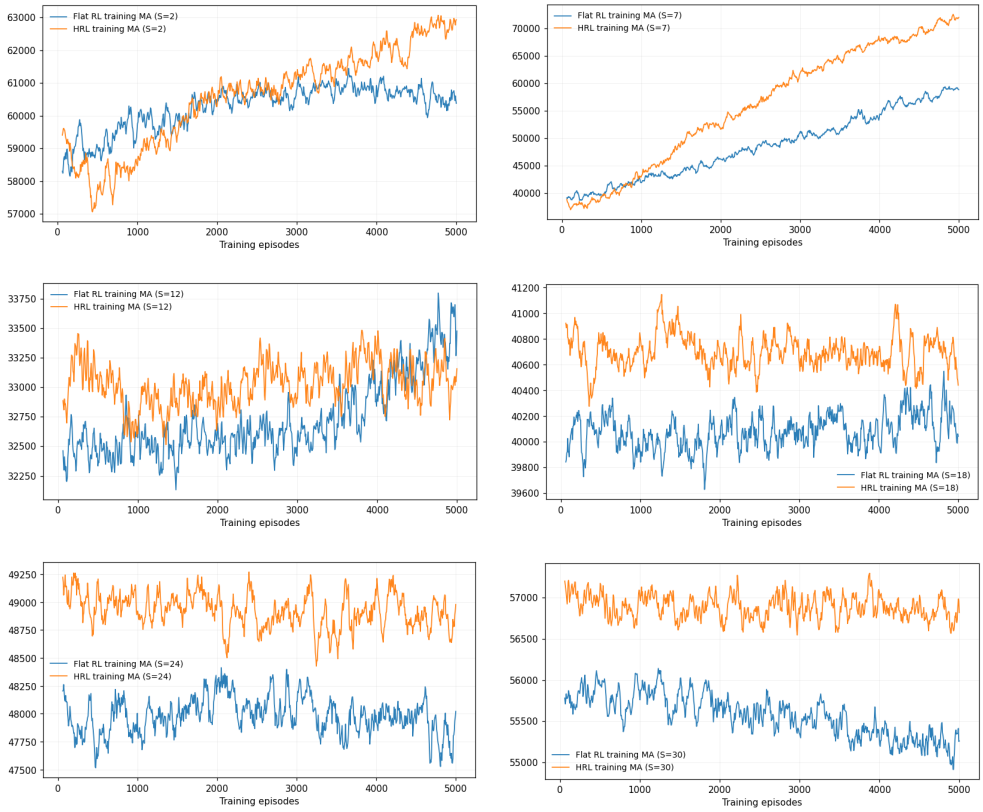

Figure 6. Results for the reward progression in Scenario 1 under the second demand configuration (Merton-only demand profiles).

Based on the results presented in Table 5, several conclusions may be drawn. First, under the fully shock-driven demand setting, the proposed HRL-PPO framework remains effective overall, especially relative to the two simple heuristics, reducing lost sales on average by about 10.3% relative to the base-stock/order-up-to rule and 10.5% relative to the greedy fulfillment/re-balancing rule, while also lowering holding costs by about 17.8% and 18.0%, respectively. At the same time, however, its superiority is no longer uniform when compared with the simple PPO benchmark. More specifically, PPO achieves lower lost-sales rates than HRL-PPO in all store-scale instances of Scenario 1 and in the smaller-scale cases of Scenario 2 (for instance, in Scenario 1 with 2 stores, PPO attains 0.10 versus 0.1640 for HRL-PPO; in Scenario 1 with 7 stores, 0.13 versus 0.1750; and in Scenario 2 with 2 stores, 0.13 versus 0.1380), whereas HRL-PPO regains an advantage from 12 stores onward in Scenario 2 and remains consistently superior throughout Scenario 3. Second, a direct comparison with the blend-demand case shows that the shock effect is substantial, since the mean HRL-PPO lost-sales level increases by about 45.4% under the Merton-only demand setting, indicating that the simultaneous exposure of all products to abrupt disturbances compresses the service-side advantage of the hierarchical framework.

A compact comparative reading of the cost-side metrics points in the same direction. Relative to the first demand configuration, holding costs increase by about 2.0% for the two heuristic policies, 5.0% for PPO, and 3.0% for HRL-PPO, while inter-seller node transshipment rises by about 1.5% for the base-stock policy, 6.0% for the greedy policy and PPO, and 2.0% for HRL-PPO. This pattern is consistent with the demand profile considered here: when all products are exposed to jump-like disturbances at the same time, replenishment becomes less predictable, local shortages emerge more frequently, and the network must rely more heavily on protective inventory positioning and emergency stock reallocation. In this sense, the more moderate increases observed for HRL-PPO suggest that hierarchical coordination still contains part of the disruption burden, even though its resilience advantage over flat PPO becomes clearly more scenario-dependent. Moreover, the scenario-level picture remains qualitatively informative: resilience still appears to depend less on local capacity abundance alone and more on the extent to which the overall capacity structure supports responsive replenishment and

 doi:10.20944/preprints202603.1244.v1

coordination. However, under the Merton-only demand setting, differences across scenarios become narrower and the relative advantage of hierarchy becomes more scenario-dependent, suggesting that extreme shocks reduce the exploitable benefit of more favorable capacity configurations. Finally, the scale effect becomes even more critical in this setting, as the expansion in the number of stores further amplifies coordination complexity and disturbance propagation across the network. Hence, the comparative reading of the two tables suggests that the proposed framework preserves competitiveness and adaptability under the harshest operating conditions, but its resilience advantage over flat PPO is clearly compressed and no longer uniform when shocks become system-wide rather than partially absorbed through a blended demand structure.

Table 5. Comparative results for the second demand configuration (Merton-type demand for all products) across the business scenarios studied.

