SABREAGENT: LANGUAGE MODELS AT DESIGN TIME FOR LOST-SALES INVENTORY CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

SabreAgent uses a language model at design time to construct two components for lost-sales inventory control: a product-specific seasonal prior and a validation-selected capped base-stock policy family. During operation, statistical forecasting and inventory optimization use these frozen artifacts to determine orders, with zero language-model calls. We evaluate the approach on the 1,320 instances of InventoryBench. Under the benchmark's cost assumptions, the operations-research core draws on a zero-lead-time optimality result and a projected-inventory rule for positive deterministic lead times. The latter computes replenishment shortfalls by propagating inventory using sales along simulated demand paths. The seasonal prior adds forecast variants alongside the original forecaster, and the selected policy family handles stochastic lead times with order destruction. SabreAgent scores 0.6311, compared with 0.5380 for the strongest published baseline, and ranks first in all six benchmark cells. Ablations attribute most of the gain to the OR core. In the paired analysis, the seasonal component adds +0.0179 across the three real-data cells, and the search component adds +0.0230 across the two stochastic-lead-time cells. These results demonstrate how model-generated priors and policy structure can improve an OR controller through design-time use.

1 INTRODUCTION

InventoryBench gives each policy only five initial demand observations, while real instances also provide product descriptions (Baek et al., 2026). Effective replenishment requires combining these limited observations and seasonal cues with inventory costs and supply dynamics. Nine of the benchmark's ten published baselines consult a language model every period. Each resulting order commits inventory before its outcome is observed. Work on planning and self-correction identifies limitations of using generated reasoning to assess such decisions (Huang et al., 2024; Liu et al., 2025).

Design-time use lets external procedures inspect and evaluate a model-generated policy, prior, or representation before deployment. This division of labor underlies verifier-guided program search (Romera-Paredes et al., 2024; Liu et al., 2024; Ye et al., 2024; Novikov et al., 2025) and the broader "model proposes, verifier disposes" view (Kambhampati et al., 2024). Prior work in operations research has used language models to formulate and translate optimization problems (Ramamonjison et al., 2023; AhmadiTeshnizi et al., 2024; Xiao et al., 2024). SabreAgent applies this division of labor to inventory control, with statistical forecasting and optimization handling sequential decisions during operation. Figure 1 illustrates this division of labor.

Across InventoryBench's 1,320 lost-sales instances, all ten published baselines score below a hindsight diagnostic that chooses one constant order-up-to level per instance. This offline reference motivates an explicit treatment of inventory costs and dynamics.

SabreAgent builds its control core from the benchmark's cost structure. Orders have no purchase or setup cost and leftover inventory has no salvage value. At zero lead time, some optimal post-order inventory level is bounded above by $\max\{I_t, y_t^*\}$, where y_t^* is the myopic critical-fractile level. Under stationary demand, the myopic policy is optimal. For positive deterministic lead times, we project the committed pipeline through the lost-sales dynamics and price only the residual shortfall. A textbook inventory-position target treats demand that was already lost as if it had depleted the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

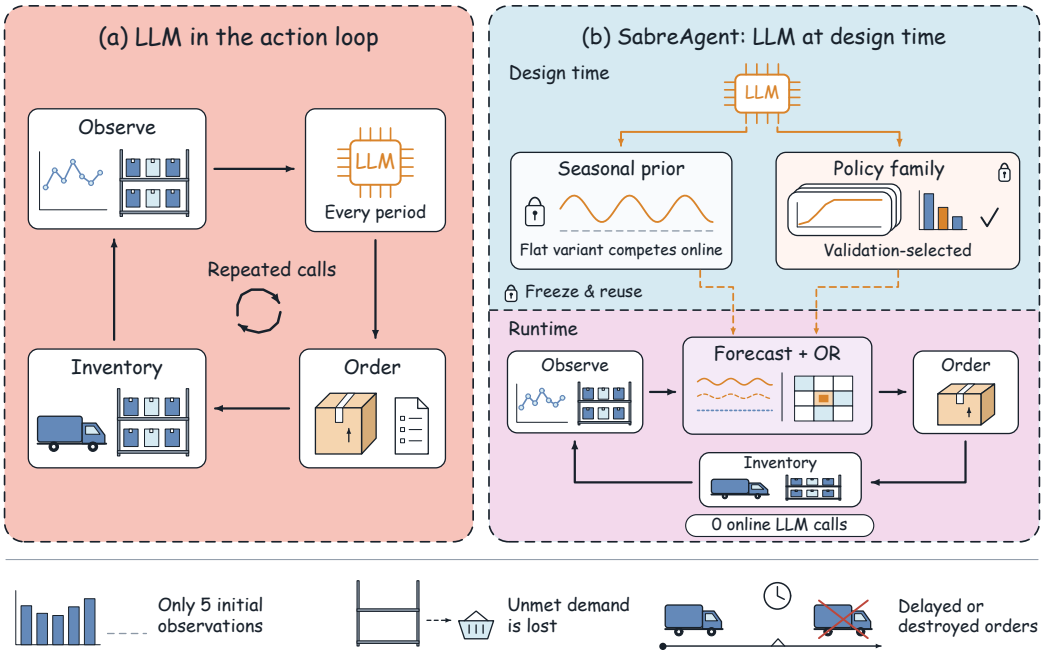


Figure 1: Two placements of language models in inventory control. (a) A model generates orders inside the decision loop. (b) SabreAgent uses a frozen seasonal prior alongside the forecaster without seasonal adjustment and a validation-selected policy family. Statistical forecasting and OR control generate orders without runtime language-model calls.

pipeline. For stochastic lead times, where an order may be destroyed, the policy instead evaluates a small family of replenishment rules using the same simulated demand and supply trajectories.

The language model supplies two additions to this core. We elicit a normalized seasonal shape from product descriptions and use it to construct additional forecast variants. The original forecaster without seasonal adjustment remains in the mixture; §3.3 gives the associated predictive-loss bounds. An automated research loop also proposes policy families for the lossy-supply cells. Scores on separately generated validation instances guide the selection of a capped base-stock family. Building on established capped base-stock policies in lost-sales control (Xin, 2021), we link the cap to destroyed-order catch-up bursts and combine design-time family selection with online, instance-specific parameter search.

The final policy scores 0.6311, versus 0.5380 for the strongest published baseline, and leads in every benchmark cell. The ablation separates the sources of this performance: the frozen OR core reaches 0.6180 on the batched evaluation path, and the two model-authored components add +0.0138 on that same path. Paired bootstrap intervals exclude zero for the seasonal component on real-data cells and the search component on stochastic-lead-time cells. A same-model online-adjustment control adds bounded per-period order changes to the complete policy. On its 124-instance subset, the real stochastic cell is the only cell with a positive estimate exceeding two standard errors.

The analysis and ablations identify the contributions of the cost-based control core, seasonal forecasting, and policy-family selection. Related work and proofs appear in the appendix.

2 Preliminaries

2.1 Control problem and evaluation

Each InventoryBench instance (Baek et al., 2026) tracks one product at one location for T weeks, with $T = 50$ for synthetic instances and 47 for real ones. Inventory starts at zero. The nonnegative demand path d_t is fixed by the instance, and unmet demand is lost (Zipkin, 2008a; Bijvank & Vis, 2011). In the benchmark's own timing the policy first observes on-hand stock I_t, the in-transit total X_t and the position $\mathrm{IP}_t = I_t + X_t$, all before that period's arrivals are booked. It then

places $q_t \in \mathbb{Z}_{\geq 0}$, scheduled for $t + L_t$ when $L_t < \infty$ and destroyed otherwise. Arrivals $A_t = \sum_{t' \leq t} q_{t'} \mathbf{1}\{L_{t'} < \infty,\ t' + L_{t'} = t\}$ are added to stock next, the sum including $t' = t$, so at $L = 0$ an order lands in the period that places it. Sales are $s_t = \min(d_t, I_t + A_t)$, and holding is charged on the ending inventory $I_t^{\text{end}} = I_t + A_t - s_t$, which also carries over:

$$I_{t+1} = I_t^{\text{end}}, \qquad R_t = p\, s_t - h\, I_t^{\text{end}}. \tag{1}$$

Costs are $(p, h) \in \{(1, 1), (4, 1), (19, 1)\}$, so the critical ratio $\rho = p/(p + h)$ is 0.50, 0.80 or 0.95, all three present in every cell.

The policy receives a partial observation of the system state. Initially it receives five dated demand observations, the promised lead time, p, h, and, for a real instance, a product description. In later periods it sees the date, I_t, X_t, and the previous demand, order, and arrival, then maps the resulting history $\mathcal{H}_t$ to q_t. In-transit inventory is only a total; its age profile and the current realization of L_t are hidden when the order is placed. Demand itself is uncensored, unlike the usual lost-sales setting in which stockouts conceal the upper tail and couple learning to control (Huh & Rusmevichientong, 2009). No exploration term is therefore needed for demand estimation.

Two regimes set L_t to 0 or 4. The stochastic regime uses $\{1, 2, 3, \infty\}$; $L_t = \infty$ denotes permanent order destruction. The benchmark fixes one lead-time sequence per cell and shares it across that cell's instances. A policy is told only a promised lead time of 2 and must learn survival from observed inventory movements. Because demand and supply paths are fixed during scoring, the expectation in the policy's planning objective is internal to its predictive model; the reported score is computed on the realized benchmark path. Crossing two data sources with three lead-time regimes gives the six cells in Table 7.

Each instance is scored by a normalized reward, and the headline number is its unweighted mean over the 1,320 instances:

$$\text{NR} = \max\left\{0,\ \frac{\sum_t R_t}{p \sum_t d_t}\right\}, \qquad \text{S} = \frac{1}{1{,}320} \sum_{i=1}^{1{,}320} \text{NR}_i. \tag{2}$$

The denominator is the reward from serving every unit immediately with no ending inventory. It is attainable with perfect demand information when $L = 0$. With positive lead time, however, an empty initial system cannot serve demand before its first possible arrival, and destroyed orders impose an additional loss. Thus $\text{NR} = 1$ is the ideal-service reference for normalization; §4.2 reports a supply-feasible clairvoyant diagnostic.

2.2 What the cost structure implies

Purchase cost, fixed ordering cost, and salvage value are all zero in the benchmark. Inventory can normally be valuable because it avoids a future purchase or setup cost (Scarf, 1960), or because it covers demand that no future order can reach (Arrow et al., 1951; Zipkin, 2000; Porteus, 2002). Under (1), a future order is free. At $L = 0$, it can also arrive in time to serve that period's demand. Carrying an extra unit therefore offers no saving on future orders and incurs h in each period it remains unsold. The analysis takes $p, h > 0$, orders unbounded above, and demand exogenous, nonnegative and independent across periods with known marginals.

