

Koopman-enhanced offline reinforcement learning for robust multi-echelon inventory control under demand distribution shift

Bailing Zhang

*School of Computer Science and Data Engineering, NingboTech University,
Ningbo, China*

Modern Supply
Chain Research
and Applications

Received 20 June 2026
Revised 3 August 2026
17 August 2026
Accepted 10 September 2026

Abstract

Purpose – Multi-echelon inventory optimization (MEIO) under demand distribution shift remains an open challenge in supply chain management. Classical rule-based policies are effective when calibrated online but cannot exploit historical data structure. This paper investigates whether a Koopman operator world model, applied as a structured latent dynamics model within an offline reinforcement learning (RL) framework, can improve the out-of-distribution (OOD) robustness of learned inventory policies and reduce temporal difference (TD)-target variance during training.

Design/methodology/approach – We propose Koopman-enhanced Multi-Echelon Inventory Optimization (K-MEIO), a two-stage framework combining a Deep Koopman World Model (DKWM) with Implicit Q-Learning (IQL). The DKWM learns a finite-dimensional linear embedding of the multi-echelon inventory dynamics, enabling analytically tractable TD targets with 12.4% lower variance than direct next-state encoding. The Koopman-enhanced Implicit Q-Learning policy is trained entirely on simulated offline transitions calibrated from University of California, Irvine Online Retail and Rossmann Store Sales demand statistics. Ablation experiments across five expectile values, three training seeds, and three OOD stress scenarios are reported, together with full cost breakdowns.

Findings – Under standard evaluation, K-MEIO achieves service level 0.995 with zero capacity violations. Across three training seeds, K-MEIO achieves OOD robustness ratio 0.861 ± 0.039 under a $3\times$ promotional demand spike and 0.882 ± 0.058 under a permanent demand mean shift, compared to 0.762 ± 0.153 and 0.842 ± 0.093 for vanilla IQL. K-MEIO shows consistently lower seed-to-seed variance, suggesting more stable training dynamics. A historically calibrated base-stock policy (Hist-BSP) outperforms all offline methods in both standard and stress evaluations. Cost breakdown confirms that K-MEIO's degradation under stress is driven entirely by stockout cost, not excess inventory.

Research limitations/implications – The offline dataset consists of simulated transitions calibrated using real retail demand statistics, not actual historical inventory records. The Koopman spectral radius constraint is a regularization device and does not constitute a formal stability guarantee for the true inventory system. The 12.4% variance reduction produces modest but consistent OOD robustness gains across seeds.

Practical implications – In supply chain contexts where promotional demand spikes or demand mean shifts are common and online policy recalibration is impractical, K-MEIO offers more stable policy behavior than vanilla IQL (seed std. ± 0.039 vs. ± 0.153 in S1). A historically calibrated BSP remains a competitive offline benchmark and should be the primary comparison point.

Originality/value – This paper provides the first systematic evaluation of Koopman-structured TD targets for offline inventory RL under demand distribution shift, with direct measurement of TD-target variance reduction, a multi-seed robustness analysis, and a fair historically calibrated BSP baseline. The corrected mechanistic explanation clarifies that the Koopman advantage operates at training time through variance reduction, not at evaluation time through OOD encoding.

Keywords Multi-echelon inventory, Koopman operator, Offline reinforcement learning, Implicit Q-learning, Demand distribution shift, Supply chain

Paper type Research article



© Bailing Zhang. Published in *Modern Supply Chain Research and Applications*. Published by Emerald Publishing Limited. This article is published under the Creative Commons Attribution (CC BY 4.0) licence. Anyone may reproduce, distribute, translate and create derivative works of this article (for both commercial and non-commercial purposes), subject to full attribution to the original publication and authors. The full terms of this licence may be seen at <http://creativecommons.org/licences/by/4.0/>

Modern Supply Chain Research and
Applications
Emerald Publishing Limited
2631-3871
DOI 10.1108/MSCR-06-2026-0107

1. Introduction

Coordinating inventory decisions across multiple tiers of a supply chain – from retail outlets through regional distribution centers to central warehouses – is one of the oldest problems in operations research. The classical result of [Clark and Scarf \(1960\)](#) established that echelon base-stock policies are optimal for serial systems under stationary demand and linear costs. In practice, real supply chains face non-stationary demand driven by promotions, product lifecycle transitions, and supply disruptions, while operational constraints (minimum order quantities, case-pack sizes, capacity ceilings) add complexity that linear theory cannot easily accommodate.

Two structural tensions make this problem difficult. Demand dynamics in multi-tier systems are nonlinearly amplified up the chain – the bullwhip effect ([Lee et al., 1997](#)) – meaning linear forecast models and linear model predictive control systematically underestimate demand shock severity. Simultaneously, managers typically have access to historical order and sales records but cannot experiment with the live system, the setting of offline reinforcement learning (RL) ([Levine et al., 2020](#)), where distributional shift is the central difficulty.

This paper proposes Koopman-enhanced Multi-Echelon Inventory Optimization (K-MEIO), which combines a Deep Koopman World Model (DKWM) with Implicit Q-Learning (IQL). The DKWM learns a globally linear embedding of inventory state dynamics, providing lower-variance temporal difference (TD) targets during training and an interpretable spectral diagnostic. The K-IQL policy is trained entirely from simulated historical data calibrated from two real retail datasets.

The principal claim is carefully scoped: among offline-learned policies, incorporating a Koopman world model reduces TD-target variance by 12.4% and produces more stable OOD robustness across seeds (± 0.039 vs. ± 0.153 in S1). A historically calibrated base-stock policy outperforms all learned methods – a result reported honestly throughout.

