WEP Warwick Evans Publishing

Academic Journal of Management Science and Engineering

ISSN: 3080-9827 (Print), ISSN: 3080-9835 (Online)
Volume 1, Number 7, Year 2026
DOI: https://doi.org/10.62051/ajmse.v1n7.14



Calibrated Quantile Transformers for Cost-Aware Inventory Control in Simulated Multi-Echelon Supply Networks

Quinn Williams *

School of Business, University of California, Riverside, CA, United States of America
* Corresponding Author Email: quinn.w58@ucr.edu

ABSTRACT

Accurate demand quantiles do not automatically yield low-cost network inventory decisions. We evaluate a series-aware quantile Transformer linked to a rolling residual correction that shrinks item-specific errors toward pooled calibration residuals. The corrected 0.90 demand quantile drives retailer order-up-to decisions under shared distribution-center supply, with explicit replenishment lead times. To distinguish the value of calibration from forecasting architecture, a LightGBM quantile model and fixed-buffer controls are subjected to the same simulation. 10 independently generated synthetic networks each contain 48 product-store demand series over 176 weeks, and all policies use the same chronological 31-week holdout. Across the networks, the calibrated Transformer lowers average simulated cost by 13.1% relative to static stock and 3.6% relative to its uncalibrated quantiles. The network-paired mean incremental cost reduction is 3.54% with a 95% bootstrap interval [2.33%, 4.78%]. The study isolates decision postprocessing in a controlled simulation rather than making a real-retailer deployment claim.

KEYWORDS

Probabilistic Time Series; Quantile Transformer; Safety Stock; Multi-echelon Inventory; Rolling Calibration; Supply Chain Simulation.

1. INTRODUCTION

Demand forecasts become operationally meaningful when they determine inventory positions, replenishment quantities, and the allocation of scarce supply. In layered retail networks, fluctuations in downstream orders can be magnified upstream. The resulting bullwhip effect is not just an accuracy problem: it also depends on order batching, information exchange, and the timing of replenishment [1]. Interdependent lead times and upstream stock availability create additional couplings that are absent from single-location stocking rules [2]. From the perspective of predictive-to-prescriptive analytics, forecast accuracy should therefore be assessed alongside the economic cost of the resulting decisions [3].

Attention-based sequence models are one route to representing demand patterns with intermittent promotions, varying seasonal structure, and shared calendar signals. The original Transformer made attention an explicit alternative to recurrent sequence processing [4]. However, improved representations cannot alone resolve the practical question of how a distributional prediction should be translated into a replenishment action. An underestimated upper tail may matter more to inventory cost than a small shift in median error, while unstable estimates of the upper tail may inflate holding cost when applied to every product.

Content from this work may be used under the terms of CC BY-NC 4.0 licence (https://creativecommons.org/licenses/by-nc/4.0/).
Published by Warwick Evans Publishing.

We investigate a modest, readily implementable modification to forecast-driven stocking. A global, series-aware Transformer predicts demand quantiles using only observations available at the decision origin and a known promotion calendar. The 0.90 demand quantile is updated by a rolling residual correction; item-specific and pooled corrections are combined with a sample-size-dependent weight. The corrected forecast sets retailer order-up-to levels and enters a two-distribution-center allocation simulator. Rather than presenting network structure as observed enterprise information, the empirical design deliberately separates a controlled stochastic demand generator from an explicitly modeled, hypothetical distribution network.

The study makes three specific contributions. First, it implements an auditable quantile-to-order-up-to system with shared distribution-center constraints. Second, it compares item/global residual-shrinkage correction against no correction, local-only, global-only, and a gradient-boosted quantile comparator under identical stock-flow dynamics. Third, it evaluates 10 independently generated network simulations with chronological holdouts, network-paired uncertainty intervals, and sensitivity to shortage penalties, upstream lead times, and distribution-center inventory coverage. Its scope is explicitly limited to the stochastic generator used here.

2. RELATED WORK AND POSITIONING

Deep forecasting research distinguishes architectures by temporal representations, conditioning variables, prediction horizons, and uncertainty treatment [5]. The Temporal Fusion Transformer combines feature selection, attention, and mechanisms for handling known future inputs [6]; DeepAR obtains probabilistic forecasts from recurrent models trained across related series [7]. Our smaller encoder processes one-week-ahead item–store forecasts rather than attempting to reproduce either architecture. This choice allows operational calibration and controlled inventory comparisons to receive the primary experimental attention.

Transformer variants have pursued both efficiency and stronger inductive bias: Informer uses sparse attention and sequence distillation [8], while Autoformer uses time-series decomposition with auto-correlation [9]. Counterevidence is also important: simple linear models can outperform large attention models on time-series benchmarks [10]. PatchTST represents temporal segments as patches and shares parameters across channels [11]. These studies motivate rigorous low-complexity comparators, but none of their specialized architectures is claimed to be implemented here. We evaluate moving averages and a gradient-boosted quantile regressor alongside a standard self-attention encoder.