Theorem 1 (zero-lead-time policy structure)**.** *Assume zero purchase, setup, and salvage costs and $L = 0$. Let y_t^* be a ρ-fractile of period-t demand. Then V_t is non-increasing, and some optimal post-order level is no greater than $\max\{I_t, y_t^*\}$. Lookahead may reduce the myopic order but cannot improve the value by increasing it. If the fractiles admit a non-decreasing selection over time—in particular, under stationary demand—then V_t is constant on $[0, y_t^*]$ and the myopic policy $y_t = \max\{I_t, y_t^*\}$ is optimal.*

The proof, integer-order extension, and a counterexample when fractiles fall over time appear in Appendix B. Classical work studies near-myopic behavior in broader inventory settings (Karlin, 1958; Morton, 1971; Zipkin, 2008b). The exact statement here uses instantaneous, free replenishment. Our positive lead-time heuristic applies the theorem's cost intuition through a projected-inventory surrogate.

Zero salvage also fixes the rollout boundary condition: inventory left after the last scored period has no terminal value. The deterministic rule below therefore prices the marginal service benefit of an order against the holding cost of its projected surplus, without an added continuation credit.

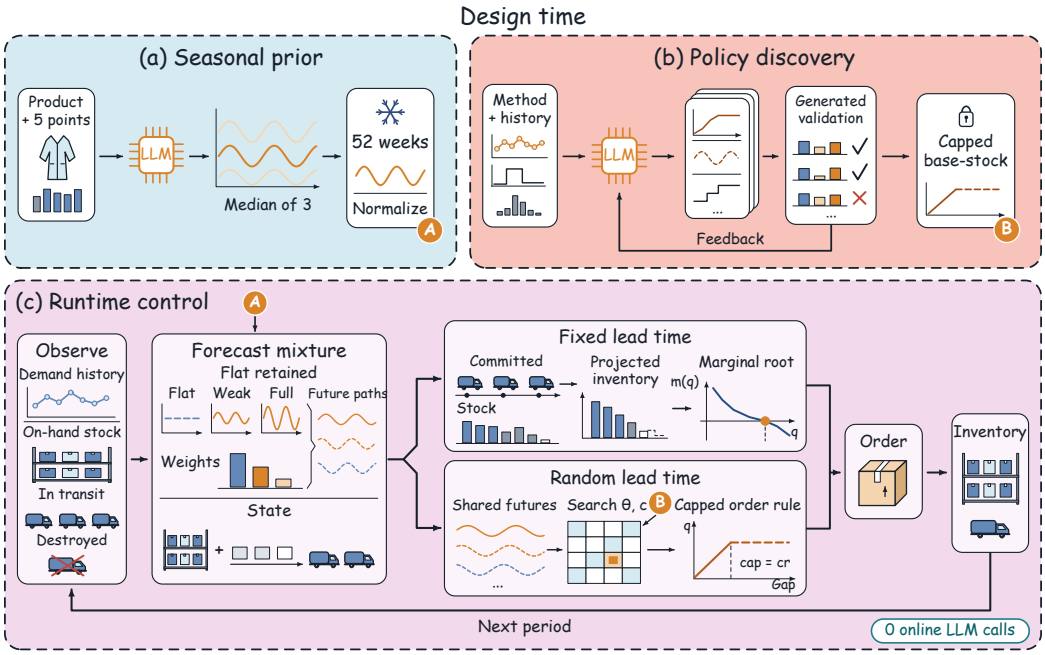


Figure 2: SabreAgent's two-stage architecture. (A) The frozen seasonal prior enters a forecast mixture that retains the flat variant; the associated bounds concern predictive log loss. (B) Validation selects the frozen capped base-stock family, whose numerical parameters are searched online. Fixed and stochastic lead times use projected-inventory control and rollouts on common demand and supply paths, respectively. Runtime uses no language-model calls.

3 SABREAGENT

3.1 DESIGN-TIME ARCHITECTURE

SabreAgent has three independently configurable components: a forecaster (§3.2), an optional seasonal-shape layer (§3.3), and the policy family searched when lead times are stochastic (§3.5). We designed the forecaster and the initial policy; DeepSeek-V4-Flash produced the other two artifacts before the final configuration was frozen. All language-model calls occur before deployment. During operation, the policy uses the frozen artifacts to update forecasts and search policy parameters online; a fixed random seed makes these computations reproducible. Restoring all components to their initial settings gives the frozen core, enabling component ablations within the same policy implementation (Appendix D).

The two model-authored components use different selection procedures. For policy search, an automated loop receives the current method, the validation protocol, and the full record of earlier successes and failures. It proposes one self-contained policy change per round. Candidates advance according to scores from the validation harness. Candidate scoring uses only generated validation instances separate from the benchmark test instances. Section 4.1 records the benchmark information available during development and the resulting limits on test independence.

Seasonal shapes are frozen per item before benchmark evaluation. The validation instances lack product text and therefore provide no held-out target for selecting these shapes. The shape layer adds forecast variants beside the unchanged flat variant; §3.3 states the corresponding log-loss bounds and their conditions.

3.2 ONLINE FORECASTING IN TWO SPACES

Each instance begins with only five observations, so the forecast bank $\mathcal{M}$ uses simple, low-parameter members and learns their mixture weights online. This choice follows the strong small-sample performance of simple forecasts and forecast combinations (Makridakis et al., 2020; Hyndman & Khandakar, 2008). We use discounted Bayesian model averaging (Hoeting et al., 1999; Raftery

et al., 2010), with a forgetting factor γ that discounts past fit geometrically; Appendix D lists the nine members of one bank and gives the weight update.

For deterministic lead times, nine members are fit in both raw and $\phi(u) = \log(1 + u)$ space, giving $M = 18$. The transform can better represent right-skewed weekly demand, while the raw-space members retain alternative shapes when only five observations are available (Box & Cox, 1964). The online weights choose between them. Stochastic-lead-time cells use only the raw-space bank because their decision rule takes a local demand rate as its forecast input.

We compare members across transform spaces through their raw-space scoring densities, using a change-of-variables correction. A log-space member receives raw-space log density $\log q(\log(1 + d_t)) - \log(1 + d_t)$; the Jacobian term is absent for a raw-space member. Omitting it persistently biases the log-space weights upward relative to the raw-space weights. Lemma 6 quantifies this bias. For both decision rules, the demand-path sampler uses the learned scoring weights to select a member; its within-member sampling distribution is specified separately. Member definitions, scoring densities, and sampling details are in Appendix D.

3.3 Product text as a seasonal prior

Product descriptions in the real instances provide seasonal timing cues, such as "winter coat", before those patterns appear in the short demand history. Synthetic instances lack this text. At design time we ask DeepSeek-V4-Flash for a 52-week seasonal curve η with mean one, used as a multiplicative demand factor. We draw three curves per product and aggregate the accepted, normalized samples by their week-wise median. The aggregate is then clipped, renormalized to mean one, and frozen before evaluation (Appendix D). The curve encodes relative timing only; the online forecaster continues to estimate the demand level.

Our validation generator supplies neither product text nor a real calendar for selecting prior strength. We therefore carry three strengths, $\alpha \in \{0, 1/2, 1\}$. Each runs a copy of the bank on demand deseasonalized by η^{α_g}, and the variants are weighted online by the same discounted predictive likelihood used within the bank. The associated Jacobian keeps their raw-space scoring densities comparable (Appendix D).

The flat variant $\alpha = 0$ applies no seasonal adjustment and recovers the core forecaster's scoring density and path-sampling distribution. Assume uniform initial variant weights and positive mixture density at every observation. If this variant's single-period excess log loss relative to the best variant in that period is always bounded by δ, its mixture weight cannot fall below $G^{-1}e^{-\delta/(1-\gamma)}$, and the mixture's per-period excess log loss over it is at most $\log G + \delta/(1 - \gamma)$. These discounted bounds require the stated single-period condition. With undiscounted weights, the cumulative excess log loss relative to the best fixed variant is at most $\log G$ for every sequence; discounting trades that horizon-free guarantee for adaptation over the year (Proposition 13). Both results characterize the log loss of the scoring densities; the shape layer's effect on control reward is evaluated empirically.

3.4 Deterministic lead times

With deterministic L, the order at t first affects available stock at $t + L$. Theorem 1 characterizes the zero-lead-time case. Positive lead times introduce a pipeline state (Remark 2), which we handle with the following projected-inventory surrogate. Write $U = d_{t+L} - \widehat{I}_{t+L}$ for the shortfall immediately before that order arrives, where $\widehat{I}_{t+L}$ is obtained by simulating the committed pipeline. The q-th unit earns p if $U \geq q$. Otherwise it contributes to a surplus of $(q - U)^+$ units. Approximating clearance at predictive mean rate μ, we assign marginal value

$$m(q) = p\,\mathbb{P}(U \geq q) - h\,\mathbb{E}\left[\left(1 + \kappa\frac{(q - U)^+}{\mu}\right)\mathbf{1}\{U < q\}\right]. \tag{3}$$

The marginal value $m(q)$ defines the surrogate objective $\int_0^q m(u)\,du$. The fitted multiplier κ adjusts the mean-rate clearance approximation. Under the conditions of Proposition 7, m is non-increasing and $q^0 = \inf\{q \geq 0 : m(q) \leq 0\}$ is the least maximizer of this integrated surrogate. When $\kappa = 0$, this reduces to the nonnegative ρ-fractile of U. The zero-lead-time theorem motivates the

cost-based construction; the implemented deterministic policy uses the fitted depletion multiplier, including $\kappa = 1$ at $L = 0$. Appendix D.4 gives its parameterization, validation fit, and root search; §4.1 describes how its functional form was chosen.

The lost-sales recursion projects stock using sales along each sampled path. A textbook inventory-position target uses cumulative demand over the protection interval (Zipkin, 2000). It therefore treats demand observed at an empty shelf as if it had reduced inventory position, even though that demand was lost. For the same sampled path, the textbook shortfall equals our projected shortfall plus the lost sales inside the interval (Proposition 8). It is thus no smaller on any path, and the means differ by expected lost sales. This identity concerns shortfalls and their means. The order gap depends on the nonlinear quantile or crossing in (3). Appendix 3 gives the associated quantile bound.

3.5 Stochastic lead times

In the stochastic cells, $L_t \in \{1, 2, 3, \infty\}$ and an infinite lead time destroys the order. The realized sequences destroy 32% of synthetic orders and 30% of real orders. The deterministic rule in §3.4 assigns an order to one arrival period; stochastic arrivals couple it to several possible future periods. We evaluate replenishment rules through rollouts over these supply outcomes. The evaluator's accounting information identifies whether a positive order survived. Specifically, $q_t \mathbf{1}\{L_t < \infty\} = X_{t+1} - X_t + A_t$; the survival indicator is therefore known one period later (Proposition 9). We use these observations to update the survival probability used in rollouts.