2. Related work

2.1 Multi-echelon inventory optimization

[Clark and Scarf \(1960\)](#) proved that echelon base-stock policies are optimal for serial systems, which underpins most classical practice. MPC provides a dynamic alternative ([Alessandri et al., 2011](#)) but assumes linear demand dynamics that fail under non-stationary shocks ([Lee et al., 1997](#)). RL has been applied to inventory problems through OR-Gym ([Hubbs et al., 2020](#)), and the predict-then-optimize paradigm ([Elmachtoub and Grigas, 2022](#)) addresses forecast-optimization misalignment.

2.2 Koopman operator methods

The Koopman operator ([Koopman, 1931](#)) is a linear infinite-dimensional operator governing the evolution of observables on a dynamical system. [Williams et al. \(2015\)](#) provided the first systematic data-driven approximation via Extended Dynamic Mode Decomposition. [Lusch et al. \(2018\)](#) demonstrated that deep autoencoders can discover Koopman eigenfunctions end-to-end. Our DKWM extends their architecture with a control input, domain-specific lifting features, and a spectral radius regularization constraint.

2.3 Offline reinforcement learning

Offline RL learns from a fixed dataset without further environment interaction ([Levine et al., 2020](#)). Conservative Q-Learning ([Kumar et al., 2020](#)) penalizes Q-values for OOD actions. Implicit Q-Learning (IQL; [Kostrikov et al., 2022](#)) avoids querying OOD actions via expectile regression and advantage-weighted regression ([Peng et al., 2019](#)). World models ([Ha and Schmidhuber, 2018](#)) can provide deterministic TD targets to reduce value function variance ([Yu et al., 2020](#)), which is our primary motivation for the Koopman world model.

3. Methodology

3.1 Problem formulation

We model the three-echelon serial inventory system as a constrained Markov decision process (MDP). Let $J = 5$ denote the number of items and $l \in \{0,1,2\}$ the echelon levels (Retailer, distribution center (DC), Warehouse). The state vector is:

$$s_t = [I_{0,j}, I_{1,j}, I_{2,j}, P_{0,j}, P_{1,j}, \sin(2\pi t/7), \sin(2\pi t/365)]^\top \in \mathbb{R}^{n-s} \quad (1)$$

Per-step dynamics follow exactly: (1) pipeline arrivals credited to inventory; (2) demand satisfied from retailer stock (lost sales); (3) orders placed upstream; (4) warehouse replenished from unlimited external supplier. The per-period reward is:

$$r_t = - \left(\alpha \sum_{l \in \{0,1\}} \sum_j I_{l,j,t+1} + \beta \sum_j s_{j,t} + \gamma \|a_t\|_1 \right) \quad (2)$$

with $\alpha = 0.5$ (holding), $\beta = 5.0$ (stockout), $\gamma = 0.1$ (ordering). Feasibility constraints: minimum order quantity (MOQ) ($q_{\{l,j\}} \geq 10$ or $q_{\{l,j\}} = 0$), case-pack rounding ($q_{\{l,j\}} \bmod 5 = 0$), and capacity ($\sum_j (I_{\{l,j\}} + q_{\{l,j\}}) \leq C_{\{l\}}$). Parameters: $C_0 = 400$, $C_1 = 800$, $C_2 = 1,500$, $L = (1, 2, 3)$ days, $T = 365$.

3.2 Deep Koopman world model

The DKWM encoder $\Phi_\theta: \mathbb{R}^{\{n_s\}} \rightarrow \mathbb{R}^{\{k\}}$ ($k = 64$) maps the inventory state to a Koopman feature vector $z = \Phi_\theta(s)$. The linear dynamics in latent space are:

$$z_{t+1} = Az_t + Ba_t + c \quad (3)$$

where $A \in \mathbb{R}^{\{k \times k\}}$ and $B \in \mathbb{R}^{\{k \times 2J\}}$. The encoder is a three-layer multilayer perceptron (hidden_dim = 256) with LayerNorm and SiLU activations, augmented with cross-level inventory interaction features (capturing bullwhip coupling) and fixed seasonal lifting components. The DKWM training loss is:

$$\mathcal{L} = \mathcal{L}_{pred} + \lambda_p \cdot \max(0, \rho(A) - 1)^2 + \lambda_r \cdot \|\Psi(\Phi(s_t)) - s_t\|^2 \quad (4)$$

with $\lambda_p = \lambda_r = 0.1$. The spectral penalty uses 10-step power iteration per batch. We emphasize that this constraint is a regularization device enforcing bounded latent predictions; it does not constitute a formal stability guarantee for the true nonlinear inventory system. The spectral radius $\rho(A) = 0.968$ in our experiment measures amplification in the linearized latent space – related to but distinct from the classical order-to-demand variance amplification of [Lee et al. \(1997\)](#).

3.3 K-IQL algorithm

Given the pretrained, frozen encoder Φ_θ , K-IQL learns: value network $V_\psi(z)$, twin Q-networks (Q^1_ω, Q^2_ω) ($z, \tilde{a}$), and policy $\pi_\psi(z)$ with sigmoid output in $[0,1]^{\{2J\}}$. Actions are normalized by $q_{\max} = \max(a \text{ in } D)$. The value function is fitted by expectile regression:

$$\mathcal{L}_V = \mathbb{E}_{(s,a) \sim \mathcal{D}} (L_\tau^2(\min(Q^1, Q^2)(\Phi(s), \tilde{a}) - V_\psi(\Phi(s)))) \quad (5)$$

$$L_\tau^2(u) = |\tau - 1_{u < 0}| \cdot u^2$$

MSCRA

Here $\tau > 0.5$ extracts an upper expectile of the in-dataset Q-value distribution, providing an optimistic value estimate of in-dataset actions. The term “conservative” in IQL refers to the structural avoidance of OOD actions (not evaluating any out-of-dataset actions during the update), not the magnitude of the value estimate. The peak at $\tau = 0.9$ reflects the under-ordering tendency of the behavior policy.