Scenario	Stores S	Base-stock / order-up-to rule			Greedy fulfillment / re-balancing rule			PPO			HRL-PPO		
		Holding cost	Inter-seller net transshipment	Lost sales rate	Holding cost	Inter-seller net transshipment	Lost sales rate	Holding cost	Inter-seller net transshipment	Lost sales rate	Holding cost	Inter-seller net transshipment	Lost sales rate
Scenario 1	2	218.89 ± 7.30	79.60 ± 4.24	0.12 ± 0.01	205.76 ± 6.81	88.10 ± 4.58	0.11 ± 0.01	200.38 ± 5.48	77.66 ± 3.39	0.10 ± 0.01	186.86 ± 4.63	70.33 ± 2.83	0.1640 ± 0.0085
	7	505.08 ± 16.80	245.49 ± 12.83	0.14 ± 0.01	534.40 ± 17.98	240.98 ± 12.53	0.15 ± 0.01	486.16 ± 13.13	221.43 ± 9.97	0.13 ± 0.01	447.25 ± 11.05	195.37 ± 8.07	0.1750 ± 0.0090
	12	808.29 ± 26.88	496.06 ± 26.94	0.17 ± 0.01	761.29 ± 25.23	549.14 ± 29.06	0.16 ± 0.01	739.01 ± 19.58	479.24 ± 21.58	0.15 ± 0.01	674.60 ± 16.66	421.03 ± 16.93	0.1980 ± 0.0101
	18	1177.90 ± 39.78	755.78 ± 41.81	0.19 ± 0.01	1245.81 ± 41.92	741.93 ± 39.98	0.20 ± 0.01	1127.34 ± 31.02	687.66 ± 31.68	0.17 ± 0.01	1026.03 ± 25.87	606.16 ± 25.67	0.2080 ± 0.0107
	24	1558.48 ± 53.45	1027.77 ± 58.97	0.22 ± 0.01	1468.09 ± 50.90	1137.73 ± 62.40	0.21 ± 0.01	1420.68 ± 39.81	990.06 ± 46.65	0.19 ± 0.01	1278.31 ± 33.55	865.02 ± 36.63	0.2260 ± 0.0119
	30	1946.72 ± 68.78	1307.22 ± 76.35	0.24 ± 0.01	2058.05 ± 73.44	1283.27 ± 71.62	0.25 ± 0.02	1858.52 ± 53.03	1180.61 ± 55.63	0.22 ± 0.01	1667.35 ± 42.90	1020.79 ± 41.07	0.2480 ± 0.0132
Scenario 2	2	233.31 ± 7.88	92.72 ± 5.03	0.14 ± 0.01	246.96 ± 8.43	91.02 ± 4.81	0.15 ± 0.01	224.24 ± 6.28	83.36 ± 3.75	0.13 ± 0.01	204.28 ± 5.26	73.86 ± 3.13	0.1380 ± 0.0078
	7	547.66 ± 18.78	284.65 ± 15.45	0.17 ± 0.01	515.90 ± 17.89	315.11 ± 16.68	0.16 ± 0.01	493.91 ± 13.85	275.05 ± 12.39	0.14 ± 0.01	444.47 ± 11.20	239.66 ± 9.89	0.1570 ± 0.0086
	12	877.35 ± 30.09	574.19 ± 32.35	0.19 ± 0.01	929.43 ± 32.22	563.67 ± 30.92	0.20 ± 0.01	843.76 ± 23.65	514.46 ± 24.24	0.17 ± 0.01	756.92 ± 19.47	446.34 ± 19.38	0.1680 ± 0.0093
	18	1280.04 ± 45.23	874.28 ± 49.28	0.22 ± 0.01	1204.23 ± 42.97	966.85 ± 53.02	0.21 ± 0.01	1153.14 ± 32.90	824.51 ± 38.85	0.19 ± 0.01	1033.64 ± 26.60	716.16 ± 30.33	0.1840 ± 0.0101
	24	1692.05 ± 60.66	1187.39 ± 69.35	0.25 ± 0.02	1792.18 ± 64.59	1165.63 ± 66.18	0.26 ± 0.02	1617.14 ± 46.14	1072.38 ± 51.66	0.22 ± 0.01	1439.13 ± 37.77	921.17 ± 39.01	0.2020 ± 0.0116
	30	2116.93 ± 77.00	1508.83 ± 89.65	0.28 ± 0.02	1990.64 ± 73.07	1670.26 ± 96.43	0.27 ± 0.02	1895.50 ± 56.01	1421.21 ± 68.45	0.24 ± 0.01	1688.12 ± 45.17	1219.34 ± 51.65	0.2280 ± 0.0128
Scenario 3	2	250.80 ± 8.73	107.85 ± 5.97	0.16 ± 0.01	236.30 ± 8.31	119.40 ± 6.44	0.17 ± 0.01	226.20 ± 6.69	102.50 ± 4.62	0.14 ± 0.01	205.12 ± 5.49	90.62 ± 3.83	0.1296 ± 0.0076
	7	591.95 ± 20.91	332.09 ± 18.38	0.20 ± 0.01	626.16 ± 22.35	326.00 ± 17.90	0.19 ± 0.01	551.56 ± 16.02	293.41 ± 13.21	0.17 ± 0.01	492.15 ± 13.17	253.10 ± 10.72	0.1512 ± 0.0084
	12	951.31 ± 34.60	669.31 ± 38.41	0.23 ± 0.01	895.99 ± 32.88	741.05 ± 41.36	0.22 ± 0.01	840.12 ± 24.82	634.34 ± 29.23	0.19 ± 0.01	745.41 ± 19.95	547.61 ± 23.19	0.1728 ± 0.0096
	18	1390.81 ± 52.04	1017.91 ± 59.45	0.26 ± 0.02	1470.95 ± 55.49	999.26 ± 56.73	0.27 ± 0.02	1299.51 ± 39.73	899.34 ± 44.26	0.22 ± 0.01	1148.65 ± 30.74	769.24 ± 33.39	0.1944 ± 0.0109
	24	1842.41 ± 70.84	1382.27 ± 83.56	0.30 ± 0.02	1735.52 ± 67.25	1530.16 ± 89.03	0.29 ± 0.02	1618.81 ± 51.14	1291.19 ± 63.55	0.25 ± 0.01	1411.82 ± 39.23	1104.81 ± 47.96	0.2268 ± 0.0127
	30	2306.16 ± 91.08	1756.52 ± 109.78	0.33 ± 0.02	2438.98 ± 96.99	1724.33 ± 103.90	0.35 ± 0.02	2112.85 ± 66.75	1545.85 ± 77.70	0.28 ± 0.02	1840.71 ± 51.15	1310.82 ± 56.91	0.2484 ± 0.0139

5.3. How the Capacities on Specific Nodes Affect the Resilience of the Network?

Previous analysis validated that HRL-PPO may serve as a promising decision-support framework for resilient inventory control under capacitated omnichannel settings, particularly under demand uncertainty and shock exposure. At the same time, the insights obtained so far indicated that resilience is not shaped by capacity abundance in a generic sense, but rather by the way specific capacity elements interact across the network, especially those related to store-side storage, lateral transshipment capability, and replenishment support from the upstream node. Building upon these findings, this subsection aims to develop business-oriented insights regarding how capacity placed at specific nodes influences network resilience and service preservation, with particular emphasis on the inventory-positioning logic emerging in the capacitated problem analyzed. In this direction, the focus shifts from the comparative performance of policies to the structural interpretation of capacity allocation, so as to better inform how inventory positioning across the nodes of the network contributes to loss mitigation, responsiveness, and robust inventory propagation under disturbances.