The policy evaluates a small family of complete replenishment rules over the remaining horizon. Candidate rules use common simulated demand and supply paths, sharing demand, survival, and finite-lead-time draws to reduce comparison variance. A coarse pass locates promising parameters, a local grid refines them jointly, and candidate actions are blended according to estimated value relative to simulation uncertainty. The full search is specified in Appendix D.

The design-time loop selected the capped base-stock family

$$\mathcal{K}: \qquad q = \min\Big\{ (\theta r - \mathrm{IP})^+ ,\ cr \Big\}, \tag{4}$$

where r is a local demand forecast and θ, c are searched online. The position term reacts to low inventory, while the cap prevents a destroyed order from triggering an arbitrarily large catch-up order. A pure rate rule sets orders from the demand rate alone, limiting catch-up bursts while leaving its order unchanged across inventory levels. The capped family contains the pure position family at $c = \theta$ and approaches the pure rate family as $\theta \to \infty$ (Proposition 11). On a finite lost-sales instance with order destruction, Appendix C provides a numerical certificate of strict reward separation, conditional on the validity of the estimated slope bounds.

Capped base-stock control is a standard family also used by InventoryBench's OR baseline. In our implementation, its cap scales with a local forecast rate, both parameters are selected online for each instance, and the family was retained by the design-time validation loop. In the small exhaustive examples in Appendix C, the advantage disappears when the lead-time support prevents surviving orders from arriving in a lump, and is positive in most nearby configurations where such a lump can occur.

4 Experiments

4.1 Benchmark, baselines, and protocol

We evaluate all 1,320 InventoryBench instances (Baek et al., 2026); the two stochastic cells account for one third of the overall mean. Table 1 reports the ten public baselines. The OR row is the benchmark authors' capped base-stock heuristic. The remaining rows use GPT-5 Mini, Gemini 3 Flash, or Grok 4.1 Fast each period, either to choose the order directly (*LLM*), estimate inputs to the OR rule (*LLM→OR*), or revise the OR recommendation (*OR→LLM*). We recompute each cell from the released instance-level results; their weighted means reproduce the published overall scores. SabreAgent scores 0.6311 against 0.5380 for the best public baseline and leads all six cells. This comparison varies both model choice and placement: SabreAgent uses DeepSeek-V4-Flash at design time. The same-model online-adjustment control below measures the effect of adding bounded per-period adjustments by that model to the complete SabreAgent policy.

Method	Synthetic			Real			Overall
	$L = 0$	$L = 4$	stoch.	$L = 0$	$L = 4$	stoch.	
Gemini 3 Flash, OR→LLM	0.8391	0.5395	0.4107	0.7439	0.3071	0.3531	0.5380
Grok 4.1 Fast, OR→LLM	0.8493	0.5473	0.3574	0.6914	0.2918	0.3034	0.5139
Gemini 3 Flash, LLM→OR	0.8604	0.5576	0.2434	0.7716	0.3033	0.2375	0.5009
Gemini 3 Flash, LLM	0.8253	0.4341	0.3865	0.7364	0.2400	0.3119	0.4945
Grok 4.1 Fast, LLM→OR	0.8584	0.5555	0.2364	0.7696	0.3085	0.1981	0.4934
GPT-5 Mini, OR→LLM	0.8525	0.4937	0.1643	0.7645	0.2840	0.1819	0.4611
GPT-5 Mini, LLM	0.8371	0.4900	0.1875	0.7646	0.2879	0.1643	0.4597
Grok 4.1 Fast, LLM	0.8387	0.3166	0.3959	0.7068	0.1555	0.3073	0.4593
GPT-5 Mini, LLM→OR	0.8452	0.5302	0.1593	0.7319	0.2797	0.1681	0.4578
OR (capped base-stock)	0.8179	0.5359	0.1068	0.6708	0.3017	0.2100	0.4447
SabreAgent (ours)	**0.8649**	**0.6685**	**0.5063**	**0.8018**	**0.4447**	**0.4714**	**0.6311**
Constant level, hindsight-tuned	*0.8366*	*0.6220*	*0.4653*	*0.7051*	*0.4185*	*0.4563*	*0.5892*
Supply-feasible ceiling	*1.0000*	*0.9155*	*0.6801*	*1.0000*	*0.8945*	*0.7039*	*0.8656*

Table 1: Normalized reward by benchmark cell. Bold marks the best deployable policy in each column. Public rows are recomputed from released instance-level results; SabreAgent is scored by the official evaluator. Italic rows use hindsight and provide diagnostic reference scores.

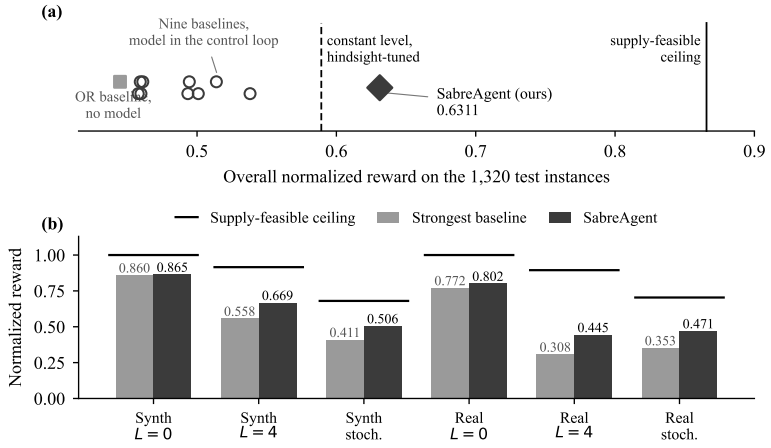
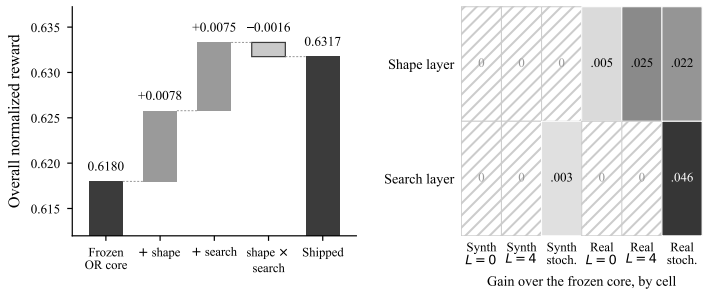


Figure 3: Performance relative to two hindsight diagnostics. (a) Public baselines against a constant order-up-to level chosen per instance (dashed) and a clairvoyant supply-feasible upper bound (solid, 0.8656). (b) Cell-wise results; the upper bound equals 1 only when lead time is zero, so equal vertical gaps do not represent equal attainable headroom.

Automated selection uses fresh-seed instances generated from the documented synthetic families and only the five initial demands available to a policy. The harness cannot access benchmark instances, and the final configuration is chosen from validation scores (Appendix F). Reusing this validation set across rounds incurs adaptive-selection bias within the validation process (Dwork et al., 2015; Blum & Hardt, 2015). Development also includes two indirect test-informed choices. We chose the functional form of κ after observing a systematic cell-level mismatch, then fit its coefficients on validation. The proposer context contains four aggregate test diagnostics that identify stochastic supply as the weak regime, although it contains no test score for any proposed candidate. The reported test result therefore reflects validation-based candidate selection within a development process informed by these two test-diagnostic channels.

4.2 Supply-imposed headroom

S $= 1$ sets the normalization reference. Attainable reward also depends on the supply process. We construct a clairvoyant plan that knows demand and realized lead times but remains subject to feasible arrival times and order destruction. Because orders are free and uncapacitated, each unit of demand can be assigned independently to its latest feasible arrival whenever its profit exceeds the intervening holding cost (Appendix F). The resulting ex-post upper bound is 0.8656 overall, exactly 1 in both zero-lead-time cells, and 0.6801 and 0.7039 in the stochastic cells. Much of their gap to 1 is therefore imposed by the supply constraints.



	Synthetic			Real			
Configuration	$L = 0$	$L = 4$	stoch.	$L = 0$	$L = 4$	stoch.	Overall
Frozen OR core	0.8649	0.6692	0.5026	0.7979	0.4205	0.4163	0.6180
+ shape	+0.0000	+0.0000	+0.0000	+0.0047	+0.0248	+0.0220	+0.0078
+ search	+0.0000	+0.0000	+0.0033	+0.0000	+0.0000	+0.0458	+0.0075
+ shape + search (shipped)	+0.0000	+0.0000	+0.0033	+0.0047	+0.0248	+0.0575	+0.0138
+ forecaster (pooled, not shipped)	−0.0002	+0.0008	+0.0010	+0.0010	+0.0027	+0.0091	+0.0022
+ shape + forecaster	−0.0002	+0.0008	+0.0010	+0.0053	+0.0295	+0.0291	+0.0100
+ search + forecaster	−0.0002	+0.0008	+0.0033	+0.0010	+0.0027	+0.0456	+0.0082
all three	−0.0002	+0.0008	+0.0033	+0.0053	+0.0295	+0.0578	+0.0147

Figure 4: Factorial ablation over six cells and three seeds on the batched path used during design-time search. Moving to the official one-instance-at-a-time path changes the shipped score by -0.0006; values in this figure should therefore be compared within this execution path. Table 1 reports the official path. The plot decomposes the overall gain, including interaction; the table reports the frozen core in absolute terms and all other rows as deltas from it.

With $\kappa = 1$ and realized future demand supplied as a point-mass forecast, the projected base-stock rule matches this bound in all four deterministic cells. This observation concerns the fixed rule with exact demand inputs; positive-lead-time optimality under a demand distribution is a separate question. Appendix F specifies the oracle configuration.

A second diagnostic chooses one constant order-up-to level per instance with hindsight and never revises it. It scores 0.5892, above all ten public baselines overall. The contrast is strongest at lead time four: the constant scores 0.6220 and 0.4185 on synthetic and real data, compared with 0.5576 and 0.3085 for the strongest public baseline in each cell. At zero lead time, eight of ten baselines exceed the constant in each cell, so this observation is specific to settings with a protection interval.

4.3 Component contributions and online-adjustment control

The frozen OR core reaches 0.6180 on the batched ablation path, above every published baseline's overall score. The two model-authored components add +0.0138 on that same path, which is 15% of the gap between the final score and the strongest public baseline. The seasonal layer acts only on the three real-data cells, where its paired mean effect is +0.0179. The search layer acts only on the two stochastic-lead-time cells, where its paired mean effect is +0.0230 (Table 9). Averaging in the zero effects from the remaining cells reduces their overall deltas. The additive decomposition uses the batched path throughout; official one-instance-at-a-time scores are reported separately.