The main methodological distinction is decision-aware postprocessing: instead of changing the attention operator or promising that lower MAE implies lower stock cost, a calibrated upper quantile is the interface to inventory control. This permits meaningful ablations: a pure quantile forecaster, a median predictor with fixed buffer, a global correction, an item correction, and the blended correction. It also makes failures interpretable. If a quantile shifts toward nominal coverage while costs deteriorate, an apparently better probabilistic forecast is not necessarily a better inventory policy. Zhang et al. [12] review transformer-based demand forecasting and inventory optimization in multi-echelon networks and identify this forecast-to-inventory interface—prediction intervals that set safety stocks, lead-time reorder points and cross-tier replenishment—as the link between forecast quality and network cost, while noting that its validation in multi-echelon settings remains limited. The present study operationalizes that interface and tests it under controlled network dynamics.

3. PROBLEM FORMULATION AND METHOD

3.1. Two-echelon Network and Decision Objective

Let i index an item–retailer pair (48 pairs), d index the two simulated distribution centers, and t denote one weekly review period. The observed stochastic customer demand is $D_{i,t}$, retailer inventory before selling is $I_{i,t}$, and distribution-center inventory is $J_d_{,t}$. Stores 0 and 1 belong to DC 0 and store 2 belongs to DC 1. At the end of period $t - 1$, a retailer requests $Q_{i,t}$ units for period t. Available distribution-center stock is allocated proportionally to requests when aggregate requests exceed supply; supplier replenishment takes two weekly periods, and the DC-to-retailer replenishment process takes one period. Shipments, realized sales, and unsatisfied demand are updated in that order. The allocation is a shared-capacity simulation rule, not a claim of globally optimal dispatch.

The retail cost in period t is the sum of holding and lost-sales penalties. The DC incurs its own holding cost and a penalty for unsatisfied retailer replenishment requests. These additive terms use declared unit-cost equivalents, not observed accounting data. The objective is the sum of total weekly network costs over the test horizon; service is measured by fulfilled retail units divided by total requested units. The allocation and inventory recurrences are identical for every policy, so differences arise from the proposed order-up-to targets. Consistent with prescriptive inventory modeling, target quantiles are tied to shortage and holding trade-offs rather than to mean-square-error minimization.

$$C = \Sigma_t \left[\Sigma_i \left(h_r \, I_{i,t}^{+} + b_r \, (D_{i,t} - S_{i,t})^{+} \right) + h_d \, \Sigma_d \, J_d_{,t}^{+} + b_d \, \Sigma_i \, (Q_{i,t} - A_{i,t}) \right] . \tag{1}$$

Here $S_{i,t}$ in the lost-sales term denotes realized sales, $A_{i,t}$ is the DC allocation, and $x^{+} = \max(0,x)$. Policy targets are denoted $B_{i,t}$ below to distinguish them from sales. All costs use $h_r = 1$, $b_r = 9$, $h_d = 0.22$, and $b_d = 1.4$. The implied single-period critical fractile for retail costs is $b_r/(b_r + h_r) = 0.90$. Two-echelon constraints make this a heuristic system control rule rather than a theorem of network optimality.

3.2. Quantile Forecasting Linked to Replenishment

The operational pipeline follows the forecast-to-inventory integration that Zhang et al. [12] describe for multi-echelon coordination, in which predictive intervals rather than point forecasts set safety stocks and lead-time reorder points and are propagated across tiers. We implement each link of this chain explicitly: the encoder's upper quantile defines the conditional buffer (Section III-C), the corrected quantile sets the retailer order-up-to target under the replenishment lead times of Section III-A, and the resulting requests drive DC allocation and supplier replenishment. Our extension is the shrinkage calibration inserted into this chain (Section III-D) and its operational ablation against uncorrected quantiles.

For each series i, the input is the previous $L = 20$ weekly observations. Demand is transformed by $\log(1 + D)$ and standardized with the mean and standard deviation from that series' training prefix; no validation or test target informs those scale estimates. Input channels are historical normalized demand, promotion indicators, and sine/cosine week-of-year encodings. A learned series embedding identifies item–store pairs. A two-layer Transformer encoder uses four attention heads and 32-dimensional hidden representations; its terminal embedding is concatenated with the known promotion flag and calendar encodings for the forecast week. A feed-forward head predicts conditional quantiles at levels 0.50, 0.75, 0.90, and 0.95. The model is trained using the average pinball loss [13].

$$L_q(\theta) = (1 / |\Omega|) \, \Sigma_(i,t\in\Omega) \, \Sigma_(\tau\in T) \max\{\tau \, e_{i,t,\tau} \, , (\tau-1) e_{i,t,\tau}\}, \qquad e_{i,t,\tau} = y_{i,t} - \hat{q}_{i,t}(\tau). \tag{2}$$