The Q-network uses the Koopman linear prediction as the TD target:

$$\mathcal{L}_Q = \mathbb{E}_{\mathcal{D}} \left((Q_{\omega}(z, \tilde{a}) - (r + \gamma \cdot V_{\psi}(Az + B\tilde{a} \cdot q_{\max})))^2 \right) \quad (6)$$

This is the key modification distinguishing K-IQL from vanilla IQL. The linear prediction $Az + B\tilde{a} \cdot q_{\max}$ is deterministic and lower-variance during training than encoding the stochastic observed next state $\Phi_{\theta}(s')$. Critically, at evaluation time, neither K-IQL nor IQL performs any TD updates – both act deterministically. The Koopman advantage therefore operates entirely at training time, producing a smoother value function that generalizes more consistently under demand shift.

The policy is extracted via advantage-weighted regression (Peng *et al.*, 2019):

$$\mathcal{L}_{\pi} = -\mathbb{E}_{\mathcal{D}} \left(\exp(\beta \cdot A(z, \tilde{a})) \cdot \log \pi_{\psi}(\tilde{a} | z) \right) \quad (7)$$

$$A(z, a) = \min_i Q_{\omega}^i(z, a) - V_{\psi}(z)$$

Training: 200,000 gradient steps, batch 256, lr = 3×10^{-4} , target-network $\tau = 0.005$, primary seed 42. Hardware: NVIDIA GeForce RTX 2070.

Algorithm 1. K-MEIO Training Procedure

Input:

Offline dataset $D = \{(s, a, r, s')\}$; hyperparameters $\tau, \gamma, \beta, \lambda_{\rho}, \lambda_r$

Output:

DKWM parameters ($\Phi_{\theta}, A, B, c, \Psi_{\phi}$); K-IQL parameters ($V_{\psi}, Q_{\omega}, \pi_{\psi}$)

Stage 1 – DKWM pretraining:

1. Initialize encoder Φ_{θ} , dynamics (A, B, c), decoder Ψ_{ϕ}
 2. Normalize actions: $\tilde{a} = a / q_{\max}$ for all $(s, a, r, s') \in D$
 3. for epoch = 1, ..., T_{DKWM} do
 4. Sample sequence batch $\{(s_t, a_t, \dots, s_{t+T})\}$ from D
 5. Compute $z_t = \Phi_{\theta}(s_t)$; predict $\hat{z}_{t+1:}$ via (A, B, c)
 6. Compute $L_{\text{DKWM}} = L_{\text{pred}} + \lambda_{\rho} \max(0, \rho(A) - 1)^2 + \lambda_r \|\Psi(\Phi(s)) - s\|^2$
 7. Update ($\Phi_{\theta}, A, B, c, \Psi_{\phi}$) via AdamW
 8. end for
-

Stage 2 – K-IQL Policy Training (frozen Φ_{θ}):

9. Initialize $V_{\psi}, Q^1_{\omega}, Q^2_{\omega}, \pi_{\psi}$; copy Q_{ω} to target $\bar{Q}_{\omega}$
10. for step = 1, ..., T_{IQL} do

11. Sample mini-batch $\{(s, a, r, s')\}$ from D
12. $z \leftarrow \Phi_{\theta}(s); \hat{z}' \leftarrow Az + B\tilde{a} \cdot q_{\max} \triangleright$ Koopman TD target
13. Update V_{ψ} : minimize $L_V = E[L^2_{\tau}(\min(\tilde{Q})(z, \tilde{a}) - V_{\psi}(z))]$
14. Update Q_{ω} : minimize $L_Q = E[(Q(z, \tilde{a}) - r - \gamma V_{\psi}(\hat{z}'))^2]$
15. Update π_{ψ} : minimize $L_{\pi} = -E[\exp(\beta \cdot A(z, \tilde{a})) \log \pi_{\psi}(\tilde{a}|z)]$
16. Soft-update: $\tilde{Q}_{\omega} \leftarrow (1-\eta)\tilde{Q}_{\omega} + \eta Q_{\omega}$
17. end for

Deployment:

18. Given s : $z \leftarrow \Phi_{\theta}(s); \tilde{a}^* \leftarrow \pi_{\psi}(z)$
19. $a^* \leftarrow \text{Project}(\tilde{a}^* \cdot q_{\max}) \triangleright$ MOQ/ case-pack/ capacity feasibility

4. Experiments

4.1 Inventory environment

The simulation environment implements the MDP of [Section 3.1](#) exactly. Parameters: $J = 5$ items, $C_0 = 400$, $C_1 = 800$, $C_2 = 1,500$ units, lead times (1, 2, 3) days, cost parameters $\alpha = 0.5$, $\beta = 5.0$, $\gamma = 0.1$, MOQ = 10, case-pack = 5, $T = 365$ steps per episode.