Figure 4 crosses three binary choices: seasonal shapes on or off, the initial or selected stochastic policy family, and the core forecaster or the best alternative found during development. The search contrast includes both research passes: local-rate normalization from the first and the cap from the second. Table 8 separates them on validation. This factorial interpretation is valid because accepted changes affect separate components; Appendix D.7 describes the checks used when composing them.

The seasonal and search layers overlap only in the real stochastic cell; their interaction contributes -0.0016 to the overall weighted mean. The search family and alternative forecaster were selected from validation candidates. Their test contrasts also reflect the test-informed development choices described in §4.1. The seasonal shapes were not selected from test outcomes. Paired bootstrap intervals exclude zero for the seasonal layer, the search layer, and their combination, both overall and within the cells where each operates. The interval for the pooled forecaster includes zero (Table 9).

	Synthetic			Real		
A model returned to the per-period loop	$L = 0$	$L = 4$	stoch.	$L = 0$	$L = 4$	stoch.
change against the shipped policy	-0.0010	-0.0020	$+0.0057$	-0.0008	-0.0299	$+0.0104$
standard error	0.0008	0.0081	0.0059	0.0009	0.0111	0.0046

Table 2: Same-model online-adjustment control. DeepSeek-V4-Flash adds per-period order adjustments, using a bounded multiplier, to the complete SabreAgent policy. Entries are changes in normalized reward on a stratified 124-instance test subset; standard errors are shown in the second row.

For each bootstrap replicate, we sample test instances with replacement and retain the complete row of scores for all arms, preserving their pairing. Across the $1{,}320$ test instances, the combined layers yield a single-seed paired mean gain of $+0.0139$ over the frozen core, with a 95% percentile interval of $[+0.0115, +0.0164]$ (Table 9). The combined layers change scores on 62% of instances, with the ten largest movers contributing 14% of total absolute movement. This estimate uses the instance-level scores for one seed; the +0.0138 gain in Figure 4 averages three seeds. The final configuration's across-seed standard deviation of 0.0001 measures rollout Monte Carlo noise on fixed data. Together, these summaries distinguish rollout variability from sensitivity to the composition of the fixed test set, which is the scope of the paired intervals.

Most proposed changes are rejected. Across two research passes, one of fifty-two candidates contributes to the final search component, and that candidate includes the useful part of the first pass. No proposal targeting deterministic cells survives validation (Appendix F). This screening record shows which proposed changes met the validation criterion before deployment.

On the stratified subset in Table 2, we add bounded per-period order adjustments by DeepSeek-V4-Flash to the complete SabreAgent policy. All four deterministic-cell estimates are negative, but only the real lead-time-four estimate exceeds two standard errors in magnitude. The two stochastic estimates are positive. Within this subset, the real stochastic cell is the only cell with a positive estimate exceeding two standard errors. These results apply to DeepSeek-V4-Flash, this bounded adjustment rule, and this test subset. Broader conclusions about deterministic supply require further evaluation.

The highest-scoring validation configuration pools information across instances, which violates the official one-policy-per-instance protocol. We therefore do not submit it. Its mean gain is +0.0010 over three seeds, while the single-seed paired interval includes zero (Table 9). The submitted configuration preserves the required per-instance information boundary.

Budget and reproducibility. Producing the shipped artifacts used 674 calls to DeepSeek-V4-Flash, mostly three shape samples for each of the 200 real products. Calling a model once per period over the full test set would require 64,200 calls, about 95 times as many (Figure 6). During operation, a fixed seed makes the policy reproducible; it is rerun end to end and scored with the official evaluator as detailed in Appendix F.

5 Conclusion

On InventoryBench, most of SabreAgent's performance gain comes from an OR policy that uses the benchmark's free replenishment, zero salvage, observable demand, and explicit order destruction. Product-specific seasonal priors supply timing cues in the real-data cells, and validation selects a capped base-stock family for lossy supply. These components provide additional gains while statistical forecasting and inventory optimization determine orders during operation. The final policy leads every benchmark cell with zero runtime language-model calls.

The zero-lead-time theorem identifies conditions for exact myopic optimality; positive purchase or setup costs change the value of carried inventory. The controller for positive deterministic lead times uses a surrogate objective, and the seasonal construction gives predictive-log-loss bounds under the conditions in §3.3. Inventory reward is measured in the benchmark experiments. The online-adjustment control evaluates bounded order changes by one model on 124 instances. Matched comparisons across models and larger samples would clarify when these additional runtime adjustments improve the complete policy.

REFERENCES

Ali AhmadiTeshnizi, Wenzhi Gao, and Madeleine Udell. Optimus: Scalable optimization modeling with (mi) lp solvers and large language models. *arXiv preprint arXiv:2402.10172*, 2024.

Kenneth J Arrow, Theodore Harris, and Jacob Marschak. Optimal inventory policy. *Econometrica: Journal of the Econometric Society*, pp. 250–272, 1951.

Jackie Baek, Yaopeng Fu, Will Ma, and Tianyi Peng. Ai agents for inventory control: Human-llm-or complementarity. *arXiv preprint arXiv:2602.12631*, 2026.

Gah-Yi Ban and Cynthia Rudin. The big data newsvendor: Practical insights from machine learning. *Operations Research*, 67(1):90–108, 2019.

Dimitri P. Bertsekas and Steven E. Shreve. *Stochastic optimal control: the discrete-time case*. Academic Press, New York, 1978.

Marco Bijvank and Iris FA Vis. Lost-sales inventory theory: A review. *European Journal of Operational Research*, 215(1):1–13, 2011.

Avrim Blum and Moritz Hardt. The ladder: A reliable leaderboard for machine learning competitions. In *International Conference on Machine Learning*, pp. 1006–1014. PMLR, 2015.

George EP Box and David R Cox. An analysis of transformations. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 26(2):211–243, 1964.

Alexander Capstick, Rahul G Krishnan, and Payam Barnaghi. Autoelicit: Using large language models for expert prior elicitation in predictive modelling. In *International Conference on Machine Learning*, 2025.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge University Press, 2006.

Kristy Choi, Chris Cundy, Sanjari Srivastava, and Stefano Ermon. Lmpriors: Pre-trained language models as task-specific priors. *arXiv preprint arXiv:2210.12530*, 2022.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, 2015.

Kris Johnson Ferreira, Bin Hong Alex Lee, and David Simchi-Levi. Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & service operations management*, 18(1):69–88, 2016.

Robert Fildes, Shaohui Ma, and Stephan Kolassa. Retail forecasting: Research and practice. *International Journal of Forecasting*, 38(4):1283–1318, 2022.

Paul H Garthwaite, Joseph B Kadane, and Anthony O'Hagan. Statistical methods for eliciting probability distributions. *Journal of the American statistical Association*, 100(470):680–701, 2005.

Joren Gijsbrechts, Robert N Boute, Jan A Van Mieghem, and Dennis J Zhang. Can deep reinforcement learning improve inventory management? performance on lost sales, dual-sourcing, and multi-echelon problems. *Manufacturing & Service Operations Management*, 24(3):1349–1368, 2022.

David A Goldberg, Dmitriy A Katz-Rogozhnikov, Yingdong Lu, Mayank Sharma, and Mark S Squillante. Asymptotic optimality of constant-order policies for lost sales inventory models with large lead times. *Mathematics of Operations Research*, 41(3):898–913, 2016.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in neural information processing systems*, 36:19622–19635, 2023.

Erik Hemberg, Stephen Moskal, and Una-May O'Reilly. Evolving code with a large language model. *Genetic Programming and Evolvable Machines*, 25(2):21, 2024.

Jennifer A Hoeting, David Madigan, Adrian E Raftery, and Chris T Volinsky. Bayesian model averaging: a tutorial (with comments by m. clyde, david draper and ei george, and a rejoinder by the authors. *Statistical science*, 14(4):382–417, 1999.

Kejia Hu, Jason Acimovic, Francisco Erize, Douglas J Thomas, and Jan A Van Mieghem. Forecasting new product life cycle curves: Practical approach and empirical analysis: Finalist–2017 m&som practice-based research competition. *Manufacturing & Service Operations Management*, 21(1):66–85, 2019.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *International conference on learning representations*, volume 2024, pp. 32808–32824, 2024.

Woonghee Tim Huh and Paat Rusmevichientong. A nonparametric asymptotic analysis of inventory planning with censored demand. *Mathematics of Operations Research*, 34(1):103–123, 2009.

Woonghee Tim Huh, Ganesh Janakiraman, John A Muckstadt, and Paat Rusmevichientong. Asymptotic optimality of order-up-to policies in lost sales inventory systems. *Management Science*, 55 (3):404–420, 2009.

Rob J Hyndman and Yeasmin Khandakar. Automatic time series forecasting: the forecast package for r. *Journal of statistical software*, 27:1–22, 2008.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. In *International conference on learning representations*, volume 2024, pp. 23857–23880, 2024.

Subbarao Kambhampati, Karthik Valmeekam, Lin Guan, Mudit Verma, Kaya Stechly, Siddhant Bhambri, Lucas Saldyt, and Anil Murthy. Llms can't plan, but can help planning in llm-modulo frameworks. *arXiv preprint arXiv:2402.01817*, 2024.

Samuel Karlin. Inventory models of the arrow-harris-marschak type with time lag. *Studies in the mathematical theory of inventory and production*, 1958.

Retsef Levi, Robin O Roundy, and David B Shmoys. Provably near-optimal sampling-based policies for stochastic inventory control models. *Mathematics of Operations Research*, 32(4):821–839, 2007.

Fei Liu, Xialiang Tong, Mingxuan Yuan, Xi Lin, Fu Luo, Zhenkun Wang, Zhichao Lu, and Qingfu Zhang. Evolution of heuristics: Towards efficient automatic algorithm design using large language model. *arXiv preprint arXiv:2401.02051*, 2024.

Jifei Liu, Zhi Chen, and Yuanguang Zhong. Large language newsvendor: Decision biases and cognitive mechanisms. *arXiv preprint arXiv:2512.12552*, 2025.

Chris Lu, Cong Lu, Robert Tjarko Lange, Yutaro Yamada, Shengran Hu, Jakob Foerster, David Ha, and Jeff Clune. Towards end-to-end automation of ai research. *Nature*, 651(8107):914–919, 2026.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1): 54–74, 2020.

Thomas E Morton. The near-myopic nature of the lagged-proportional-cost inventory problem with lost sales. *Operations Research*, 19(7):1708–1716, 1971.

Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco JR Ruiz, Abbas Mehrabian, et al. Alphaevolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint arXiv:2506.13131*, 2025.