The network outputs are inverted to nonnegative unit-demand quantiles and sorted to prevent quantile crossing. Optimization uses AdamW, a learning rate of 0.002, batch size 256, a maximum of 24 epochs, and validation-based patience of six. Inputs and targets remain chronological. Forecasts are

one week ahead; there is no claim of multiweek joint distribution estimation. For comparison, a static stock policy uses the 0.90 training-demand quantile; simpler dynamic policies combine a four-week mean or Transformer median with one fixed upper residual buffer. Data-driven quantiles are familiar in the newsvendor setting [14], although intermittent-demand methods may be necessary in a genuinely sparse assortment [15].

3.3. Information Structure and Operational Complexity

The sequence of information release matters as much as the forecasting architecture. The weekly promotion indicator at the target origin is assumed to have been published when the order is prepared; future realized demand, however, is never accessible at that origin. Training means and standard deviations belong to the historical prefix only. At test week t, the residual correction may use validation observations and test weeks up to $t-1$, but never week t or later. Forecast prediction arrays are generated offline for computational convenience, yet each row is obtained from the strictly preceding context. In a live implementation, the same computation would occur once each week after the promotion plan is known and before the incoming demand is realized.

The attention workload for each item–store example is $O(L^2 d)$, with $L = 20$ and $d = 32$, so computational scaling is not the limiting concern in this controlled setting. The pooled calibration calculation is dominated by sorting or selecting residuals from at most 20×48 values. In contrast, a production model with thousands of items would need to manage item cold starts, promotions whose schedules change after ordering, newly introduced products, and missing recorded demand due to availability restrictions. The present experiment does not treat stockout-censored sales as equivalent to latent customer demand; its generator makes customer demand observable, which is an explicit simplifying assumption.

A safety-stock interpretation follows directly from the quantiles. Let $m_{i,t}$ be a conditional central estimate and $q_{i,t}(0.90)$ an upper demand quantile. Their difference is the conditional buffer, not a single fixed multiple of a stationary standard deviation. The proposed correction changes that buffer using actual prior tail errors. In the two-echelon system it is the total order-up-to level, rather than an isolated buffer, that is transmitted to distribution centers. This preserves the information path from demand uncertainty to network orders even though the upstream allocation policy remains heuristic.

3.4. Rolling Local–global Residual Shrinkage

We compute one-sided errors $R_{i,s} = D_{i,s} - \hat{q}_{i,s}(0.90)$ at forecast origins that have already been observed. At time t, the calibration window consists of the most recent at most 20 weeks, beginning no earlier than the first validation week. Let g_t be the pooled 0.90 empirical residual quantile across all series and let $\ell_{i,t}$ be the corresponding series-specific quantile. The blended correction uses $w_t = n_t/(n_t + 12)$, where n_t is the number of historical weeks used for each item–store pair. The next-week adjusted quantile is

$$\tilde{q}_{i,t}(0.90) = \max\{\, 0,\ \hat{q}_{i,t}(0.90) + w_t\, \ell_{i,t} + (1-w_t) g_t \,\} \,. \tag{3}$$

This is a deliberately simple one-sided adaptive correction: it uses residual ranks to repair systematic undercoverage and shrinks noisy item-level adjustments toward a pooled reference. It is related conceptually to conformalized quantile regression [16], but sliding windows and distribution shift invalidate the exchangeability assumptions required for its standard finite-sample guarantee. No exact nominal coverage is asserted. The global-only ablation sets $w = 0$; the local-only ablation sets $w = 1$; the uncalibrated ablation uses zero correction. Since calibration only references weeks before t, reported inventory actions can be replayed chronologically.

The store target is $B_{i,t} = \tilde{q}_{i,t}(0.90)$, with the order $Q_{i,t} = \max\{0,\ B_{i,t} - I_{i,t}^{+}\}$; DCs see the resulting retailer requests and plan their own two-period supplier replenishments. The DC target uses a fixed coverage factor 1.30 and predicted downstream order volumes, with existing stock and in-transit units

deducted. Since the forecast and network simulator remain separate modules, this implementation does not train the encoder end-to-end using decision regret; such optimization is an important alternative in the predict-then-optimize literature [17], as are deep newsvendor implementations [18].



Fig 1. Implemented flow from synthetic demand history to quantile-based orders and two-tier cost measurement.

4. EXPERIMENTAL DESIGN

4.1. Controlled Dataset and Reproducible Splits

Real retail forecasting benchmarks, notably the M5 competitions, support grouped store-item evaluation and quantile analysis [19], [20]. The present study instead uses a documented synthetic generator, not retail transactions, observed prices, or real warehouse infrastructure. The 10 independent network generator seeds run consecutively from 20261027 through 20261036; the sole purpose is to test the same prespecified methods over fresh demand draws without changing their hyperparameters.