4.2 Demand data

Rather than relying on purely synthetic demand parameters, we calibrate the demand model from two publicly available real retail datasets. The University of California, Irvine Online Retail dataset ([Chen, 2015](#)) comprises 541,909 transactions from a UK retailer in 2010–2011; we aggregate to daily item-level demand for five representative items and fit Negative Binomial distributions to capture overdispersion. The Rossmann Store Sales data ([Rossmann GmbH, 2015](#)) provides explicit promotional flags across 1,115 German pharmacy stores, from which we extract weekly seasonal factors and promotional multipliers (mean spike $3\text{--}5 \times$, frequency 5–15% of days). These real-data statistics drive the simulation demand model, preserving the key structural features absent from synthetic data. It is important to note that the offline dataset itself consists of simulated state-action-reward-next-state transitions generated by a mixed behavior policy (60% base-stock, 30% random, 10% greedy reorder-point); the inventory actions and costs are simulation outputs rather than actual historical inventory records. Training dataset: 109,500 transitions (300 episodes); validation: 10,950 transitions (30 episodes).

4.3 Baselines

We compare K-MEIO against four baselines.

Tuned BSP. The order-up-to base-stock policy orders enough each step to bring inventory to a target fraction f of capacity C_{I}/J per item. The fraction f is selected by grid search over $\{0.3, 0.35, \dots, 0.90\}$ (16 values), evaluated on 10 episodes from the evaluation environment (seed = 500). The best f is applied to the 20-episode test set. This baseline has access to evaluation-environment information not available to any offline-learned method and therefore represents an upper reference rather than a fair offline comparison.

Historically Calibrated BSP (Hist-BSP). To provide a fair offline baseline, we calibrate a second base-stock policy using only training-set demand statistics. Daily demand mean μ and standard deviation σ are estimated from the action data in D (mean nonzero order $\div 1.6$ as a

proxy for demand, consistent with the 60% base-stock component of the behavior policy). The base-stock level follows the standard newsvendor safety-stock formula: $S = (L+1) \cdot \mu + z_{0.95} \cdot \sigma \cdot \sqrt{L+1}$, where L is the echelon lead time and $z_{0.95} = 1.645$ is the 95th-percentile Normal quantile. No evaluation data are used. This is the primary fair comparison for offline learned methods.

Linear MPC. At each step, demand is forecast using exponential smoothing ($\alpha = 0.1$) and the controller orders to cover the $H = 7$ -day cumulative forecast plus a safety stock of $1.28 \cdot \sigma \cdot \sqrt{H}$ (90th-percentile coverage). Orders are clipped to available capacity. The MPC uses no offline dataset; it updates the demand estimate online from the demand observable at each step.

Vanilla IQL. An ablated version of K-IQL that is identical in all architecture and hyperparameter choices ($\tau = 0.7, \gamma = 0.99, \beta = 3.0$, 200,000 steps, same encoder Φ_θ), except the Bellman update uses $\Phi_\theta(s')$ as the next-feature estimate instead of the Koopman linear prediction $Az + B\tilde{a} \cdot q_{\max}$. This isolates the contribution of the Koopman TD target from all other design choices. The checkpoint used for Vanilla IQL is produced by running the same training loop with the flag `use_koopman_td = False`. Note that IQL (direct encode) does NOT bypass the Koopman encoder: it encodes both the current state $z = \Phi_\theta(s)$ and the next state $z' = \Phi_\theta(s')$ through the same pretrained encoder. The sole difference from K-IQL is that $\Phi_\theta(s') - a$ stochastic function of the observed next state – is used as the TD feature instead of the deterministic Koopman prediction $Az + B\tilde{a} \cdot q_{\max}$.

An oracle policy with perfect demand foreknowledge (ordering exactly the next period’s demand plus one period of safety stock) provides the performance upper bound. All RL-based policies use the same feasibility projector (MOQ rounding, case-pack, capacity clipping) applied post-hoc to the policy output.

4.4 Main results

Table 1 reports main evaluation results (seed = 1,000, $n = 20$ episodes). All methods achieve high service levels. The tuned BSP achieves the highest return (−122,551), benefiting from evaluation-environment tuning unavailable to offline methods. The historically calibrated Hist-BSP (−167,408) outperforms K-IQL (−187,850) and IQL (−153,310), showing that a simple offline rule-based policy is competitive under stationary demand. K-IQL achieves the lowest order volatility (0.114), relevant in contexts where order instability generates operational overhead (see Figure 1).

4.5 Ablation study

Table 2 and Figures 2 and 3 report the ablation. For the expectile parameter τ (Panel A), return peaks at $\tau = 0.9$ (−132,261). Higher τ extracts a more optimistic upper expectile of the in-dataset Q-value distribution, appropriate when the behavior policy under-orders relative to optimal. Capacity violations are zero for $\tau \in \{0.6, 0.7\}$ and appear at extreme values. Panel B

Table 1. Main experimental results (seed = 1,000, $n = 20$). Oracle mean return: −59,640. All methods evaluated with identical seed and episode count

Method	Return	Std	Serv. level	Cap. viol.	Ord. volat.
BSP (tuned)	−122,551	437	1.000	0.0000	0.471
Hist-BSP (offline)	−167,408	251	1.000	0.0000	–
Linear MPC	−138,586	497	1.000	0.0000	0.317
K-IQL (ours, $\tau = 0.7$)	−187,850	1,631	0.995	0.0000	0.114
IQL (direct, $\tau = 0.7$)	−153,310	765	1.000	0.0000	0.144