Afshin Oroojlooyjadid, MohammadReza Nazari, Lawrence V Snyder, and Martin Takáč. A deep q-network for the beer game: Deep reinforcement learning for inventory optimization. *Manufacturing & Service Operations Management*, 24(1):285–304, 2022.

Evan L Porteus. *Foundations of stochastic inventory theory*. Stanford University Press, 2002.

Meng Qi, Yuanyuan Shi, Yongzhi Qi, Chenxin Ma, Rong Yuan, Di Wu, and Zuo-Jun Shen. A practical end-to-end inventory management model with deep learning. *Management Science*, 69 (2):759–773, 2023.

Adrian E Raftery, Miroslav Kárnỳ, and Pavel Ettler. Online prediction under model uncertainty via dynamic model averaging: Application to a cold rolling mill. *Technometrics*, 52(1):52–66, 2010.

Rindranirina Ramamonjison, Timothy Yu, Raymond Li, Haley Li, Giuseppe Carenini, Bissan Ghaddar, Shiqi He, Mahdi Mostajabdaveh, Amin Banitalebi-Dehkordi, Zirui Zhou, et al. Nl4opt competition: Formulating optimization problems based on their natural language descriptions. In *NeurIPS 2022 competition track*, pp. 189–203. PMLR, 2023.

James Requeima, John Bronskill, Dami Choi, Richard E Turner, and David Duvenaud. Llm processes: Numerical predictive distributions conditioned on natural language. *Advances in Neural Information Processing Systems*, 37:109609–109671, 2024.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M Pawan Kumar, Emilien Dupont, Francisco JR Ruiz, Jordan S Ellenberg, Pengming Wang, Omar Fawzi, et al. Mathematical discoveries from program search with large language models. *Nature*, 625(7995):468–475, 2024.

Herbert Scarf. The optimality of (s, s) policies in the dynamic inventory problem. In Kenneth J. Arrow, Samuel Karlin, and Patrick Suppes (eds.), *Mathematical Methods in the Social Sciences, 1959*, chapter 13, pp. 196–202. Stanford University Press, 1960.

David Selby, Yuichiro Iwashita, Kai Spriestersbach, Mohammad Saad, Dennis Bappert, Archana Warrier, Sumantrak Mukherjee, Koichi Kise, and Sebastian Vollmer. Had enough of experts? quantitative knowledge retrieval from large language models. *Stat*, 14(2):e70054, 2025.

Mingtian Tan, Mike A Merrill, Vinayak Gupta, Tim Althoff, and Thomas Hartvigsen. Are language models actually useful for time series forecasting? *Advances in Neural Information Processing Systems*, 37:60162–60191, 2024.

Sébastien Thomassey and Antonio Fiordaliso. A hybrid sales forecasting system based on clustering and decision trees. *Decision Support Systems*, 42(1):408–421, 2006.

Karthik Valmeekam, Matthew Marquez, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. Planbench: An extensible benchmark for evaluating large language models on planning and reasoning about change. *Advances in Neural Information Processing Systems*, 36:38975–38987, 2023.

RM Van Steenbergen and Martijn RK Mes. Forecasting demand profiles of new products. *Decision support systems*, 139:113401, 2020.

Ziyang Xiao, Dongxiang Zhang, Yangjun Wu, Lilin Xu, Yuan Wang, Xiongwei Han, Xiaojin Fu, Tao Zhong, Jia Zeng, Mingli Song, et al. Chain-of-experts: When llms meet complex operations research problems. In *International Conference on Learning Representations*, volume 2024, pp. 48519–48537, 2024.

Linwei Xin. Understanding the performance of capped base-stock policies in lost-sales inventory models. *Operations Research*, 69(1):61–70, 2021.

Linwei Xin and David A Goldberg. Optimality gap of constant-order policies decays exponentially in the lead time for lost sales models. *Operations Research*, 64(6):1556–1565, 2016.

Haoran Ye, Jiarui Wang, Zhiguang Cao, Federico Berto, Chuanbo Hua, Haeyeon Kim, Jinkyoo Park, and Guojie Song. Reevo: Large language models as hyper-heuristics with reflective evolution. *Advances in neural information processing systems*, 37:43571–43608, 2024.

Hanyu Zhang, Chuck Arvin, Dmitry Efimov, Michael W Mahoney, Dominique Perrault-Joncas, Shankar Ramasubramanian, Andrew Gordon Wilson, and Malcolm Wolff. Llmforecaster: Improving seasonal event forecasts with unstructured textual data. *arXiv preprint arXiv:2412.02525*, 2024.

Paul Zipkin. Old and new methods for lost-sales inventory systems. *Operations research*, 56(5): 1256–1263, 2008a.

Paul Zipkin. On the structure of lost-sales inventory models. *Operations research*, 56(4):937–944, 2008b.

Paul H Zipkin. *Foundations of inventory management*. McGraw-Hill/Irwin, Boston, 2000.

A Related work

Lost-sales inventory control. Positive-lead-time lost-sales systems generally require the full pipeline state, and ordinary base-stock policies need not be optimal (Karlin, 1958; Zipkin, 2008b;a; Bijvank & Vis, 2011). Different asymptotic regimes favor order-up-to policies when lost-sales penalties grow (Huh et al., 2009), constant-order policies when lead times grow (Goldberg et al., 2016; Xin & Goldberg, 2016), and capped base-stock policies between those extremes (Xin, 2021). InventoryBench already uses a capped base-stock baseline (Baek et al., 2026). Building on this established family, we connect the cap to catch-up bursts caused by destroyed orders, scale it by a local demand rate, and select both parameters online. Theorem 1 concerns a different, degenerate regime: without purchase or setup costs there is no speculative or batching motive, and instantaneous replenishment removes protection-time value as well.

Data-driven inventory policies and forecasting. Data-driven approaches replace a known distribution with samples (Levi et al., 2007), estimate decisions from features (Ban & Rudin, 2019), or train policies on simulated and operational data (Gijsbrechts et al., 2022; Oroojlooyjadid et al., 2022; Qi et al., 2023). InventoryBench instead provides five initial observations and no cross-series training set, which favors a mixture of simple models over a flexible learned forecaster. Discounted predictive-likelihood weighting is standard (Hoeting et al., 1999; Raftery et al., 2010), as are the short-sample forecasting members we combine (Makridakis et al., 2020; Hyndman & Khandakar, 2008). The technical detail specific to our mixture is that members fit in different transform spaces must be scored as densities of the same raw observation; otherwise the omitted Jacobian systematically favors the log-space members.

Language models for search and decision making. Verifier-guided systems use a language model to propose an artifact and an external evaluator to retain or reject it (Romera-Paredes et al., 2024; Liu et al., 2024; Ye et al., 2024; Novikov et al., 2025; Lu et al., 2026). The closest pattern is language-model variation inside evolutionary search over heuristics (Hemberg et al., 2024). In operations research, prior work has translated natural language into formal models checked by a solver (Ramamonjison et al., 2023; AhmadiTeshnizi et al., 2024; Xiao et al., 2024). SabreAgent applies this established search pattern to policy-family design. This choice also reflects evidence that language models remain unreliable as planners or self-correctors (Valmeekam et al., 2023; Huang et al., 2024; Kambhampati et al., 2024) and can show systematic biases in newsvendor decisions (Liu et al., 2025).

Text-derived priors for cold starts. Language models have also been adapted for time-series forecasting (Gruver et al., 2023; Jin et al., 2024; Zhang et al., 2024), although some reported gains survive removal of the language component (Tan et al., 2024). Our seasonal layer uses product descriptions to elicit a prior over relative demand timing. Related work has used language models to infer priors from metadata, mix parametric priors over prompts, or produce predictive beliefs from textual descriptions (Garthwaite et al., 2005; Choi et al., 2022; Capstick et al., 2025; Requeima et al., 2024; Selby et al., 2025). The resulting seasonal variant always competes with a statistical variant without seasonal adjustment. This design addresses the benchmark's cold start. When retailer data on comparable products are available, clustering, attribute regression, and life-cycle models provide more directly grounded signals (Thomassey & Fiordaliso, 2006; Ferreira et al., 2016; Hu et al., 2019; Van Steenbergen & Mes, 2020; Fildes et al., 2022).

B PROOFS

Theorem 1 is the one result the main text states formally, and it is restated here before it is proved, so the restatement carries a fresh number. The six results after it are discussed there in prose and stated formally only here, one to a subsection, in the order the main text raises them. Standing conventions: demand is nonnegative and integrable, $p > 0$ and $h > 0$, $\rho = p/(p+h) \in (0,1)$, and $(x)^+ = \max\{x, 0\}$. For a random variable Z we write $F_Z(z) = \mathbb{P}(Z \le z)$ and $F_Z(z^-) = \mathbb{P}(Z < z)$, which differ exactly at the atoms of Z; this distinction determines the one-sided conditions in Proposition 7 and is kept throughout.

Two distinctions organize the results. The exact-optimality clause of Theorem 1 rests on Assumption 2; §B.1 states this condition and gives a counterexample when it fails. Proposition 8 gives an exact identity for shortfalls and their means, together with bounds for the corresponding quantile gaps (§B.4).

B.1 THEOREM 1: ZERO OPTION VALUE AT $L = 0$

Assumption 1 (the zero-option setting, restated from §2.2)**.** The per-period reward is $p\, s_t - h\, I_t^{\mathrm{end}}$ with $p, h > 0$ and with no purchase, fixed-ordering or salvage term; unmet demand is lost; orders are unbounded above and arrive in the period in which they are placed, $L = 0$; the horizon T is finite; and demand is exogenous, nonnegative, integrable and independent across periods with known marginals F_t.

Setting. Fix an instance satisfying Assumption 1. With $L = 0$ an order placed at t lands inside period t, so the decision at t is equivalent to choosing the post-order level $y \ge I_t$. Writing d_t for the period's demand, the period earns $p\min(d_t, y) - h\,(y - d_t)^+$ and carries $(y - d_t)^+$ into $t+1$, since unmet demand is lost and holding is charged on ending inventory. Define the one-period expected reward and the dynamic program

$$G_t(y) = \mathbb{E}\Big[p\min(d_t, y) - h\,(y - d_t)^+\Big], \qquad V_t(I) = \sup_{y \ge I}\Big\{G_t(y) + \mathbb{E}\big[V_{t+1}((y - d_t)^+)\big]\Big\}, \tag{5}$$

with $V_{T+1} \equiv 0$. We write $J_t(y)$ for the bracketed objective in (5) and $C_t(y) = \mathbb{E}[V_{t+1}((y - d_t)^+)]$ for its continuation term, so $J_t = G_t + C_t$. A *ρ-fractile* of F_t is any y^* with $F_t(y^{*-}) \le \rho \le F_t(y^*)$.