Capitalizing on the evidence emerged from our analysis, this section introduces an exploratory index intended to summarize the capacity-side conditions that appear to influence network resilience most strongly. Previous results suggested that service preservation is shaped less by isolated capacity abundance and more by the interaction between local storage support, lateral inventory mobility, and

 doi:10.20944/preprints202603.1244.v1

the degree of dependence on upstream replenishment. In this direction, and in order to provide a business-oriented interpretation of inventory positioning in the capacitated network, we introduce the *Transfer–Storage-to-Central-Replenishment* metric (TSCR), formally defined in Equation 24.

$$\text{TSCR} = \frac{C_s \cdot C^{tr}}{C^{fc}} \tag{24}$$

The factors included in Equation 24 reflect the structural patterns that emerged most clearly from the problem setting and the corresponding experimental observations. The term C_s represents local storage support at the store level, while C^{tr} captures the ability of the network to re-position inventory laterally across sellers when shortages emerge. These two elements are expressed in multiplicative form not as a uniquely derived interaction law, but as a parsimonious way to summarize their joint availability in the present setting, where resilience appears to depend on their combined contribution rather than on either one in isolation. The denominator C^{fc} is introduced as a normalizing term, since the FC-to-store replenishment constitutes the main upstream support mechanism of the network. In this regard, the ratio is intended to summarize the extent to which local storage and lateral mobility can support the network relative to central replenishment dependence. Under this interpretation, higher TSCR values indicate stronger local buffering and lateral flexibility, whereas lower values indicate greater reliance on the central node for service preservation. By construction, however, TSCR does not incorporate all structural drivers varied in the experiments, such as lead-time parameters, FC inventory capacity, supplier caps, or store-to-online capacity, and should therefore be interpreted as a compact descriptive index rather than a complete resilience construct.

The use of this metric provides a compact descriptive lens through which the relationship between selected capacity-side characteristics and resilience to demand shocks can be visualized, thereby supporting the main question examined in this subsection. To document this relationship, we adopt a two-fold procedure. First, TSCR is computed for each of the three capacity settings defined by the examined business scenarios. Second, these values are paired with the corresponding mean lost-sales rates observed for each channel, each store-scale configuration, and both demand-pattern settings. In Figure 7, each circle therefore represents the mean lost-sales rate associated with a specific store scale under the corresponding TSCR value. The dashed lines connect these mean values separately for the blend and Merton-only demand settings, while the solid black and red lines trace the median path of the respective sets of means so as to emphasize the common monotonic tendency. This construction also implies that the same monotonic relationship can be read at the level of each store scale by joining the corresponding circles across the three capacity settings; for instance, the exact monotonic curve for the 12-store case is obtained by connecting the third circle in each panel. The resulting patterns should be interpreted as empirical summaries of the tested configurations rather than as statistically validated threshold rules for resilience assessment.

 doi:10.20944/preprints202603.1244.v1

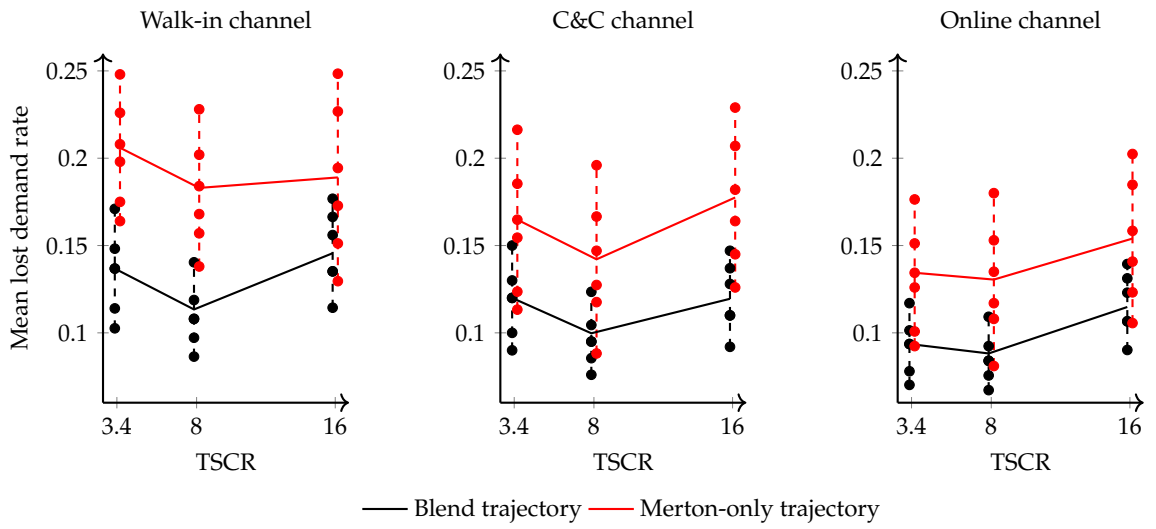

Figure 7. An aggregated view of the relationship between the mean lost sales rate per number of stores and network capacity, examined through the lens of the composite indicator TSCR.

The figure indicates two regularities that hold across the three channels. First, for any fixed TSCR level, mean lost-sales rates increase with the number of stores, which implies that network expansion amplifies coordination pressure when the capacity architecture remains unchanged. Second, the fully shock-driven product configuration shifts all channel profiles upward relative to the blended case. Quantitatively, the central walk-in loss level rises from 13.68%, 11.34%, and 14.56% to 20.60%, 18.30%, and 18.90% across the three TSCR levels, while the corresponding central levels for click-and-collect rise from 12.00%, 9.98%, and 11.95% to 16.48%, 14.21%, and 17.75%, and for online demand from 9.36%, 8.82%, and 11.48% to 13.44%, 13.05%, and 15.40%. This pattern is consistent with the adopted demand calibration: walk-in remains the most loss-exposed channel because it carries the strongest baseline flow, whereas online demand is the most shock-sensitive component because it combines the highest jump frequency and jump magnitude, while click-and-collect remains structurally thinner but deteriorates visibly when shocks propagate across the full assortment. An important notion emerging from this analysis is that the proposed formulation makes it possible to express the resilience properties of the network through a compact relationship between its capacity-side structure and the corresponding lost-sales behavior. On this basis, and by exploiting the TSCR-based representation designed above, the empirical evidence can be summarized as follows:

$$TSCR = \begin{cases} \leq 3.6 \;\Rightarrow\; \bar{LS}_w \leq 0.248,\ \bar{LS}_{cc} \leq 0.216,\ \bar{LS}_{on} \leq 0.176, \\ \leq 8.2 \;\Rightarrow\; \bar{LS}_w \leq 0.228,\ \bar{LS}_{cc} \leq 0.196,\ \bar{LS}_{on} \leq 0.180, \\ \leq 16.2 \;\Rightarrow\; \bar{LS}_w \leq 0.248,\ \bar{LS}_{cc} \leq 0.229,\ \bar{LS}_{on} \leq 0.202. \end{cases}$$