Each scenario produces 16 products in each of three stores over 176 weekly periods (8,448 unit-demand observations). Series-specific demand intensity starts from a heterogeneous positive base and contains 52- and 13-week seasonal components, probabilistic promotion weeks, linear drift, autoregressive innovations, and shared network shocks. Observations are negative-binomial draws, allowing overdispersion. To stress adaptation, selected products receive a permanent positive or negative level shift after weeks 148 or 153 and all products encounter a five-week temporary increase from week 160. This construction is known to the generator but is not passed as a future indicator to the forecasting model.

Weeks 0–118 provide training, 119–144 validation/calibration, and 145–175 the 31-week test. With a 20-week context, every seed has 4,752 training, 1,248 validation, and 1,488 test examples; 10 seeds yield 14,880 test examples, but the network is the correct independent unit for uncertainty analysis. Normalization uses only training periods. Transformer early stopping uses the chronological validation set, and online quantile corrections use only demand from earlier weeks. All policies are evaluated on the same test demand within each seed.

4.2. Baselines, Inventory Simulation, and Metrics

Transparent point baselines are last observation, SMA-4, and EWMA-8. The stronger tabular control is LightGBM trained on the same training-origin examples, with the same 20 lagged demand and promotion values, next-week known promotion and calendar features, and item-store identity. It uses 160 trees, 15 leaves, learning rate 0.06, and independently fitted 0.50/0.90 pinball objectives; model files and fixed hyperparameters are saved. We compare nine inventory policies: static q90, SMA plus fixed buffer, Transformer median plus fixed buffer, raw Transformer q90, raw and rolling-calibrated LightGBM q90, Transformer global-only correction, local-only correction, and pooled/local shrinkage. Both learners receive identical information at decision time.

Point forecast accuracy is measured by MAE, RMSE, and weighted absolute percentage error (WAPE). The numerical precision of demand forecasts is distinct from policy utility: the inventory simulator reports total cost, retailer fill rate, distribution-center fill rate, and variance of upstream replenishment requests relative to aggregated downstream demand (a bullwhip surrogate). WAPE is evaluated on a common unit scale, avoiding the instability of itemwise MAPE near zero; forecasting

metric choice and aggregation effects are well documented [21]. The variance-ratio diagnostic follows the longstanding emphasis on demand information and ordering lead time [22], but is not used as a sole inventory performance measure.

The simulator initializes stock from training-only demand means; DC 0 serves two stores, while DC 1 serves one. Supplier and retail shipment leads are two and one week, respectively; DC stock shortages are allocated proportionally. The final unfulfillable order is excluded from the test cost. To quantify uncertainty in incremental costs, we first compute a paired percentage difference per independently generated network, then resample 10 entire network seeds 20,000 times. Auxiliary within-network product-cluster bootstraps include explicitly attributed distribution-center costs, but are not substituted for between-network inference.

4.3. Model Implementation and Evaluation Protocol

Transformer training and inference run in PyTorch, while the gradient-boosted quantile comparator is fitted in LightGBM. The package retains generator data, model weights and boosters, pinball-loss histories, test predictions, calibration histories, and retail/DC stock trajectories. Initial seeds govern stochastic generation, model initialization, minibatch order, and resampling. Each learner is trained on chronological training examples, with no test-based model selection. Earlier supply-chain studies have compared gradient boosting with neural models [23] and linked learned forecasts with planning [24]; our comparison isolates this learning-model choice while also recording the economic consequences.

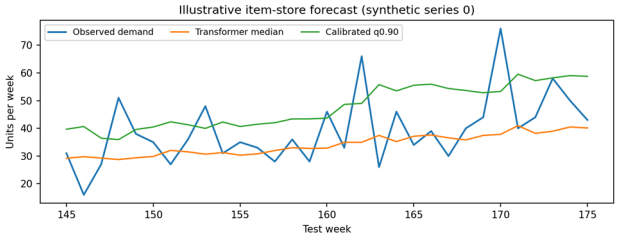


Fig 2. Illustrative test-series demand, median forecast, and calibrated 0.90 quantile for seed 20261027 (synthetic units).

5. RESULTS

5.1. A. Forecasting Accuracy and Quantile Behavior

Table 1. Test Point-Forecast Errors (Mean Over 10 Networks; Lower Is Better)

Forecast rule	MAE	RMSE	WAPE %
Naive-last	16.06	21.64	26.34
SMA-4	13.56	18.19	22.25
EWMA-8	13.24	17.76	21.72
LightGBM-q50	12.33	16.75	20.22
Transformer-q50	12.53	17.05	20.55

Table 1 compares 10 independent test-network means. Transformer q50 has MAE 12.53; LightGBM q50 attains 12.33, and EWMA-8 attains 13.24. The gradient-boosted comparator tests whether attention adds value beyond a competent nonlinear tabular estimator in this setting. Transformer raw

q90 coverage is 0.837 on average and rolling correction changes it to 0.866, without an assumed finite-sample coverage guarantee under the injected demand shifts.