Lemma 2 (the one-period function)**.** *G_t is real-valued and concave on $\mathbb{R}$, with one-sided derivatives $G_t'^{+}(y) = p - (p+h)F_t(y)$ and $G_t'^{-}(y) = p - (p+h)F_t(y^-)$. Its set of maximizers over $\mathbb{R}$ is exactly the set of ρ-fractiles of F_t, every such point is nonnegative, G_t is non-decreasing on $(-\infty, y_t^*]$ and non-increasing on $[y_t^*, \infty)$ for any ρ-fractile y_t^*.*

Proof. For each fixed $d \ge 0$ the map $y \mapsto \min(d, y)$ is concave and $y \mapsto (y - d)^+$ is convex, so the integrand is concave in y; concavity is preserved by the expectation, which is finite because $|\min(d_t, y)| \le |y|$ and $0 \le (y - d_t)^+ \le (y)^+$, so G_t is finite at every y with no integrability hypothesis at all. Integrability of d_t is needed for V_t, not here.

For the right derivative, $\min(d, y + \varepsilon) - \min(d, y) = \min((d - y)^+, \varepsilon)$, so the difference quotient $\varepsilon^{-1}\min((d - y)^+, \varepsilon)$ increases to $\mathbf{1}\{d > y\}$ as $\varepsilon \downarrow 0$ and monotone convergence gives $\mathbb{P}(d_t > y)$. Likewise $(y + \varepsilon - d)^+ - (y - d)^+ = \varepsilon\mathbf{1}\{d \le y\} + (y + \varepsilon - d)\mathbf{1}\{y < d < y + \varepsilon\}$, whose difference quotient converges to $\mathbf{1}\{d \le y\}$ boundedly, giving $F_t(y)$. Hence $G_t'^{+}(y) = p(1 - F_t(y)) - hF_t(y) = p - (p+h)F_t(y)$, and the same computation from the left gives $F_t(y^-)$.

By concavity, y^* maximizes G_t if and only if $G_t'^{-}(y^*) \ge 0 \ge G_t'^{+}(y^*)$, that is $F_t(y^{*-}) \le \rho \le F_t(y^*)$, which is the definition of a ρ-fractile. Since $d_t \ge 0$ we have $F_t(y) = 0$ for $y < 0$, so $G_t'^{+}(y) = p > 0$ there and no maximizer is negative. Monotonicity on either side of y_t^* is concavity together with the sign of the one-sided derivatives. □

Lemma 2 is where the zero purchase cost enters. A unit purchase cost $c_{\mathrm{b}} > 0$ would replace the objective in (5) by $J_t(y) - c_{\mathrm{b}}(y - I_t)$, in which the entering inventory appears as a linear credit, not only through the constraint $y \ge I_t$; a carried unit would then save c_{b} at every later period, which is precisely the option value the theorem denies.

For the exact-optimality clause of Theorem 1, we state the non-decreasing-fractile condition as a separate assumption.

Assumption 2 (monotone myopic levels)**.** The ρ-fractiles can be selected non-decreasing in time: there exist ρ-fractiles $y_1^* \le y_2^* \le \cdots \le y_T^*$ of $F_1, \ldots, F_T$.

Assumption 2 holds whenever the demands are identically distributed, in which case all the levels coincide, and more generally whenever d_t is stochastically non-decreasing in t, since then $F_{t+1} \le F_t$ pointwise and the smallest ρ-fractiles are already non-decreasing. Remark 1 and Example 1 below show that it cannot be dropped.

Theorem 3 (Exact-optimality clause of Theorem 1)**.** *Let Assumption 1 hold, let the demands be independent with known laws F_t and finite means, and let Assumption 2 hold with the selection $y_1^* \le \cdots \le y_T^*$. Then for every $t \le T$ the value function V_t of* (5) *is constant on $[0, y_t^*]$ and non-increasing on $[0, \infty)$, the supremum in* (5) *is attained at $y = \max\{I, y_t^*\}$, and the policy $y_t = \max\{I_t, y_t^*\}$ is optimal. Consequently a unit carried from $t-1$ into t has zero marginal value whenever $I_t \le y_t^*$, and the myopic newsvendor solution is exactly optimal, not asymptotically or approximately so.*

This restatement is the conditional half of Theorem 1. The unconditional half, that V_t is non-increasing and that no post-order level above $\max\{I, y_t^*\}$ is strictly better, needs neither Assumption 2 nor the induction below and is separated out as Proposition 5. It is the half the design of §3.4 rests on, so the two are proved apart to keep visible which one carries which hypothesis.

The benchmark fixes each instance's demand path in advance and takes the expectation over the policy's own randomization alone (§2.1), while the theorem treats demand as random with known marginals F_t. The theorem describes the policy's stochastic planning model. Taking $F_t = \delta_{d_t}$ degenerate recovers the latter, and the myopic policy then becomes the clairvoyant rule that §4.2 measures.

Proof. Backward induction on t, with induction hypothesis the conjunction

(H1) V_{t+1} is constant on $[0, y_{t+1}^*]$, and **(H2)** V_{t+1} is non-increasing on $[0, \infty)$,

where for $t+1 = T+1$ we read $y_{T+1}^* = +\infty$.

Base case. At $t = T$ the continuation is $V_{T+1} \equiv 0$, which is constant on all of $[0, \infty)$ and non-increasing, so (H1) and (H2) hold. This is the only place the zero salvage value is used, and it is used decisively: a salvage credit $\varsigma > 0$ would give $V_{T+1}(I) = \varsigma I$, which is strictly increasing and violates both halves of the hypothesis at once.

Inductive step. Assume (H1) and (H2) at $t+1$ and fix $t \le T$.

Step 1: the continuation is non-increasing. For every $d \ge 0$ the map $y \mapsto (y-d)^+$ is non-decreasing, so by (H2) the map $y \mapsto V_{t+1}((y-d)^+)$ is non-increasing; taking expectations, C_t is non-increasing on $[0, \infty)$. Independence of d_t from the past is what makes C_t a function of y alone.

Step 2, the key step: the continuation is flat below the myopic level. Let $0 \le y \le y_t^*$. Then $(y-d)^+ \le y \le y_t^* \le y_{t+1}^*$ for every $d \ge 0$, where the last inequality is Assumption 2. So the argument of V_{t+1} lies in $[0, y_{t+1}^*]$ pathwise, and (H1) gives $V_{t+1}((y-d_t)^+) = V_{t+1}(0)$ almost surely. Hence $C_t(y) = V_{t+1}(0)$ for all $y \in [0, y_t^*]$: on that range the continuation contributes an additive constant, and the first-order condition for J_t is the first-order condition for G_t alone. This is the mechanism the theorem asserts. It is available because ordering is free and instantaneous, so any unit not carried forward can be replaced at $t+1$ at no cost, which is exactly what makes V_{t+1} flat below its own target.

Step 3: y_t^* maximizes J_t. On $[0, y_t^*]$ we have $J_t = G_t + V_{t+1}(0)$, which by Lemma 2 is non-decreasing and is maximized over that interval at y_t^*. For $y > y_t^*$, Lemma 2 gives $G_t(y) \le G_t(y_t^*)$ and Step 1 gives $C_t(y) \le C_t(y_t^*)$, so $J_t(y) \le J_t(y_t^*)$. Therefore y_t^* maximizes J_t over $[0, \infty)$, and J_t is non-increasing on $[y_t^*, \infty)$ because both of its terms are.

Step 4: re-establishing the hypothesis at t. For $I \le y_t^*$ the unconstrained maximizer y_t^* is feasible, so $V_t(I) = J_t(y_t^*)$. For $I > y_t^*$ the objective is non-increasing on the feasible set $[I, \infty)$ by

Step 3, so the supremum is attained at $y = I$ and $V_t(I) = J_t(I)$. Both cases are summarized by $V_t(I) = J_t(\max\{I, y_t^*\})$, which is the claimed optimal post-order level. It follows that V_t is constant on $[0, y_t^*]$, giving (H1) at t, and that V_t is non-increasing on $[0, \infty)$, since it is constant up to y_t^* and equals J_t, which is non-increasing, above it. This gives (H2) at t and closes the induction.

Zero marginal value of carried stock is (H1) read as a difference: for $0 \le I' \le I \le y_t^*$, $V_t(I) - V_t(I') = 0$. Optimality of $y_t = \max\{I_t, y_t^*\}$ is Step 4 applied at every period. That a Markov policy attaining the per-period maximum in (5) is optimal among all history-dependent policies is the verification theorem for a finite-horizon problem with bounded-below rewards and a one-dimensional Borel state (Bertsekas & Shreve, 1978); the expectation $\mathbb{E}[V_{t+1}((y - d_t)^+)]$ is defined because V_{t+1} is monotone and hence Borel. □

Zero fixed ordering cost is also part of the theorem's setting. With a setup charge $K > 0$ the period objective becomes $J_t(y) - K\mathbf{1}\{y > I\}$, which is not concave in y, so Lemma 2 no longer identifies the maximizer and the optimal policy is of (s, S) form, no longer base-stock (Scarf, 1960). Under $K > 0$ it is optimal not to order at all for entering inventories somewhat below the target, so V_t is not flat on $[0, y_t^*]$ and the conclusion fails even though the mechanism of Step 2 is untouched.

The effect of decreasing myopic levels depends on the reachable leftover inventory. A high-demand period preceding a low one is not by itself enough to break exact optimality. Because ordering is free, V_{t+1} is flat on $[0, y_{t+1}^*]$ by (H1), so carrying surplus into $t + 1$ costs nothing while the leftover stays under the next level, the policy having been going to order up to it in any case. What breaking exact optimality needs is that the drop be *reachable*, $\mathbb{P}((y_t^* - d_t)^+ > y_{t+1}^*) > 0$: only then does a unit ordered at the fractile face a period in which no order would have been placed for it. Reachability is necessary and not sufficient, since the holding it exposes the unit to must also outweigh the current period's marginal value. Exact backward induction over a bounded integer state exhibits all three cases. With y^* falling from 17 to 11 but demand never below 12, so that the leftover cannot exceed 11, the myopic level stays optimal at every state despite the drop; with a reach probability of 0.012 it is still optimal; and with y^* falling from 16 to 1 at a reach probability of 0.714 the optimum drops six units below the fractile.