On the managerial side, the above analysis could be translated into a more channel-sensitive capacity control logic. In simpler terms, the most prominent configuration depends not only on whether the assortment is partially or fully exposed to jump-driven demand, but also on which channel is strategically prioritized. When walk-in demand is dominant, resilience depends primarily on stronger downstream capacity, that is, relatively higher store-side inventory and local fulfillment capability, while upstream support may remain moderate but stable (e.g., higher store capacity and store-side shipping capability, with comparatively balanced FC support). When online demand becomes the main service priority, the relevant configuration shifts toward stronger upstream capacity, namely higher FC inventory availability, greater FC outbound capability, and a more responsive FC-to-store replenishment interface, since digital demand is more exposed to shock amplification and cross-node reallocation (e.g., larger FC buffers and stronger FC shipping capacity, while local expansion alone remains insufficient). By contrast, if click-and-collect is prioritized, the most effective design

 doi:10.20944/preprints202603.1244.v1

is an intermediate one, in which store-side availability is reinforced enough to preserve rapid order servicing at the local node, but without materially weakening upstream support. At the same time, the inter-seller transshipment layer should also be calibrated accordingly, since it constitutes an additional resilience lever within the TSCR logic: when local demand asymmetries are expected to be moderate, a moderate re-balancing capability across stores is sufficient, whereas under stronger shock exposure or greater online volatility, higher inter-seller transfer capacity becomes more valuable because it allows inventory to be repositioned more quickly across the network and partially compensates for local shortages. Hence, the managerial implication of the TSCR analysis is not that one node should systematically dominate the capacity design, but rather that the relative emphasis placed on store capacity, FC capacity, inter-echelon responsiveness, and inter-seller transshipment capability should be adjusted according to the expected demand profile and the channel whose service continuity is treated as operationally dominant.

6. Discussion

The results consistently indicate that the proposed HRL–PPO framework provides a highly competitive control architecture for omnichannel SCs relative to both the flat PPO benchmark and the rule-based heuristics, particularly as network complexity and demand volatility increase. Across the examined demand settings, the hierarchical formulation achieves lower holding costs, lower inter-store transshipment volumes, and lower lost-sales rates in most scenarios and store scales, which jointly suggest better coordination of inventory positioning and fulfillment timing. This advantage appears especially pronounced in medium- and large-scale instances, where the dimensionality of the control problem becomes more severe and where the operational consequences of mistimed replenishment and routing decisions are amplified. The results therefore support the central premise of the study, namely that the explicit temporal decomposition of decisions into slower replenishment cycles and faster fulfillment adjustments allows the policy to align more closely with the natural rhythm of omnichannel operations. In this sense, the hierarchical structure does not merely improve learning performance in a technical sense, but also appears to provide a more managerially meaningful representation of how inventory and service decisions are actually organized in retail networks under uncertainty.

A second important finding concerns the role of demand shocks and operating structure in shaping the relative value of intelligent control. When all products follow Merton-type demand dynamics, performance differences between methods remain substantial, but the relative advantage of HRL–PPO over flat PPO becomes more scenario-dependent, especially in more constrained scenarios and at higher store counts. This pattern suggests that abrupt and localized demand surges increase the need for adaptive coordination mechanisms capable of jointly managing scarce inventory, fulfillment capacity, and rebalancing opportunities across the network. At the same time, the comparison with the perfect-information benchmark confirms that—even though HRL–PPO substantially improves over implementable benchmarks—it still operates below a strong informational reference, as expected in a realistic stochastic environment where future demand is not known ex ante. This gap is analytically useful, because it shows both that the proposed method captures a substantial share of the attainable operational value and that further improvement remains possible through richer forecasting, stronger state representations, or more advanced hierarchical coordination mechanisms. Overall, the findings suggest that HRL is particularly promising for resilient omnichannel control in environments where demand shocks coincide with binding capacity constraints and where the cost of poorly synchronized decisions propagates across multiple channels and echelons. At the same time, these findings remain conditional on the common forecast-based interface adopted in the simulator, rather than being fully independent of forecasting assumptions.

An additional insight from the analysis concerns the structural role of capacity allocation in shaping network resilience. The results indicate that resilience does not depend only on the absolute amount of capacity available in the network, but also on how this capacity is distributed across stores, lateral transfers, and the central fulfillment node. This relationship is summarized descriptively by the

 doi:10.20944/preprints202603.1244.v1

Transfer–Storage-to-Central-Replenishment (TSCR) index introduced earlier. The empirical evidence suggests that more balanced TSCR configurations are associated with better shock absorption, as they combine local buffering capacity with sufficient inventory mobility across stores. Therefore, the advantages of the HRL framework should be interpreted not only as a consequence of the learning architecture, but also as an indication that adaptive control policies are better able to exploit balanced capacity structures when responding to demand disturbances. Nevertheless, TSCR should be viewed here as a compact interpretive index rather than as a statistically validated explanatory construct.

The proposed framework complements and also extends several recent RL-based approaches relevant to omnichannel retailing by addressing limitations related to operational integration, network structure and temporal dynamics of decision-making. Existing studies have demonstrated the potential of RL in various omnichannel contexts. For example, RL has been used for integrated replenishment and fulfillment control [23], joint pricing–inventory optimization under demand uncertainty [27] and operational store-level decisions such as picker routing [31]. Other contributions have explored RL within broader behavioral or analytics-oriented frameworks, including models that incorporate quantum-inspired customer decision dynamics [36], hybrid architectures for loyalty prediction [37], multi-objective omnichannel optimization under behavioral uncertainty [38] and RL-driven marketing analytics [39]. While these studies demonstrate the flexibility of RL in retail environments, many of them focus on either stylized retail settings, behavioral and pricing decisions, or localized operational tasks. As a result, they often provide limited representation of multi-echelon inventory interactions, lateral inventory rebalancing across locations, capacity-coupled fulfillment processes, and the explicit coordination of decisions across different operational time scales. The framework proposed in this study contributes to this literature by introducing a HRL architecture that explicitly captures the temporal structure of omnichannel decision-making while simultaneously modeling a capacitated multi-echelon network with lateral transshipment and shock-sensitive demand dynamics. In this sense, the proposed approach moves toward a more operationally integrated representation of omnichannel SCs and demonstrates how hierarchical learning can support coordinated inventory positioning and fulfillment control in complex retail networks.