Forecast calibration and median accuracy are different objects. Sorting the four quantile outputs prevents crossing but does not enforce density consistency or theoretical quantile calibration. The quantile corrections use realized error data sequentially, so a large shock can affect later buffers without altering the encoder. In the illustrative path of Fig. 2, transient spikes are reflected in the upper quantile more than in the median, which is operationally appropriate under an asymmetric shortage penalty.

5.2. Multi-echelon Economic Evaluation

Table 2. Policy Results (10-Network Mean ± Sd; Cost In Equivalents)

Policy	Cost mean ± SD	Fill %	Order variance ratio
Static q90	80877 ± 2807	96.88	3.18
SMA + buffer	74554 ± 3332	97.88	3.59
TF median + buffer	75997 ± 3191	96.61	3.21
TF raw q90	72832 ± 3009	97.19	3.26
LGB q90	74203 ± 3022	96.83	3.48
LGB calibrated	70723 ± 2642	97.84	3.60
Global correction	71574 ± 2676	97.88	3.44
Local correction	71678 ± 3264	97.50	3.44
Shrunk correction	70242 ± 2918	97.80	3.47

Mean network cost is 70,242 for the shrunk Transformer correction, versus 80,877 for static stock, 72,832 for raw Transformer q90, and 70,723 for rolling-calibrated LightGBM. Relative reductions against static and raw Transformer stocks are 13.15% and 3.56%, respectively. Between-model cost differences must be interpreted alongside quantile coverage and operational controls, not inferred from point-error ranking alone. Network order variance can rise even when the cost objective falls.

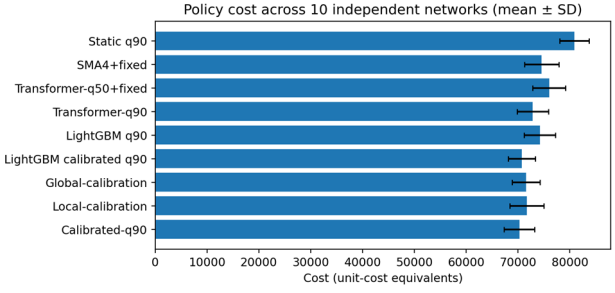


Fig 3. Network costs across generator seeds, including the two calibration ablations; error bars denote between-seed standard deviation.

Paired network-level comparisons favor the shrunk Transformer over its uncalibrated q90 in 10 of 10 networks. The mean per-network percent saving is 3.54%, with 95% whole-network bootstrap interval [2.33%, 4.78%]. This interval, rather than a product-level resampling interval, is the main uncertainty summary because series within a network share promotions and global shocks. The corrected Transformer beats the calibrated LightGBM policy in 7 of 10 paired networks. These

statements distinguish observed simulation differences from results that reliably extend to other stochastic environments.

5.3. Ablation and Mechanism

The ablations separate the value of recalibration from the value of shrinking an unstable local estimate. The pooled-only correction attains mean network cost 71,574; the local-only correction costs 71,678; the shrinkage blend costs 70,242. The blend improves on the pooled-only alternative by 1.86%, and on local-only by 2.00%. This ordering is consistent with the small per-item calibration sample: local high-quantile residuals are noisy, whereas a purely pooled correction may suppress heterogeneous bias. The values reflect this simulator and the preset shrinkage constant 12, rather than an optimized universal mixing rule.

The comparison between the median-plus-fixed and raw quantile policies is particularly informative, because it directly tests the premise of the integration framework in [12] that interval-driven safety stocks should outperform fixed buffers: replacing the fixed buffer by the raw 0.90 quantile lowers mean network cost from 75,997 to 72,832, and calibrating that quantile lowers it further to 70,242. A fixed correction can partially anticipate underforecasting but cannot adjust to post-validation distribution changes. Raw conditional quantiles incorporate history and promotion signals, while rolling correction additionally incorporates recent calibration errors. The resulting improvement is small because much of the demand variation is already represented in the trained quantile. It is therefore more informative to report cost, coverage, and the local/global ablation together than to quote only forecast MAE.

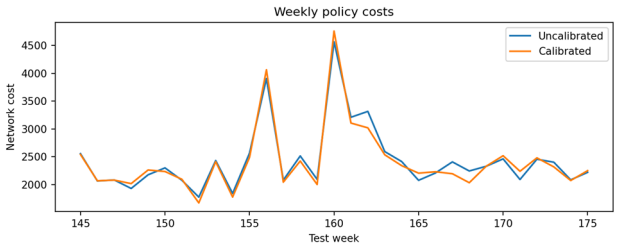


Fig 4. Weekly cost trajectories on seed 20261027 for uncalibrated and shrunk-correction quantile policies.