Corollary 4 (integer orders)**.** *Assume in addition that d_t is integer valued and that the post-order level is restricted to the integers, as the benchmark's flooring of order quantities enforces. Then $G_t(y + 1) - G_t(y) = p - (p + h)F_t(y)$ for integer y, this difference is non-increasing in y, and the smallest integer y with $F_t(y) \ge \rho$ maximizes G_t over the integers. Under the hypotheses of Theorem 3 the optimal integer post-order level is $\max\{I_t, y_t^{*,\mathbb{Z}}\}$, where $y_t^{*,\mathbb{Z}} = \min\{k \in \mathbb{Z} : F_t(k) \ge \rho\}$.*

Proof. For integer valued d and integer y, $\min(d, y + 1) - \min(d, y) = \mathbf{1}\{d \ge y + 1\}$ and $(y + 1 - d)^+ - (y - d)^+ = \mathbf{1}\{d \le y\}$. Taking expectations, $G_t(y + 1) - G_t(y) = p\,\mathbb{P}(d_t > y) - h\,\mathbb{P}(d_t \le y) = p - (p + h)F_t(y)$, which is non-increasing in y because F_t is non-decreasing. A function on $\mathbb{Z}$ with non-increasing forward differences is maximized at the first y where the forward difference turns non-positive, that is at the smallest integer with $F_t(y) \ge \rho$.

Such an integer is a ρ-fractile of F_t, but Assumption 2 is a property of a selection and the integer selection need not inherit it: with d_1 uniform on $\{2, 5\}$ and d_2 uniform on $\{1, 4\}$ at $\rho = 1/2$, the monotone selection $(2, 2)$ exists while $y^{*,\mathbb{Z}} = (2, 1)$ decreases. The theorem therefore cannot be invoked with that selection, and what carries the induction on the lattice is instead that $y_t^{*,\mathbb{Z}} \le y_t^*$, being the smallest such integer, together with the fact that the lattice value function is flat on $[0, y_{t+1}^*] \cap \mathbb{Z}$ and not merely up to $y_{t+1}^{*,\mathbb{Z}}$: either the fractile is unique, in which case the two levels coincide, or $F_{t+1} \equiv \rho$ between them, in which case the forward difference $p - (p + h)F_{t+1}$ vanishes there and G_{t+1} is flat across the gap. Step 2 then runs with $(y - d)^+ \le y \le y_t^{*,\mathbb{Z}} \le y_t^* \le y_{t+1}^*$, and Steps 1, 3 and 4 use only monotonicity of C_t and of F_t, which the restriction to the lattice preserves. □

The next result gives the upper-bound property under the zero-option setting alone; the following example identifies the role of Assumption 2 in exact optimality.

Proposition 5 (what survives without monotone myopic levels)**.** *Let Assumption 1 hold with independent, integrable demands, and let y_t^* be any ρ-fractile of F_t. Then for every t and every $I \geq 0$,*

$$V_t(I) \;=\; \sup_{I \leq y \leq \max\{I, y_t^*\}} J_t(y), \tag{6}$$

and V_t is non-increasing on $[0, \infty)$. In words, ordering above the myopic fractile never strictly pays, so some optimal post-order level lies at or below $\max\{I, y_t^*\}$: the myopic level bounds the optimum from above and never from below. When the fractile is not unique a higher level can tie, which is why the statement is (6) *and not a claim that every optimal level is at or below it.*

Proof. That V_t is non-increasing is immediate from (5), since the feasible sets $\{y \geq I\}$ shrink as I grows, and it needs no induction. Given this at $t+1$, Step 1 of the proof of Theorem 3 applies verbatim and C_t is non-increasing; with Lemma 2 this makes J_t non-increasing on $[y_t^*, \infty)$, so no $y > \max\{I, y_t^*\}$ can improve on $\max\{I, y_t^*\}$, which is (6). □

Remark 1. Assumption 2 is used only in Step 2, and only to guarantee that the reachable states $(y-d)^+$ for $y \leq y_t^*$ stay inside the flat region $[0, y_{t+1}^*]$ of V_{t+1}. When the myopic levels decrease over time this can fail, and where it does it fails in a direction: stock left over from t can land above the next period's target, where it is charged holding without displacing any future order, so the true optimum orders strictly less than the myopic fractile. A decrease is not by itself enough, since Step 2 needs only $(y_t^* - \operatorname{essinf} d_t)^+ \leq y_{t+1}^*$, which is strictly weaker than Assumption 2 and can hold while the levels fall; the discussion after Theorem 3 exhibits both sides numerically. The failure direction is the same as the inequality of Proposition 5. The depletion term κ of Proposition 7 acts in the same direction; κ remains an empirical correction fitted for a separate effect.

Example 1 (the hypothesis cannot be dropped)**.** Take $T = 2$, $p = h = 1$ so $\rho = 1/2$, demands independent with $d_1 \sim \mathrm{Uniform}[0, 100]$ and $d_2 \equiv 0$. Every hypothesis of Assumption 1 holds, and $y_1^* = 50$ while $y_2^* = 0$, so Assumption 2 fails. In period 2 the reward of a post-order level $y \geq I$ is $-y$, so $V_2(I) = -I$. Hence

$$J_1(y) = \mathbb{E}[\min(d_1, y)] - \mathbb{E}\,(y - d_1)^+ - \mathbb{E}\,(y - d_1)^+ = \mathbb{E}[\min(d_1, y)] - 2\,\mathbb{E}\,(y - d_1)^+,$$

and for $y \in [0, 100]$ this is $y - 3y^2/200$, maximized at $y = 100/3$ with value $50/3 \approx 16.67$, whereas the myopic level $y_1^* = 50$ yields 12.5. The myopic policy is strictly suboptimal, and $V_1(I) = J_1(\max\{I, 100/3\})$ is strictly decreasing on $(100/3, 50]$, so V_1 is not constant on $[0, y_1^*]$ either. Both halves of the conclusion of Theorem 3 fail, while (6) holds with room to spare.

Remark 2 (the pipeline state at $L > 0$). The zero-lead-time assumption is essential to Theorem 3. At $L > 0$, the order placed at t lands at $t + L$, so the inventory it creates cannot be topped up before it arrives and the states reachable at $t+1, \ldots, t+L$ depend on y through the entire pipeline vector rather than a scalar. Step 2 is the step that fails: the continuation is no longer flat in the current decision, because a shortage inside the protection interval cannot be repaired by a free order that arrives in time, so carried stock does acquire option value and the one-dimensional dynamic program (5) is not the right object. Lost-sales systems with positive lead time have no known simple optimal policy and a state that is the whole pipeline (Zipkin, 2008a;b; Bijvank & Vis, 2011). The rule of §3.4 uses Theorem 3 as a design motivation for a surrogate objective. It assigns surplus beyond the responsibility period holding costs without a continuation credit, and prices the responsibility period through the explicit marginal condition of Proposition 7. The proposition characterizes the maximizers of the integrated surrogate; dynamic optimality remains specific to the zero-lead-time setting.

B.2 Lemma 6: comparability of two transform spaces

Lemma 6 (Two-space comparability)**.** *Let $\gamma \in (0, 1)$, let ϕ be strictly increasing and differentiable with $\phi' > 0$, and let a member be fitted on $z = \phi(d)$ with predictive density q. Then the member's predictive density in raw space is $q(\phi(d))\,\phi'(d)$, so its raw-space predictive log-density is $\log q(\phi(d)) + \log \phi'(d)$. For $\phi(d) = \log(1+d)$ on $d \geq 0$ the correction is $\log \phi'(d) = -\log(1+d) \leq 0$. Suppose this correction is omitted. Then the log-weight of every log-space member is inflated by $\log(1+d_t)$ per period relative to every identity-space member, whose correction*

is zero, and under the discounted update $\ell_t = \gamma\ell_{t-1} + (\textit{log-density})$ *the induced relative offset* Δ_t *obeys* $\Delta_t = \gamma\Delta_{t-1} + \log(1+d_t)$ *with* $\Delta_0 = 0$*, so*

$$\mathbb{E}\Delta_t = \frac{1-\gamma^t}{1-\gamma}\,\mathbb{E}[\log(1+d)] \;\xrightarrow[t\to\infty]{}\; \frac{\mathbb{E}[\log(1+d)]}{1-\gamma}$$

for identically distributed integrable $\log(1+d_t)$*. The offset does not cancel in the softmax and does not vanish as data accumulate.*

Proof. Change of variables. Since ϕ is strictly increasing, $\{Y \le d\} = \{Z \le \phi(d)\}$ for $Z = \phi(Y)$, so the raw-space distribution function is $F_Y(d) = Q(\phi(d))$ with Q the distribution function of Z. Differentiating and using differentiability of ϕ gives the density $f_Y(d) = q(\phi(d))\phi'(d)$, and taking logarithms gives the stated decomposition. For $\phi(d) = \log(1+d)$ we have $\phi'(d) = 1/(1+d)$ and $\log\phi'(d) = -\log(1+d)$, which is nonpositive for $d \ge 0$ and zero only at $d = 0$.

Accumulation. Write $\ell^{\mathrm{c}}_{t,m}$ for the log-weight accumulated with the correction and $\ell^{\mathrm{o}}_{t,m}$ for the one that omits it. Both obey the same recursion with per-period increments differing by $-\log\phi'_m(d_t)$, so $\Delta_{t,m} = \ell^{\mathrm{o}}_{t,m} - \ell^{\mathrm{c}}_{t,m}$ satisfies $\Delta_{t,m} = \gamma\Delta_{t-1,m} - \log\phi'_m(d_t)$ with $\Delta_{0,m} = 0$. For an identity-space member $\log\phi'_m \equiv 0$ and $\Delta_{t,m} \equiv 0$; for a log1p member the increment is $\log(1+d_t) \ge 0$. First consider the deterministic recursion $l_t = \gamma l_{t-1} + b$ with a constant offset b and $\gamma \in (0,1)$: subtracting the fixed point $b/(1-\gamma)$ gives $l_t - b/(1-\gamma) = \gamma(l_{t-1} - b/(1-\gamma))$, hence $l_t = \gamma^t l_0 + b(1-\gamma^t)/(1-\gamma) \to b/(1-\gamma)$ geometrically. Taking expectations in the stochastic recursion and using that $\mathbb{E}[\log(1+d_t)] = b$ does not depend on t puts $\mathbb{E}\Delta_{t,m}$ in exactly this form with $l_0 = 0$, which is the displayed limit. When the demands are in addition independent, $\Delta_{t,m} = \sum_{s<t}\gamma^s\log(1+d_{t-s})$ converges in distribution to $\sum_{s\ge 0}\gamma^s\log(1+d_{-s})$, a proper random variable with mean $b/(1-\gamma)$; the deterministic statement is the statement about its mean.