6.1. Managerial implications

From a managerial perspective, the findings suggest that omnichannel performance depends not only on the amount of inventory available in the network, but also on the timing architecture through which decisions are made. Retail managers often face the practical challenge of combining slower tactical decisions, such as replenishment and inventory positioning, with faster operational decisions, such as daily fulfillment routing and local stock rebalancing. The results indicate that treating these decisions within a unified but hierarchically structured control framework can materially improve service reliability and cost efficiency, especially in networks exposed to demand surges and capacity bottlenecks. In practical terms, this means that firms may benefit from designing their digital control towers, planning routines, and AI-supported decision systems around differentiated decision cadences rather than relying on a single, uniform planning frequency. Such an approach is particularly relevant for retailers operating ship-from-store, BOPIS, and home-delivery models simultaneously, where the misalignment between replenishment timing and fulfillment responsiveness can quickly translate into lost sales and unnecessary inventory movement.

The results also carry implications for network design and resilience planning. Specifically, the stronger relative performance of the HRL framework under shock-prone demand conditions suggests that retailers should view adaptive learning-based control as a resilience capability rather than merely an automation tool. When demand shocks coincide with binding FC, store, or transshipment capacities, static rules appear increasingly unable to allocate scarce resources efficiently across channels and locations. Managers should therefore place greater emphasis on building data infrastructures that support real-time inventory visibility, cross-node coordination, and dynamic rebalancing decisions. At the same time, the remaining gap relative to the perfect-information oracle indicates that operational excellence will still depend on complementary investments in forecasting quality, process standardization,

 doi:10.20944/preprints202603.1244.v1

and scenario-based stress testing. Consequently, the main managerial implication is not that AI can eliminate uncertainty, but that properly structured hierarchical decision systems can help organizations absorb uncertainty more effectively and translate network flexibility into measurable economic and service gains.

6.2. Limitations and areas for further research

A limitation of the present study concerns several modeling and architectural choices that were intentionally made to preserve analytical focus and implementation tractability. First, the proposed HRL–PPO framework relies on feed-forward MLPs for both actors and critics. Although this choice is suitable for establishing the feasibility and performance of the hierarchical control logic, it does not exhaust the range of solver architectures that could be aligned with the conceptual structure of the problem. In particular, because the proposed scheme is closely related to the logic of Feudal Reinforcement Learning, future research could examine whether recently proposed feudal neural network (NN) architectures provide superior hierarchical representation, credit assignment, and scalability in large omnichannel settings. Second, the current formulation assumes homogeneous monetary units across products and channels, so that the analysis remains centered on the logistics and fulfillment complexity of the network rather than on endogenous pricing heterogeneity. While this assumption is appropriate for isolating the operational value of hierarchical control, it abstracts from important retail realities in which margins, markdown policies, channel-specific prices, and promotional interventions may differ substantially across products. Extending the model to incorporate dynamic pricing and heterogeneous revenue structures would therefore be a valuable direction for assessing how pricing policies interact with shock resistance and inventory resilience. Third, although the study considers both mixed and fully shock-driven demand settings through uniform and Merton-type processes, these specifications still represent stylized approximations of demand behavior; in practice, some products may exhibit more intense, asymmetric, or prolonged peaks than those captured in the current experiments. In the same spirit, the forecasting layer embedded in the simulator is kept deliberately simple and calibrated through exponential smoothing, so the reported resilience gains should be interpreted as conditional on the adopted forecast interface rather than as fully independent of forecasting assumptions. Fourth, the TSCR indicator introduced for interpretive purposes should be viewed as a compact descriptive index rather than as a statistically validated explanatory construct. Although it is useful for summarizing selected capacity-side relationships observed in the experiments, it does not incorporate all structural drivers varied in the analysis, nor is it benchmarked here against alternative composite metrics. Finally, the modeling framework adopts a centralized decision architecture, which is justified here because omnichannel retail operations often admit a high degree of observability and information integration across the network. Even so, centralized control is not the only plausible learning architecture. A multi-agent formulation—particularly under a centralized-training, decentralized-execution (CTDE) paradigm—remains an important alternative for future work, especially in settings where local autonomy, organizational decentralization, or computational decomposition become more prominent.

7. Concluding Remarks

This study develops and evaluates a centralized HRL framework for omnichannel SCs operating under stochastic demand and shock conditions. By explicitly separating weekly replenishment and allocation decisions from daily fulfillment and lateral rebalancing decisions, the proposed HRL–PPO scheme captures the multi-timescale structure that naturally characterizes omnichannel operations. The experimental findings show that this hierarchical timing structure yields consistent advantages over flat PPO and rule-based heuristics across different network scales, business scenarios, and demand configurations, particularly when demand shocks interact with binding inventory and fulfillment capacities. At the same time, the comparison with the perfect-information oracle confirms that the proposed method remains realistically suboptimal, while still recovering a substantial share of the attainable value under uncertainty. Overall, the study contributes to the growing literature on AI-

 doi:10.20944/preprints202603.1244.v1

driven retail operations by showing that hierarchical learning is not only a computationally viable solution method, but also a managerially meaningful control paradigm for improving resilience, service performance, and cost efficiency in complex omnichannel supply networks.

Author Contributions: Conceptualization, P.G. and T.D.; methodology, P.G.; software, P.G.; validation, P.G. and T.D.; formal analysis, P.G.; data curation, P.G.; writing—original draft preparation, P.G. and T.D.; writing—review and editing, T.D.; visualization, P.G.; supervision, T.D.; project administration, P.G.; funding acquisition, T.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially funded by the Hellenic Open University under the project "Multi-agent systems and Generative Artificial Intelligence in Management science: Innovative Applications in Human Resources and Operations management-PELOPAS".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data will be made available on request.

Acknowledgments: During the preparation of this manuscript, the authors used Generative AI tools for correcting grammatical errors in the initially developed text. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

SC	Supply Chain
FC	Fulfillment Center
RL	Reinforcement Learning
HRL	Hierarchical Reinforcement Learning
BOPS	Buy-online-pickup-in-store
MDP	Markov Decision Process
NN	Neural Network
PPO	Proximal Policy Optimization
MLP	Multi-layered Perceptron
AI	Artificial Intelligence
CTDE	Centralized Training with Decentralized Execution
RMSE	Root Mean Squared Error
TSCR	Transfer–Storage-to-Central-Replenishment

Appendix A. Centralized Benchmark Policies

This appendix summarizes the implementation logic of the three benchmark policies used in the evaluation protocol. All three policies operate on the same simulator and obey the same feasibility rules and action interface introduced in Equation 11–Equation 12. Hence, they act on the common environment state and produce the same classes of controls, namely store-level online-fulfillment shares, transshipment-inducing stock targets, and, at replenishment epochs, FC-to-store and supplier-to-FC shipment quantities. Unlike the proposed HRL scheme, however, these benchmarks are centrally coordinated and do not rely on a manager–worker decomposition. Their logic is therefore rule-based, with decisions computed directly from the current global network state.