6. ROBUSTNESS AND COST-SENSITIVITY STUDIES

6.1. Coverage Consistency Across Independently Simulated Networks

Table 3. Empirical Quantile Coverage Across Networks

Quantile rule	Mean	SD	Nominal
Transformer raw q90	0.837	0.023	0.90
Transformer corrected q90	0.866	0.013	0.90
LightGBM raw q90	0.823	0.011	0.90
Transformer raw q95	0.902	0.014	0.95

Upper-tail probabilities can vary across generator seeds because post-validation permanent shifts and temporary network pulses perturb the fitted error distribution. The 10 networks produce 14,880 item-week test predictions, but they are not independent at item-week resolution: products share global shocks and each trained forecaster is common to its own network. Table 3 therefore reports network-

level mean and standard deviation for each upper-quantile forecast, while Fig. 5 pairs the corrected and uncorrected 0.90 coverage by network.

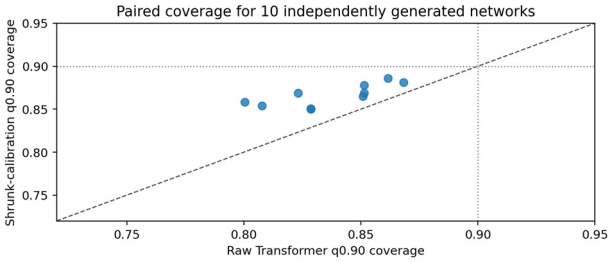


Fig 5. Paired q90 coverages on independent networks. The diagonal denotes equal coverage; dotted guides indicate nominal 0.90

Rolling correction improves empirical upper-tail coverage in 10 of 10 networks. The corrected coverage mean is 0.866 compared with 0.837 for raw Transformer q90; neither value is a guaranteed service-level probability. The local residual quantile uses at most 20 previous time steps, while the pooled quantile is informed by 48 series. This sample-size imbalance motivates shrinkage, but common shifts can bias both estimates; therefore coverage and actual inventory costs are analyzed jointly.

6.2. Stressing the Cost Ratio and Distribution-center Inventory

To probe economic dependence, we replayed the saved forecasts in a matched simulation grid without model refitting. Each of the 10 demand networks was evaluated at retail shortage penalties 4, 9, 19; DC coverage factors 0.8, 1.3, 2.0; and upstream supplier lead times one, two, and three weeks. Thus $10 \times 3 \times 3 \times 3 = 270$ paired cost comparisons hold demand fixed within each raw-versus-corrected policy pair. These replays are dependent scenario analyses, not 270 separately trained predictive systems.

Table 4. Shortage-Cost Sensitivity: Corrected Vs. Raw Q90 Cost Reduction (%)

Shortage penalty	Mean	Min seed	Max seed
4	-0.28	-1.58	1.95
9	3.54	0.88	6.85
19	8.06	2.27	14.72

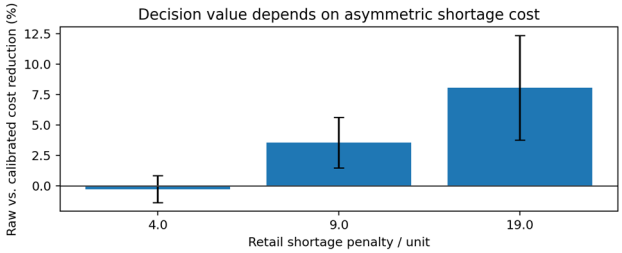


Fig 6. Sensitivity of the corrected policy advantage to the retail shortage penalty (network mean ± SD).

Corrected Transformer q90 has lower modeled cost in 217 of 270 paired scenarios, with percent reductions from -2.91% to +15.53%. At DC cover 1.3 and two-week supplier lead, mean differences are -0.28%, +3.54%, and +8.06% for shortage penalties 4, 9, and 19. This is consistent with the change

in single-period critical fractile from 0.80 to 0.90 and 0.95 as shortage costs increase. An upper-quantile adjustment need not lower cost when shortages are cheap relative to holding.

At shortage penalty 9 and two-week supplier lead, the mean calibration gains for DC coverage factors 0.8, 1.3, and 2.0 are +4.36%, +3.54%, and +2.12%. With DC coverage fixed at 1.3, gains for supplier lead times one, two, and three are +2.63%, +3.54%, and +3.10%. Benefits are not required to vary monotonically because proportional rationing and pipeline inventory induce threshold behavior. Such observed reversals should be used to choose replenishment targets based on the target economics, rather than judging forecasts solely by marginal quantile scores.