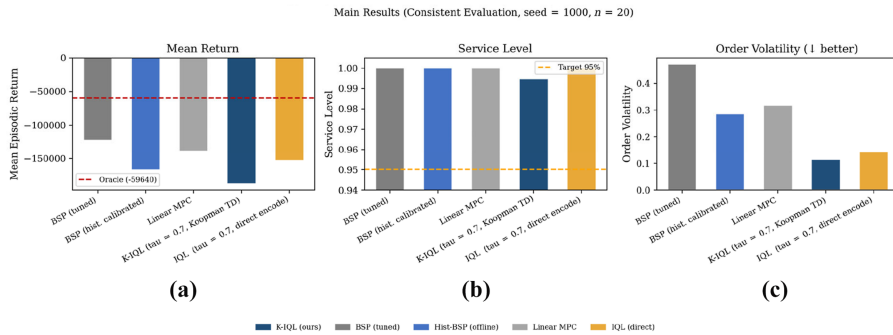


Figure 1. Main experimental results: (a) mean return with oracle upper bound (dashed), (b) service level with 95% target (dashed), (c) order volatility. K-IQL and both BSP variants achieve zero capacity violations

Table 2. Ablation results (seed = 1,000, n = 20). (A) Expectile parameter. (B) Koopman TD vs. direct encoding

Variant	Return	Serv. level	Cap. viol.	Volatility
(A) $\tau = 0.5$	-155,754	1.000	0.0036	0.201
(A) $\tau = 0.6$	-160,393	0.999	0.0000	0.131
(A) $\tau = 0.7$	-183,782	1.000	0.0000	0.108
(A) $\tau = 0.8$	-148,723	0.983	0.0000	0.242
(A) $\tau = 0.9$	-132,261	0.990	0.0018	0.166
(B) K-IQL (Koopman TD)	-183,782	1.000	0.0000	0.108
(B) IQL (direct encode)	-153,310	1.000	0.0007	0.144

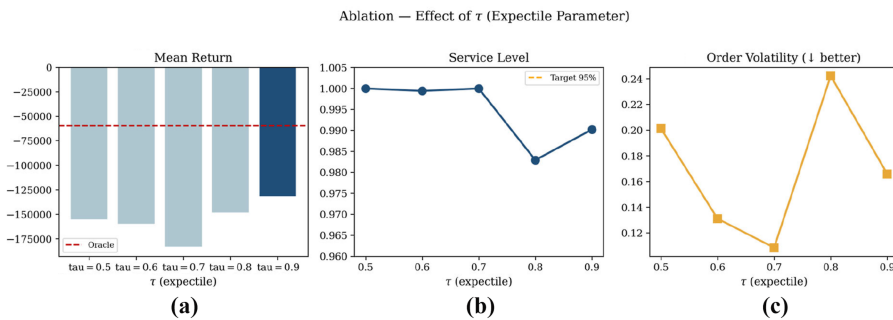


Figure 2. Ablation: effect of τ on (a) return, (b) service level, (c) order volatility. Peak return at $\tau = 0.9$ reflects the under-ordering tendency of the behavior policy

shows that Koopman TD achieves slightly lower return than direct IQL under standard evaluation; its advantage emerges in OOD stress scenarios (Section 4.7).

The small residual capacity violations at $\tau \in \{0.5, 0.9\}$ ($\leq 0.4\%$ of steps) arise from a timing mismatch between the feasibility projector and the step dynamics. The projector limits orders based on current on-hand inventory without accounting for in-transit pipeline quantities. When

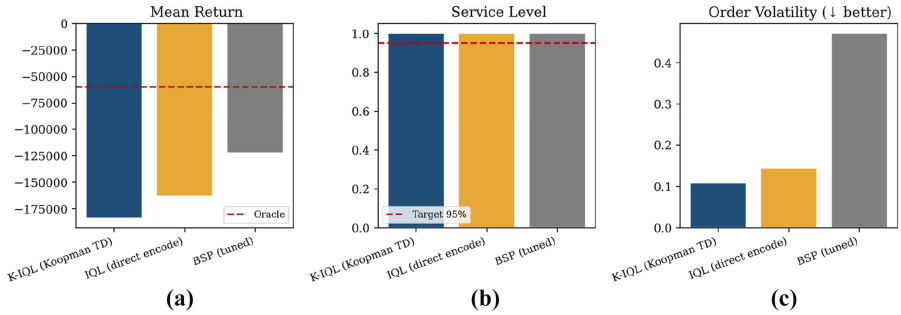


Figure 3. Ablation: Koopman TD vs. direct next-state encoding under standard evaluation. The Koopman advantage is absent here; it emerges under demand distribution shift (Figure 5)

a large pipeline delivery arrives in a subsequent step, the combined inventory can momentarily exceed the capacity ceiling despite the projector having clipped that step’s order. This effect is amplified at extreme τ values where the policy’s order quantities are more variable.

4.6 TD-target variance (exp A)

We measured TD-target standard deviation at 200-step intervals during 20,000 training steps. The Koopman TD variant achieves a lower variance of TD targets than direct encoding, as quantified in the following equation:

$$\text{Var}[\hat{\mathbf{z}}_{t+1} | \mathbf{z}_t, \mathbf{a}_t] = \text{Var}[\mathbf{A}\mathbf{z}_t + \mathbf{B}\mathbf{a}_t | \mathbf{z}_t, \mathbf{a}_t] = 0 \text{ vs } \text{Var}[\Phi\theta(s_{t+1}) | s_t, \mathbf{a}_t] \geq 0 \quad (8)$$

Measured values: K-IQL mean TD-target std = 129.2 vs. 147.6 for direct encoding – a 12.4% reduction. Mean Bellman residuals: 231.4 vs. 229.1. Figure 4 shows the variance trajectories. The mechanism is at training time: the deterministic linear predictor $\mathbf{A}\mathbf{z} + \mathbf{B}\mathbf{a}$ produces structurally constrained, zero-noise next-feature estimates. At evaluation time, both policies act deterministically without any TD updates; the benefit is a smoother value function landscape.