A common implementation feature is the use of two temporal layers. At primitive periods, the benchmark policies determine the online-routing shares and the target stock levels that guide inter-store re-balancing. At replenishment epochs, they additionally determine shipment quantities from the FC to stores and order quantities from the supplier to the FC. In all cases, the resulting controls are passed through the same simulator-side feasibility logic as the learning-based policies,

 doi:10.20944/preprints202603.1244.v1

so that inventory, transport, batch-execution, and no-split fulfillment constraints remain identical across all compared methods. The two business heuristics rely on the forecast state defined through Equation 1, whereas the perfect-information benchmark replaces forecasts with direct access to the realized demand tape.

Appendix A.1. Base-Stock/order-up-to Benchmark

The base-stock benchmark constructs, for each store–product pair (s, p), a cycle target stock level by combining forecasted local demand and a weighted share of forecasted online demand. Denoting by $\widehat{D}^{loc}_{tsp}$ the forecasted local demand rate (walk-in plus click-and-collect) and by $\widehat{D}^{on}_{tsp}$ the forecasted online demand rate, the target is

$$B^{s}_{tsp} = \min\Big\{ C^{s}_{p},\ L\,\widehat{D}^{loc}_{tsp} + \beta L\,\widehat{D}^{on}_{tsp} \Big\}, \tag{A1}$$

where L is the replenishment-cycle length (weekly in the implementation) and $\beta \in [0, 1]$ controls the intended store-side share of online demand. Letting IP^{s}_{tp} denote the store inventory position (i.e., the inventory available to support store-level fulfillment decisions), the FC-to-store shipment request is formed as

$$y_{tps} = \left(B^{s}_{tsp} - \mathrm{IP}^{s}_{tp}\right)^{+}, \tag{A2}$$

subject to the simulator-side capacity limits. The supplier order is computed analogously from an FC target

$$B^{f}_{tp} = \sum_{s \in \mathcal{S}} y_{tps} + (1 - \beta) L \sum_{s \in \mathcal{S}} \widehat{D}^{on}_{tsp}, \tag{A3}$$

so that

$$y_{tpf} = \left(B^{f}_{tp} - \mathrm{IP}^{f}_{tp}\right)^{+}, \tag{A4}$$

where IP^{f}_{tp} denotes the FC inventory position.

At the daily level, the policy converts the cycle target into a daily reference $b^{s}_{tsp} = B^{s}_{tsp}/L$. This reference is used in two ways. First, it sets the online-routing share as an increasing function of excess store inventory relative to b^{s}_{tsp}, namely

$$\alpha^{s}_{tp} = \mathrm{clip}\left(\frac{I^{s}_{tp} - b^{s}_{tsp}}{C^{s}_{p}},\ 0,\ 1 \right), \tag{A5}$$

where $\mathrm{clip}(\cdot, 0, 1)$ truncates the value to the unit interval and I^{s}_{tp} denotes on-hand store inventory. Second, the same dailyized reference is passed to the simulator as the target stock level around which transshipment is executed. Accordingly, the benchmark can be viewed as an order-up-to rule with forecast-based target positioning applied consistently across replenishment, online fulfillment, and transshipment guidance.

Appendix A.2. Greedy Fulfillment/Re-Balancing Benchmark

The greedy benchmark is more short-sighted and relies on near-term protection levels instead of cycle-level order-up-to targets. Specifically, for each store–product pair it defines a daily safety target as

$$z_{tsp} = \min\Big\{ C^{s}_{p},\ h_{\mathrm{loc}}\,\widehat{D}^{loc}_{tsp} + h_{\mathrm{on}}\,\widehat{D}^{on}_{tsp} \Big\}, \tag{A6}$$

where h_{loc} and h_{on} denote the local-demand and online-demand protection horizons, respectively. This target is passed to the simulator as the desired stock level around which inter-store transshipment is greedily executed, so that stores with positive surplus relative to z_{tsp} can support stores facing deficits.

 doi:10.20944/preprints202603.1244.v1

The online-routing share is again determined from the deviation between current stock and the safety target, according to

$$\alpha^s_{tp} = \mathrm{clip}\left(\frac{I^s_{tp} - z_{tsp}}{C^s_p}, 0, 1\right), \tag{A7}$$

thereby favoring store-based online fulfillment when local inventory is sufficiently abundant. At replenishment epochs, the benchmark forms FC-to-store shipment requirements from forecasted cycle demand plus a weighted store-side contribution of forecasted online demand, i.e.,

$$y_{tps} = \left(L\,\widehat{D}^{loc}_{tsp} + \beta L\,\widehat{D}^{on}_{tsp} - \mathrm{IP}^s_{tp}\right)^+, \tag{A8}$$

and computes supplier orders from the resulting FC inventory gap as

$$y_{tpf} = \left(\sum_{s\in\mathcal{S}} y_{tps} + (1-\beta)L \sum_{s\in\mathcal{S}} \widehat{D}^{on}_{tsp} - \mathrm{IP}^f_{tp}\right)^+. \tag{A9}$$

In this sense, the benchmark combines myopic daily re-balancing with a simple forecast-driven replenishment logic at cycle epochs.

Appendix A.3. Perfect-Information Benchmark

The third benchmark preserves the same action structure as the two business heuristics, but replaces forecast-based inputs with realized future demand values observed directly from the finite-horizon demand tape. It is therefore implemented as a perfect-information benchmark under the same simulator rules. Strictly speaking, however, it should not be interpreted as a mathematically optimal oracle or upper bound, but rather as a privileged-information benchmark policy constructed under the same action and feasibility structure.

At replenishment epochs, the benchmark computes exact cycle-ahead store requirements from realized walk-in, click-and-collect, and store-eligible online demand over the upcoming cycle. Denoting these realized cumulative quantities by $D^{loc,*}_{tsp}$ (exact local demand over the cycle) and $D^{on,*}_{tsp}$ (exact online demand over the cycle), the store requirement is

$$B^{s,*}_{tsp} = \min\left\{C^s_p,\, D^{loc,*}_{tsp} + \beta D^{on,*}_{tsp}\right\}, \tag{A10}$$

which yields

$$y^*_{tps} = \left(B^{s,*}_{tsp} - \mathrm{IP}^s_{tp}\right)^+. \tag{A11}$$

The supplier order is then formed from the exact FC requirement,

$$B^{f,*}_{tp} = \sum_{s\in\mathcal{S}} y^*_{tps} + (1-\beta)\sum_{s\in\mathcal{S}} D^{on,*}_{tsp}, \qquad y^*_{tpf} = \left(B^{f,*}_{tp} - \mathrm{IP}^f_{tp}\right)^+. \tag{A12}$$

At the daily level, the benchmark sets re-balancing targets from the exact same-day local and store-eligible online fulfillment burden and determines store-based online-routing shares from exact same-day demand and expected residual store inventory. In implementation terms, this means that the benchmark uses realized future demand in place of the forecast quantities $\widehat{D}^{loc}_{tsp}$ and $\widehat{D}^{on}_{tsp}$, while leaving the action structure and simulator-side execution rules unchanged.