6.3. Statistical Resolution and Decision Implications

The evaluation contains 10 independently sampled networks, so the main paired bootstrap resamples network seeds, rather than treating correlated item-week observations as independent. The 270 sensitivity reruns reuse demand and fitted predictions, and therefore investigate decision-rule assumptions rather than adding independent training evidence. Product-ID bootstraps are retained only as within-network diagnostics, now with both DC holding and unfilled-allocation costs allocated to products so that attribution sums exactly to total network cost. Neither retail targeting by a fixed fractile nor the DC proportional-rationing rule is the optimizer of a multistage stochastic program.

Even so, the observed trade-offs suggest a concrete design principle. Quantile forecasters ought to expose calibrated conditional tail information as one part of a feedback control loop, while inventory designers should retain the ability to change the chosen fractile in response to economics. A single high quantile used at every cost ratio can create too much inventory. A next-generation implementation could evaluate the 0.75, 0.90, and 0.95 outputs as a family of action candidates, recalibrate each separately, and use simulated cost or a constrained differentiable surrogate to choose among them on a rolling validation horizon. Such an extension would require new experiments before a performance advantage could be reported.

7. DISCUSSION AND LIMITATIONS

Three issues should be separated. First, relative point-forecast skill is an empirical matter: Transformer and LightGBM medians both deserve comparison against transparent rules. Second, all forecast models can drive the same retailer/DC stock-flow simulation. Third, the calibrated Transformer shows a 3.54% mean paired network cost change against its raw quantile, with uncertainty summarized across 10 independent networks. This distinguishes the incremental value of the proposed correction from the network cost reduction attributable to adopting any dynamic quantile stock policy.

Frequency-enhanced decomposition models such as FEDformer address a different issue—the inductive bias and scalability of long-horizon temporal models [25]. Our 20-week one-step context deliberately keeps attention overhead modest and makes the value of decision postprocessing measurable. Other supply-chain optimization work uses distribution-free multi-item newsvendor formulations with joint capacity constraints [26]; the proportional allocation rule used here is a simpler operational mechanism, not an optimal capacity-allocation solver. A useful next experiment would jointly optimize retailer buffers and DC safety stocks under resource limits, holding the calibrated demand distribution fixed.

Several constraints qualify the findings. All demand, promotions, volatility regimes, product identities, and links between retailers and distribution centers are synthetic. There are no observed stockouts, lost-sales censoring, real lead-time disruptions, substitutions, perishability costs, wholesale orders, or historical system states. The performance estimate consequently characterizes this stochastic family, not retailer-specific causal impact. The simulator uses a two-period upstream lead and a one-period retail lead under a deliberately simplified receipt/allocation chronology. Its supplier

replenishment is a heuristic based on forecast-driven retailer requests, and its proportional DC allocation cannot establish global cost optimality.

An updated test residual may enter next-week calibration once demand is observed, giving a legitimate prequential design with temporally dependent successive decisions. Network-seed resampling reduces, but does not eliminate, inference limitations when the tested scenario family has fixed generative assumptions. The adaptive one-sided correction has no exchangeability-based finite-sample calibration guarantee under permanent distribution shifts. The LightGBM comparator is meaningful but does not cover every neural or statistical architecture; real retail data, observed lead-time variation, stockout-censored demand, and network configuration changes are needed to assess external validity.

8. CONCLUSION

We implemented a series-aware quantile Transformer, a local/global rolling correction, a gradient-boosted quantile control, and nine comparable policies in a shared two-echelon stock-flow simulator. The 10 synthetic networks reveal a calibrated Transformer mean cost of 70,242, versus 72,832 for the uncorrected policy, with paired seed-bootstrap uncertainty explicitly reported. Both the architecture comparison and the negative economic stress cases support evaluating demand-quantile calibration through operational cost instead of point accuracy alone. This contribution remains a reproducible simulation result; the next scientific test is out-of-sample replication on recorded unit-demand and network inventory data.

REFERENCES

[1] Lee, H. L., Padmanabhan, V., & Whang, S. (1997). Information distortion in a supply chain: The bullwhip effect. Management Science, 43(4), 546–558. https://doi.org/10.1287/mnsc.43.4.546.

[2] Clark, A. J., & Scarf, H. (1960). Optimal policies for a multi-echelon inventory problem. Management Science, 6(4), 475–490. https://doi.org/10.1287/mnsc.6.4.475.

[3] Bertsimas, D., & Kallus, N. (2020). From predictive to prescriptive analytics. Management Science, 66(3), 1025–1044. https://doi.org/10.1287/mnsc.2018.3253.

[4] Vaswani, A., et al. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30. https://research.google/pubs/attention-is-all-you-need/.