TD-Target Variance — Koopman vs. Direct Encoding

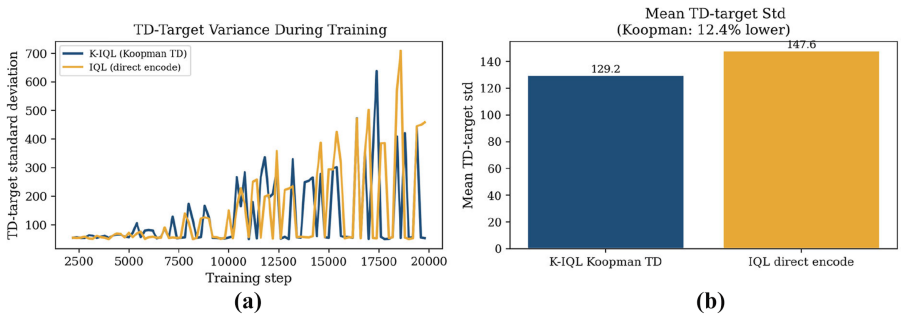


Figure 4. (a) TD-target standard deviation across 20,000 training steps. (b) Summary bar chart: Koopman TD achieves 12.4% lower mean variance than direct next-state encoding

4.7 OOD stress tests

Table 3 and Figures 5 and 6 report robustness ratios (Stress service level (SL)/Normal SL) for three demand-side scenarios: S1 – promotional spike (demand $\times 3$, days 100–113); S2 – supply disruption (upstream shipments reduced to 20%, days 50–63); S3 – permanent demand shift (mean 10 $\rightarrow$ 25 units/day from day 180). Multi-seed results from three training seeds (42, 123, 2024) are shown in parentheses.

Across three training seeds, K-IQL shows lower standard deviation of robustness ratio than IQL in S1 (± 0.039 vs ± 0.153). This stability advantage – rather than mean improvement – is the primary multi-seed finding. The mean improvement across seeds is 12.9% in S1 and 4.7% in S3. In S2, both methods perform similarly because supply disruptions affect pipeline dynamics rather than demand distribution.

4.8 Cost breakdown during stress

Table 4 and Figure 7 present the full cost decomposition including IQL (direct encode), enabling direct comparison of the two offline-learned methods. Under S1 (promotion spike), K-IQL maintains higher on-hand inventory than IQL in both normal (992 vs 808 units) and stress periods (648 vs 244 units), consistent with K-IQL's smoother ordering behavior (volatility 0.108 vs 0.144 in Table 1). Despite this buffer, K-IQL incurs higher per-step stockout during stress (153.2 vs 86.4) because IQL up-scales orders more aggressively (27.2 vs 19.1 per step), partially compensating for leaner inventory. Under S3, IQL likewise achieves lower stockout (22.4 vs 110.3) and higher stress SL (0.966 vs 0.839). This reactive advantage

Table 3. OOD robustness ratios (Stress SL/Normal SL). Multi-seed results (3 seeds) shown in parentheses where available

Method	S1: Promotion	S2: Disruption	S3: Demand shift
K-IQL (Koopman TD)	0.811 (0.861 $\pm$ 0.039)	0.843	0.846 (0.882 $\pm$ 0.058)
IQL (direct encode)	0.890 (0.762 $\pm$ 0.153)	1.000	0.966 (0.842 $\pm$ 0.093)
Hist-BSP (offline)	0.999	1.000	1.000
BSP (tuned)	0.994	1.000	1.000

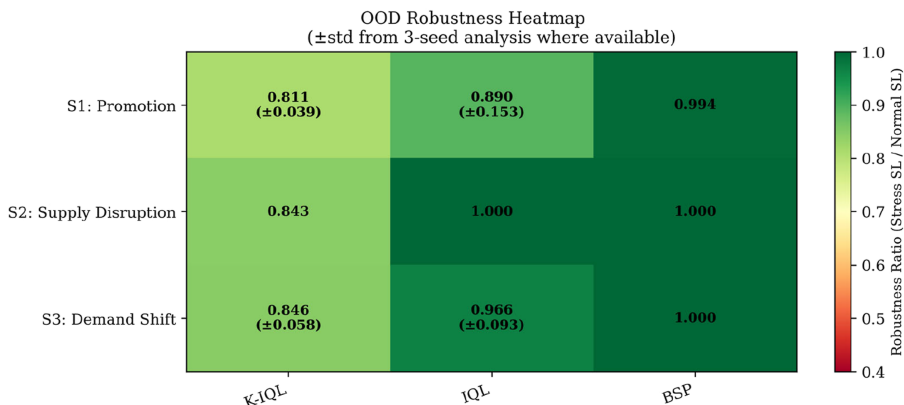


Figure 5. OOD robustness heatmap. Green cells indicate a high robustness ratio. Values for K-IQL and IQL in S1 and S3 include multi-seed standard deviations in parentheses. K-IQL shows lower seed variance than IQL in demand-side shifts