Therefore, the distinction between the two business heuristics and the perfect-information benchmark does not lie in the action space or in the simulator constraints, but in the information basis on which decisions are formed: the former rely on the forecast state, whereas the latter uses exact realized demand over the relevant forward window. This makes the benchmark a useful high-performance reference point for assessing how closely the learned policy approaches the best decisions that can be formed when future demand information is fully available under the same simulator rules.

 doi:10.20944/preprints202603.1244.v1

References

1. Alemany, M.M.E.; Alarcón, F.; Lario, F.-C.; Boj, J.J. An application to support the temporal and spatial distributed decision-making process in supply chain collaborative planning. *Comput. Ind.* **2011**, *62*(5), 519–540. https://doi.org/10.1016/j.compind.2011.02.002.
2. Boute, R. N.; Gijsbrechts, J.; van Jaarsveld, W.; Vanvuchelen, N. Deep reinforcement learning for inventory control: A roadmap. *Eur. J. Oper. Res.* **2022**, *298*(2), 401–412. https://doi.org/10.1016/j.ejor.2021.07.016.
3. Boysen, N.; Stephan, K.; Weidinger, F. Manual order consolidation with put walls: The batched order bin sequencing problem. *EURO J. Transp. Logist.* **2019**, *8*, 169–193. https://doi.org/10.1007/s13676-018-0116-0.
4. Cai, Y.; Lo, C.K.Y. omnichannel management in the new retailing era: A systematic review and future research agenda. *Int. J. Prod. Econ.* **2020**, *229*, 107729. https://doi.org/10.1016/j.ijpe.2020.107729.
5. Chen, Z.; Su, S.I.I. Omnichannel consignment supply chain cooperation: A comparative analysis of game-theoretical models. *Int. J. Manag. Sci. Eng. Manag.* **2021**, *16*, 151–164. https://doi.org/10.1080/17509653.2021.1911004.
6. Gardner, E. S. Jr. Exponential smoothing: The state of the art—Part II. *Int. J. Forecast.* **2006**, *22*(4), 637–666. https://doi.org/10.1016/j.ijforecast.2006.03.005.
7. Giannopoulos, P.G.; Malamas, V.; Dasaklis, T.K. Coopetition dynamics in platform-based supply chains: When and with whom to cooperate? *SSRN* **2025**.
8. Giannopoulos, P.G.; Malamas, V.; Verykios, V.; Dasaklis, T.K. Mitigating Covariate Shift in Managerial Decision-Making: A Tailored Data Augmentation Approach for Offline Behavioral Cloning. In Proceedings of the 16th International Conference on Information, Intelligence, Systems & Applications (IISA), Mytilene, island of Lesvos, Greece, 1–8, 2025. https://doi.org/10.1109/IISA66859.2025.11311267.
9. Goedhart, J.; Haijema, R.; Akkerman, R. Modelling the influence of returns for an omnichannel retailer. *Eur. J. Oper. Res.* **2023**, *306*, 1248–1263. https://doi.org/10.1016/j.ejor.2022.08.021.
10. Guo, J.; Keskin, B.B. Designing a centralized distribution system for omnichannel retailing. *Prod. Oper. Manag.* **2023**, *32*, 1724–1742. https://doi.org/10.1111/poms.13936.
11. Gupta, V.K.; Dakare, S.; Fernandes, K.J.; Thakur, L.S.; Tiwari, M.K. Bilevel programming for manufacturers operating in an omnichannel retailing environment. *IEEE Trans. Eng. Manag.* **2023**, *70*, 3958–3975. https://doi.org/10.1109/TEM.2021.3090653.
12. Hammami, R.; Frein, Y. A capacitated multi-echelon inventory placement model under lead time constraints. *Prod. Oper. Manag.* **2014**, *23*, 446–462. https://doi.org/10.1111/poms.12060.
13. Han, X. LG-H-PPO: Offline hierarchical PPO for robot path planning on a latent graph. *Front. Robot. AI* **2026**, *12*, 1737238. https://doi.org/10.3389/frobt.2025.1737238.
14. Huang, S.; Xie, H.; Zhang, Y.; Chiu, C.H. Buy-online-pick-up-at-store benefits supply chains considering supplier encroachment. *Prod. Oper. Manag.* **2025**. https://doi.org/10.1177/10591478251342693.
15. Hui, Y.P.J.; Qu, T.; Pan, Y.; Wang, L.; Ding, L.; Huang, G.Q. Multi-echelon distribution network inventory optimization for cross-border omnichannel e-commerce. *Ind. Manag. Data Syst.* **2025**, 1–36. https://doi.org/10.1108/IMDS-05-2025-0723.
16. Hyndman, R. J.; Koehler, A. B.; Snyder, R. D.; Grose, S. A state space framework for automatic forecasting using exponential smoothing methods. *Int. J. Forecast.* **2002**, *18*(3), 439–454. https://doi.org/10.1016/S0169-2070(01)00110-8.
17. Ishfaq, R.; Defee, C.C.; Gibson, B.J.; Raja, U. Realignment of the physical distribution process in omnichannel fulfillment. *Int. J. Phys. Distrib. Logist. Manag.* **2016**, *46*, 543–561. https://doi.org/10.1108/IJPDLM-02-2015-0032.
18. İzmirli, D.; Yetkin Ekren, B.Y.; Kumar, V. Inventory share policy designs for a sustainable omnichannel e-commerce network. *Sustainability* **2020**, *12*, 10022. https://doi.org/10.3390/su122310022.
19. İzmirli, D.; Yetkin Ekren, B.Y.; Kumar, V.; Pongsakornrungsilp, S. omnichannel network design towards circular economy under inventory share policies. *Sustainability* **2021**, *13*, 2875. https://doi.org/10.3390/su13052875.
20. Jia, Y.; Zhou, X.Y. Policy gradient and actor-critic learning in continuous time and space: Theory and algorithms. *J. Mach. Learn. Res.* **2022**, *23*(275), 1–50. Available online: https://jmlr.org/papers/v23/21-1387.html.
21. Karimi-Mamaghan, M.; Mohammadi, M.; Meyer, P.; Karimi-Mamaghan, A.M.; Talbi, E.-G. Machine learning at the service of meta-heuristics for solving combinatorial optimization problems: A state-of-the-art. *Eur. J. Oper. Res.* **2022**, *296*(2), 393–422. https://doi.org/10.1016/j.ejor.2021.04.032.