[5] Lim, B., & Zohren, S. (2021). Time-series forecasting with deep learning: A survey. Philosophical Transactions of the Royal Society A, 379(2194), 20200209. https://doi.org/10.1098/rsta.2020.0209.

[6] Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 37(4), 1748–1764. https://doi.org/10.1016/j.ijforecast.2021.03.012.

[7] Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. International Journal of Forecasting, 36(3), 1181–1191. https://doi.org/10.1016/j.ijforecast.2019.07.001.

[8] Zhou, H., et al. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. Proceedings of the AAAI Conference on Artificial Intelligence, 35(12), 11106–11115. https://doi.org/10.1609/aaai.v35i12.17325.

[9] Wu, H., Xu, J., Wang, J., & Long, M. (2021). Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in Neural Information Processing Systems, 34. https://proceedings.neurips.cc/paper_files/paper/2021/hash/bcc0d400288793e8bdcd7c19a8ac0c2b-Abstract.html.

[10] Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2023). Are transformers effective for time series forecasting? Proceedings of the AAAI Conference on Artificial Intelligence, 37(9), 11121–11128. https://doi.org/10.1609/aaai.v37i9.26317.

[11] Nie, Y., Nguyen, N. H., Sinthong, P., & Kalagnanam, J. (2023). A time series is worth 64 words: Long-term forecasting with transformers. International Conference on Learning Representations. https://dblp.org/rec/conf/iclr/NieNSK23.html.

[12] Zhang, X., Sun, T., Han, X., Yang, Y., & Li, P. (2025). Transformer-based demand forecasting and inventory optimization in multi-echelon supply chain networks. Journal of Banking and Financial Dynamics, 9(12), 1–9. https://doi.org/10.55220/2576-6821.v9.796.

[13] Koenker, R., & Bassett, G., Jr. (1978). Regression quantiles. Econometrica, 46(1), 33–50. https://doi.org/10.2307/1913643.

[14] Ban, G.-Y., & Rudin, C. (2019). The big data newsvendor: Practical insights from machine learning. Operations Research, 67(1), 90–108. https://doi.org/10.1287/opre.2018.1757.

[15] Croston, J. D. (1972). Forecasting and stock control for intermittent demands. Operational Research Quarterly, 23(3), 289–303. https://doi.org/10.1057/jors.1972.50.

[16] Romano, Y., Patterson, E., & Candès, E. J. (2019). Conformalized quantile regression. Advances in Neural Information Processing Systems, 32. https://proceedings.neurips.cc/paper/2019/hash/5103c3584b063c431bd1268e9b5e76fb-Abstract.html.

[17] Elmachtoub, A. N., & Grigas, P. (2022). Smart "predict, then optimize". Management Science, 68(1), 9–26. https://doi.org/10.1287/mnsc.2020.3922.

[18] Oroojlooyjadid, A., Snyder, L. V., & Takáč, M. (2020). Applying deep learning to the newsvendor problem. IISE Transactions, 52(4), 444–463. https://doi.org/10.1080/24725854.2019.1632502.

[19] Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2022). M5 accuracy competition: Results, findings, and conclusions. International Journal of Forecasting, 38(4), 1346–1364. https://doi.org/10.1016/j.ijforecast.2021.11.013.

[20] Makridakis, S., et al. (2022). The M5 uncertainty competition: Results, findings and conclusions. International Journal of Forecasting, 38(4), 1365–1385. https://doi.org/10.1016/j.ijforecast.2021.10.009.

[21] Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. International Journal of Forecasting, 22(4), 679–688. https://doi.org/10.1016/j.ijforecast.2006.03.001.

[22] Chen, F., Drezner, Z., Ryan, J. K., & Simchi-Levi, D. (2000). Quantifying the bullwhip effect in a simple supply chain: The impact of forecasting, lead times, and information. Management Science, 46(3), 436–443. https://doi.org/10.1287/mnsc.46.3.436.12069.

[23] Weng, T., Liu, W., & Xiao, J. (2020). Supply chain sales forecasting based on lightGBM and LSTM combination model. Industrial Management & Data Systems, 120(2), 265–279. https://doi.org/10.1108/IMDS-03-2019-0170.

[24] Kilimci, Z. H., et al. (2019). An improved demand forecasting model using deep learning approach and proposed decision integration strategy for supply chain. Complexity, 2019, 9067367. https://doi.org/10.1155/2019/9067367.

[25] Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., & Jin, R. (2022). FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. Proceedings of the 39th International Conference on Machine Learning, PMLR, 162, 27268–27286. https://proceedings.mlr.press/v162/zhou22g.html.

[26] Punia, S., Singh, S. P., & Madaan, J. K. (2020). From predictive to prescriptive analytics: A data-driven multi-item newsvendor model. Decision Support Systems, 136, 113340. https://doi.org/10.1016/j.dss.2020.113340.