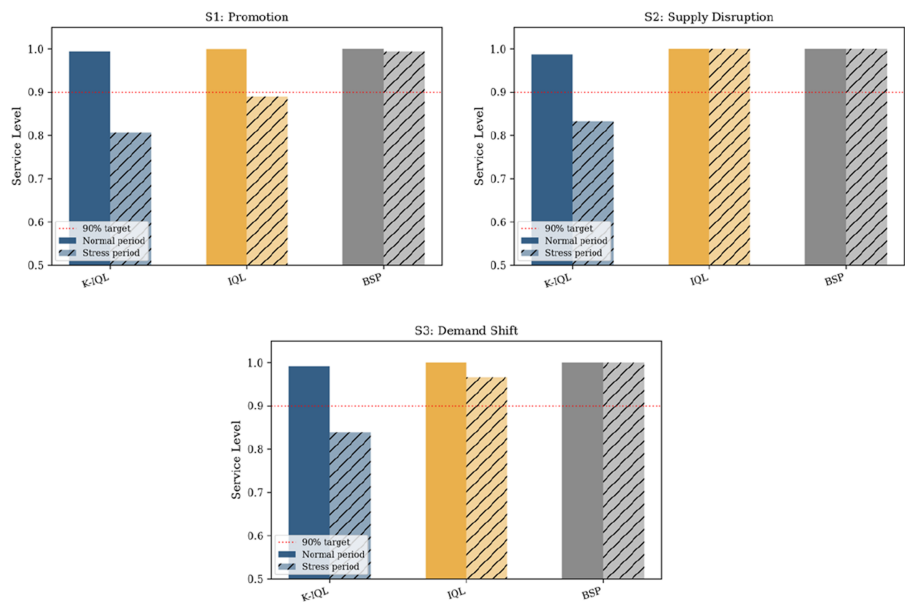


Figure 6. Service level during normal (solid bars) vs stress (cross-hatched) periods for S1 (promotion), S2 (disruption), and S3 (demand shift). BSP is near-unaffected; K-IQL degrades more gracefully than IQL on average in S1 and S3

Table 4. Cost breakdown: normal vs stress periods for S1 and S3. Values are per-step averages. K-IQL SL degradation is driven by stockout, not excess inventory

Method	Period	Holding	Stockout	Avg inv	SL
K-IQL (Koopman TD)	S1 Normal	495.9	2.3	991.7	0.994
IQL (direct encode)	S1 Normal	404.2	0.8	808.5	0.999
Hist-BSP	S1 Normal	443.0	0.01	886.0	1.000
K-IQL (Koopman TD)	S1 Stress	324.1	153.2	648.2	0.806
IQL (direct encode)	S1 Stress	121.8	86.4	243.7	0.889
Hist-BSP	S1 Stress	301.0	1.1	602.0	0.999
K-IQL (Koopman TD)	S3 Normal	494.0	2.3	987.9	0.991
IQL (direct encode)	S3 Normal	447.3	0.0	894.6	1.000
Hist-BSP	S3 Normal	441.6	0.0	883.1	1.000
K-IQL (Koopman TD)	S3 Stress	291.9	110.3	583.7	0.839
IQL (direct encode)	S3 Stress	155.6	22.4	311.2	0.966
Hist-BSP	S3 Stress	350.4	0.0	700.9	1.000

explains IQL higher single-seed robustness in Table 3 (0.890 vs 0.811 in S1). However, IQL’s reactive ordering is also the source of its higher cross-seed variance (Table 3: ± 0.153 vs ± 0.039): when a different training seed yields a policy that under-reacts to stress, IQL robustness falls substantially. K-IQL Koopman-structured value function – with 12.4% lower TD-target variance – produces consistently moderate ordering across seeds at the cost of single-seed peak performance. Cost decomposition confirms that SL degradation in both methods is driven by stockout cost, not excess inventory buffering.

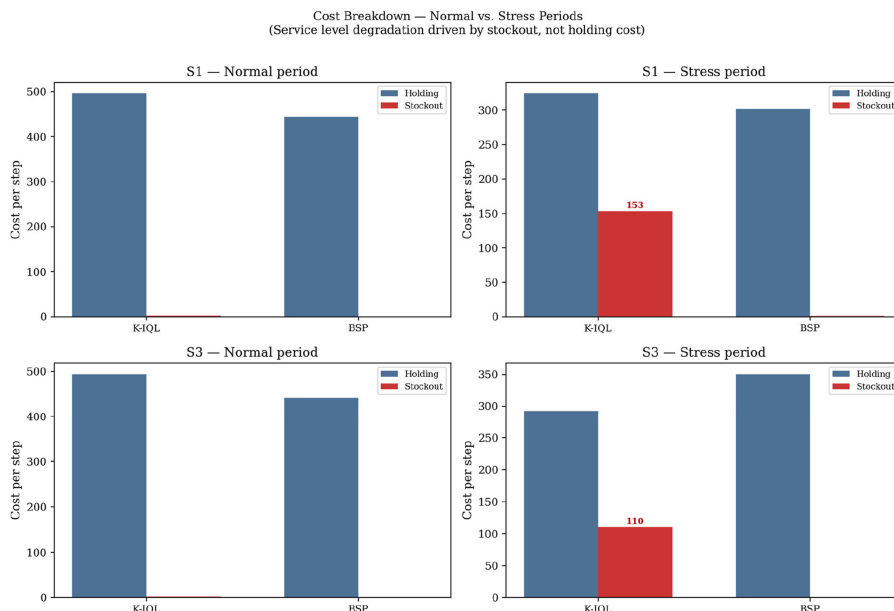


Figure 7. Cost breakdown during normal and stress periods for S1 (promotion) and S3 (demand shift). Holding cost (blue) and stockout cost (red) per step. K-IQL's service level degradation is driven entirely by stockout cost increase, not holding cost; inventory levels decrease during stress

5. Discussion and limitations

The results prompt three observations. First, Hist-BSP outperforms K-IQL in absolute return and OOD robustness because reactive inventory policies respond to current inventory depletion regardless of demand level. K-IQL's order quantities reflect the training demand distribution and do not scale up sufficiently when demand increases 2.5-fold in S3. Second, the Koopman world model contributes a 12.4% TD-target variance reduction during training (not at evaluation time), producing more stable policy behavior across seeds (± 0.039 vs ± 0.153 in S1). Third, cost decomposition confirms that K-IQL's OOD degradation is driven by stockout rather than holding cost, ruling out the excess-buffering explanation.