 doi:10.20944/preprints202603.1244.v1

22. Kim, N.; Montreuil, B.; Klibi, W.; Babai, M.Z. Network inventory deployment for responsive fulfillment. *Int. J. Prod. Econ.* **2023**, *255*, 108664. https://doi.org/10.1016/j.ijpe.2022.108664.
23. Kolyaei, M.; Zhang, L.; Blom, M.L. Inventory replenishment and fulfilment decisions for an omnichannel retailer: A reinforcement learning-based method. *Int. J. Prod. Res.* **2025**, *63*, 9571–9592. https://doi.org/10.1080/00207543.2025.2520596.
24. Li, R. Reinvent retail supply chain: Ship-from-store-to-store. *Prod. Oper. Manag.* **2020**, *29*, 1825–1836. https://doi.org/10.1111/poms.13195.
25. Liang, E.; Chang, K.-H. Digital twin-enabled nested Q-learning for multi-layer production planning and inventory control. *Int. J. Prod. Econ.* **2026**, in press. https://doi.org/10.1016/j.ijpe.2026.109979.
26. Liang, K.; Zhang, G.; Guo, J.; Li, W. An actor-critic hierarchical reinforcement learning model for course recommendation. *Electronics* **2023**, *12*, 4939. https://doi.org/10.3390/electronics12244939.
27. Liu, S.; Wang, J.; Wang, R.; Zhang, Y.; Song, Y.; Xing, L. Data-driven dynamic pricing and inventory management of an omnichannel retailer in an uncertain demand environment. *Expert Syst. Appl.* **2024**, *244*, 122948. https://doi.org/10.1016/j.eswa.2023.122948.
28. Liu, Y.; Yan, B.; Fan, J. Inventory strategy of fresh products for omnichannel supply chains. *J. Oper. Res. Soc.* **2024**, *75*, 673–688. https://doi.org/10.1080/01605682.2023.2198561.
29. Liu, X.; Hu, M.; Peng, Y.; Yang, Y. Multi-Agent Deep Reinforcement Learning for Multi-Echelon Inventory Management. *Prod. Oper. Manag.* **2025**, *34*, 1836–1856. https://doi.org/10.1177/10591478241305863.
30. Mahapatra, A.S.; Sengupta, S.; Dasgupta, A.; Sarkar, B.; Goswami, R.T. What is the impact of demand patterns on integrated online-offline and buy-online-pickup in-store (BOPS) retail in a smart supply chain management? *J. Retail. Consum. Serv.* **2025**, *82*, 104093. https://doi.org/10.1016/j.jretconser.2024.104093.
31. Neves-Moreira, F.; Amorim, P.S. Learning efficient in-store picking strategies to reduce customer encounters in omnichannel retail. *Int. J. Prod. Econ.* **2024**, *267*, 109074. https://doi.org/10.1016/j.ijpe.2023.109074.
32. Omar, H.; Klibi, W.; Babaï, M.Z.; Ducq, Y. Basket data-driven approach for omnichannel demand forecasting. *Int. J. Prod. Econ.* **2023**, *257*, 108748. https://doi.org/10.1016/j.ijpe.2022.108748.
33. Pateria, S.; Subagdja, B.; Tan, A.-H.; Quek, C. Hierarchical Reinforcement Learning: A Comprehensive Survey. *ACM Comput. Surv.* **2021**, *54*(5), Article 109, 35 pp. https://doi.org/10.1145/3453160.
34. Patterson, A.; Neumann, S.; White, M.; White, A. Empirical design in reinforcement learning. *J. Mach. Learn. Res.* **2024**, *25*(318), 1–63. Available online: https://jmlr.org/papers/v25/23-0183.html.
35. Rolf, B., Jackson, I., Müller, M., Lang, S., Reggelin, T., & Ivanov, D. (2023). A review on reinforcement learning algorithms and applications in supply chain management. *Int. J. Prod. Res.*, *61*(20), 7151–7179. https://doi.org/10.1080/00207543.2022.2140221.
36. Roosta, S.; Sadjadi, S.J.; Makui, A. Dynamic pricing modeling and inventory management in omnichannel retail using quantum decision theory and reinforcement learning. *PLoS ONE* **2025**, *20*, e0333068. https://doi.org/10.1371/journal.pone.0333068.
37. Roosta, S.; Sadjadi, S.J.; Makui, A. Predicting customer loyalty in omnichannel retailing using purchase behavior, socio-cultural factors, and learning techniques. *PLoS ONE* **2025**, *20*, e0330338. https://doi.org/10.1371/journal.pone.0330338.
38. Roosta, S.; Sadjadi, S.J.; Makui, A. A dynamic multi-objective optimization framework for omnichannel retailing integrating customer loyalty, channel coordination, and reinforcement learning. *Knowl.-Based Syst.* **2026**, *334*, 115171. https://doi.org/10.1016/j.knosys.2025.115171.
39. Si, Z.; Ali, D.A.; Rosli, R.B.; Bhaumik, A.A.; Ghosh, A. omnichannel retail marketing effect evaluation framework integrating big data and artificial intelligence. *Edelweiss Appl. Sci. Technol.* **2025**, *9*, 568–583.
40. Sumiea, E.H.; Abdulkadir, S.J.; Alhussian, H.S.; Al-Selwi, S.M.; Alqushaibi, A.; Ragab, M.G.; Fati, S.M. Deep deterministic policy gradient algorithm: A systematic review. *Heliyon* **2024**, *10*(9), e30697. https://doi.org/10.1016/j.heliyon.2024.e30697.
41. Sutton, R. S.; Barto, A. G. *Reinforcement Learning: An Introduction*, 2nd ed.; MIT Press: Cambridge, MA, USA, 2018. http://incompleteideas.net/book/the-book-2nd.html.
42. Syntetos, A. A.; Boylan, J. E. The accuracy of intermittent demand estimates. *Int. J. Forecast.* **2005**, *21*(2), 303–314. https://doi.org/10.1016/j.ijforecast.2004.10.001.
43. Wang, X.; Xiao, Y.; Dou, Y. Reselling or hosting? Examining platform's co-opetition strategy with third-party sellers. *Int. J. Prod. Econ.* **2025**, *282*, 109520. https://doi.org/10.1016/j.ijpe.2025.109520.

 doi:10.20944/preprints202603.1244.v1

disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.