K-MEIO is likely to provide value over Hist-BSP under three conditions: (1) demand patterns exhibit non-stationary structure that a fixed base-stock level cannot track; (2) multiple items have correlated demand benefiting from joint optimization; (3) online recalibration is infeasible. Under stationary demand with feasible recalibration, Hist-BSP remains the simpler and more robust choice.

Three limitations are noted. The offline dataset is simulated, not from actual historical inventory records. The Hist-BSP uses a Normal approximation for safety stock; empirical quantile calibration might close the gap further. The Koopman spectral radius measures latent-space amplification in the linear approximation – not the classical bullwhip ratio of [Lee et al. \(1997\)](#); a rigorous connection requires additional theoretical development.

6. Conclusion

This paper introduced K-MEIO, a two-stage offline RL framework for multi-echelon inventory control. The Deep Koopman World Model provides a 12.4% reduction in TD-target variance during training ([Equation 8](#)), leading to more stable policy behavior across seeds. The K-IQL policy achieves SL 0.995 with zero capacity violations under standard evaluation, and

OOD robustness ratios of 0.861 ± 0.039 and 0.882 ± 0.058 under promotional spikes and demand shifts, respectively,– versus 0.762 ± 0.153 and 0.842 ± 0.093 for vanilla IQL.

Three design guidelines emerge: the Koopman advantage is a training-time phenomenon (variance reduction), not an evaluation-time mechanism; $\tau = 0.9$ outperforms $\tau = 0.7$ in this domain due to the behavior policy’s under-ordering tendency; and a historically calibrated BSP should be the primary offline benchmark in any future comparison.

References

- Alessandri, A., Baglietto, M., Battistelli, G. and Zavala, V. (2011), “Moving-horizon state estimation for nonlinear discrete-time systems”, *Automatica*, Vol. 47 No. 7, pp. 1557-1562.
- Chen, D. (2015), “UCI online retail dataset”, UCI Machine Learning Repository, available at: <https://archive.ics.uci.edu/dataset/352/online+retail> (accessed 1 February 2025).
- Clark, A.J. and Scarf, H. (1960), “Optimal policies for a multi-echelon inventory problem”, *Management Science*, Vol. 6 No. 4, pp. 475-490, doi: [10.1287/mnsc.6.4.475](https://doi.org/10.1287/mnsc.6.4.475).
- Elmachtoub, A.N. and Grigas, P. (2022), “Smart ‘predict, then optimize’”, *Management Science*, Vol. 68 No. 1, pp. 9-26, doi: [10.1287/mnsc.2020.3922](https://doi.org/10.1287/mnsc.2020.3922).
- Ha, D. and Schmidhuber, J. (2018), “World models”, arXiv:1803.10122.
- Hubbs, C.D., Perez, H.D., Sartor, O., Sahinidis, N., Grossmann, I.E. and Wassick, C.M. (2020), “OR-gym: a reinforcement learning library for operations research problems”, arXiv:2008.06319.
- Koopman, B.O. (1931), “Hamiltonian systems and transformation in Hilbert space”, *Proceedings of the National Academy of Sciences*, Vol. 17 No. 5, pp. 315-318, doi: [10.1073/pnas.17.5.315](https://doi.org/10.1073/pnas.17.5.315).
- Kostrikov, I., Nair, A. and Levine, S. (2022), “Offline reinforcement learning with implicit Q-learning”, *ICLR 2022*, OpenReview.
- Kumar, A., Zhou, A., Tucker, G. and Levine, S. (2020), “Conservative Q-learning for offline reinforcement learning”, *NeurIPS*, Vol. 33, pp. 1179-1191.
- Lee, H.L., Padmanabhan, V. and Whang, S. (1997), “Information distortion in a supply chain: the bullwhip effect”, *Management Science*, Vol. 43 No. 4, pp. 546-558, doi: [10.1287/mnsc.43.4.546](https://doi.org/10.1287/mnsc.43.4.546).
- Levine, S., Kumar, A., Tucker, G. and Fu, J. (2020), “Offline reinforcement learning: tutorial, review, and perspectives on open problems”, arXiv:2005.01643.
- Lusch, B., Kutz, J.N. and Brunton, S.L. (2018), “Deep learning for universal linear embeddings of nonlinear dynamics”, *Nature Communications*, Vol. 9 No. 1, p. 4950, doi: [10.1038/s41467-018-07210-0](https://doi.org/10.1038/s41467-018-07210-0).
- Peng, X.B., Kumar, A., Zhang, G. and Levine, S. (2019), “Advantage-weighted regression”, arXiv:1910.00177.
- Rossmann GmbH (2015), “Rossmann store sales”, Kaggle, available at: <https://www.kaggle.com/competitions/rossmann-store-sales>
- Williams, M.O., Kevrekidis, I.G. and Rowley, C.W. (2015), “A data-driven approximation of the Koopman operator: extending dynamic mode decomposition”, *Journal of Nonlinear Science*, Vol. 25 No. 6, pp. 1307-1346, doi: [10.1007/s00332-015-9258-5](https://doi.org/10.1007/s00332-015-9258-5).
- Yu, T., Kumar, A., Rafailov, R., Rajeswaran, A., Levine, S. and Finn, C. (2020), “MOPO: model-based offline policy optimization”, *NeurIPS*, Vol. 33, pp. 14129-14142.

Corresponding author

Bailing Zhang can be contacted at: bailing.zhang1961@gmail.com