Non-cancellation. The softmax is invariant under a shift common to all members: for any $c \in \mathbb{R}$, $\exp(\ell_m + c)/\sum_{m'}\exp(\ell_{m'} + c) = \exp(\ell_m)/\sum_{m'}\exp(\ell_{m'})$. A Jacobian correction identical across members would therefore be immaterial, and this is why one can omit a shared normalizing constant with no consequence. The correction here is not shared: it is 0 on the nine identity-space members and $-\log(1+d_t)$ on the nine log1p members. What survives in the weights is the difference, and for a log1p member m and an identity member m' the omitted version multiplies the odds $w_{t,m}/w_{t,m'}$ by $\exp(\Delta_{t,m})$, whose logarithm has expectation tending to $\mathbb{E}[\log(1+d)]/(1-\gamma)$. Since $d \ge 0$ the offset has a fixed sign, so it is a standing tilt, not a fluctuation that averages out, and because the limit is a nonzero constant whenever $\mathbb{P}(d > 0) > 0$ it does not shrink as the series lengthens. Adding $\log\phi'$ is thus sufficient to put the two halves of the bank on one scale, and it is necessary up to a member-independent additive constant, since those are exactly the transformations the softmax cannot see. □

At the shipped $\gamma = 0.92$ the geometric sum is $1/(1-\gamma) = 12.5$ periods, so a series running at a level of about one hundred units, where $\log(1+d) \approx 4.6$, produces a steady-state log-odds tilt of roughly 58 nats in favor of the log1p half of the bank. At this scale, omitting the correction concentrates the mixture weight on the log1p space.

B.3 Proposition 7: the marginal ordering condition

Proposition 7 (Marginal ordering condition)**.** *Let* U *be a real random variable with* $\mathbb{E}|U| < \infty$*, let* $p, h, \mu > 0$ *and* $\kappa \ge 0$*, and define*

$$\Phi(q) \;=\; p\,\mathbb{P}(U \ge q) \;-\; h\,\mathbb{E}\Big[\Big(1 + \kappa\tfrac{(q-U)^+}{\mu}\Big)\mathbf{1}\{U < q\}\Big]. \tag{7}$$

Then:

(i) $\Phi(q) = p - (p+h)\mathbb{P}(U < q) - \frac{h\kappa}{\mu}\mathbb{E}\,(q-U)^+$*, and* Φ *is non-increasing and left-continuous on* $\mathbb{R}$*;*

(ii) $\Phi \equiv p$ *on* $(-\infty, \operatorname{essinf} U]$*; if* $\kappa > 0$ *then* Φ *is strictly decreasing on* $(\operatorname{essinf} U, \infty)$*; if* $\kappa = 0$ *then for* $q_1 < q_2$ *one has* $\Phi(q_1) > \Phi(q_2)$ *if and only if* $\mathbb{P}(q_1 \le U < q_2) > 0$*;*

(iii) if $\kappa > 0$ *then* $\Phi(q) \to -\infty$ *at least linearly as* $q \to \infty$*; if* $\kappa = 0$ *then* $\Phi(q) \to -h < 0$*;*

(iv) in both cases $q^0 = \inf\{q \geq 0 : \Phi(q) \leq 0\}$ *is finite,* $\Phi > 0$ *on* $[0, q^0)$ *and* $\Phi \leq 0$ *on* (q^0, ∞)*, so the crossing is unique, and* $q \mapsto \Psi(q) = \int_0^q \Phi$ *is concave. If* $\Phi(0) < 0$*, which is the case when the projected pipeline already covers the responsible demand, then* $q^0 = 0$ *and* $\arg\max_{q \geq 0} \Psi = \{0\}$*. Otherwise* $\Phi(q^0) \geq 0$ *by left-continuity and* $\arg\max \Psi = \{q \geq 0 : \Phi(q^+) \leq 0 \leq \Phi(q)\}$*, an interval whose left endpoint is* q^0*. On the integer lattice concavity puts the maximizer at* $\lfloor q^0 \rfloor$ *or* $\lceil q^0 \rceil$ *and the deployed rule takes the floor, which can be strictly suboptimal;*

(v) bisection on any bracket $[\mathrm{lo}, \mathrm{hi}] \ni q^0$ *that moves* lo *when* $\Phi(\mathrm{mid}) > 0$ *and* hi *otherwise converges to* q^0*, with error at most* $2^{-(k+1)}(\mathrm{hi} - \mathrm{lo})$ *after* k *iterations;*

(vi) setting $\kappa = 0$ *gives* $\Phi(q) = p - (p + h)\mathbb{P}(U < q)$*, whose crossing on* $\mathbb{R}$ *is a* ρ*-fractile* q *of* U*; since* q^0 *is an infimum over* $q \geq 0$ *and* q *is negative exactly when the projected pipeline covers the responsible demand,* $q^0 = (q)^+$*.*

Proof. (i) On $\{U \geq q\}$ we have $(q - U)^+ = 0$, so the indicator in the second term of (7) is redundant on the κ part and

$$\mathbb{E}\Big[\Big(1 + \kappa \tfrac{(q-U)^+}{\mu}\Big)\mathbf{1}\{U < q\}\Big] = \mathbb{P}(U < q) + \frac{\kappa}{\mu}\mathbb{E}\,(q - U)^+ .$$

Substituting and using $\mathbb{P}(U \geq q) = 1 - \mathbb{P}(U < q)$ gives the displayed form. The map $q \mapsto \mathbb{P}(U < q)$ is non-decreasing and left-continuous, and $q \mapsto \mathbb{E}\,(q - U)^+ = \int_{-\infty}^q \mathbb{P}(U < u)\,du$ is non-decreasing, convex and continuous, so Φ is a nonpositive combination of non-decreasing functions plus a constant, hence non-increasing, and left-continuous.

(ii) For $q \leq \operatorname{essinf} U$ both $\mathbb{P}(U < q)$ and $\mathbb{E}\,(q - U)^+$ vanish, so $\Phi(q) = p$, and Φ is constant there. This is the qualification the main text needs: below the essential infimum of the shortfall no unit is ever wasted, so the marginal value is the full margin and nothing in q moves it. For $\operatorname{essinf} U < q_1 < q_2$ the integral representation gives

$$\mathbb{E}\,(q_2 - U)^+ - \mathbb{E}\,(q_1 - U)^+ = \int_{q_1}^{q_2} \mathbb{P}(U < u)\,du \;\geq\; (q_2 - q_1)\mathbb{P}(U < q_1) \;>\; 0,$$

since $\mathbb{P}(U < u) \geq \mathbb{P}(U < q_1) > 0$ for $u > q_1 > \operatorname{essinf} U$. With $h\kappa/\mu > 0$ this makes $\Phi(q_1) - \Phi(q_2) > 0$, so Φ is strictly decreasing there. When $\kappa = 0$ only the term $(p + h)\mathbb{P}(U < q)$ varies, and $\mathbb{P}(U < q_2) - \mathbb{P}(U < q_1) = \mathbb{P}(q_1 \leq U < q_2)$, which gives the stated equivalence.

(iii) Suppose $\kappa > 0$. Pick any a with $\delta := \mathbb{P}(U \leq a) > 0$, for instance a median. Then $\mathbb{E}\,(q - U)^+ \geq (q - a)\delta$ for $q \geq a$, so $\Phi(q) \leq p - \frac{h\kappa}{\mu}\delta(q - a) \to -\infty$, and the divergence is at least linear in q. If instead $\kappa = 0$ then $\Phi(q) = p - (p + h)\mathbb{P}(U < q) \to p - (p + h) = -h$ as $q \to \infty$. The limit is finite, so divergence to $-\infty$ holds only for $\kappa > 0$; what the argument in (iv) needs is not divergence but that Φ is eventually negative, and $-h < 0$ supplies that.

(iv) By (iii) there is $\bar{q}$ with $\Phi(\bar{q}) < 0$, so the set $\{q \geq 0 : \Phi(q) \leq 0\}$ is nonempty and q^0 is finite. Because Φ is non-increasing, $\Phi > 0$ on $[0, q^0)$ by definition of the infimum and $\Phi \leq 0$ on (q^0, ∞); the crossing point is therefore unique, and for $q^0 > 0$ left-continuity forces $\Phi(q^0) \geq 0$. If $\Phi(0) \leq 0$ the infimum is attained at 0 and $q^0 = 0$; if the inequality is strict then $\Phi \leq \Phi(0) < 0$ on all of $[0, \infty)$, so Ψ is strictly decreasing, $\arg\max \Psi = \{0\}$, and the crossing set below is empty, which is why that case is separated in the statement. Since Φ is non-increasing, $\Psi(q) = \int_0^q \Phi(u)\,du$, the total expected value of ordering q units when the q-th unit is worth $\Phi(q)$, is concave, and its maximizers are exactly the points where its derivative changes sign, that is the crossing set described. When Φ is strictly decreasing at q^0, which by (ii) holds whenever $\kappa > 0$ and $q^0 > \operatorname{essinf} U$, that set is the single point q^0 and coincides with the strict-inequality variant $\inf\{q \geq 0 : \Phi(q) < 0\}$. Otherwise Φ vanishes on an interval, every point of which maximizes Ψ, so the two definitions select different but equally optimal points.

(v) The update maintains the invariant $\mathrm{lo} \leq q^0 \leq \mathrm{hi}$: if $\Phi(\mathrm{mid}) > 0$ then $\mathrm{mid} < q^0$ and raising lo to mid preserves it, and if $\Phi(\mathrm{mid}) \leq 0$ then $q^0 \leq \mathrm{mid}$ and lowering hi preserves it. The bracket width halves at each iteration, so the midpoint after k iterations is within $2^{-(k+1)}$ of the initial width of q^0, which is linear convergence. The initial bracket used by the numerical procedure is valid: it takes

$\mathrm{lo} = 0$, and $\mathrm{hi} = 1.05 \max\{\max_k U^{(k)}, 1\}$ on the empirical distribution of the sampled shortfalls, for which $\Phi(q) = -h(1 + \kappa\,\overline{(q-U)}/\mu) < 0$ once q exceeds every sample.

(vi) With $\kappa = 0$, the same one-sided derivative argument as in Lemma 2 characterizes the unconstrained maximizers by $F_U(q^-) \le \rho \le F_U(q)$. Let q be the smallest such ρ-fractile. Restricting orders to the nonnegative domain and using the infimum definition in (iv) gives $q^0 = (q)^+$. At an atom, the two one-sided inequalities apply; an equality such as $\mathbb{P}(U < q) = \rho$ need not hold. The empirical strict-inequality frequency used to evaluate Φ is consistent with this infimum definition. □

B.4 Proposition 8: pathwise over-ordering under lost sales

This proposition derives an accounting identity on each fixed sample path, valid for every demand distribution. The probabilistic statements then follow from the pathwise equalities.

Set-up. Fix a period t and a deterministic lead time $L \ge 0$ with $t + L \le T$, without which d_{t+L} does not exist and orders scheduled past the horizon sit in X_t without ever arriving. At the decision epoch of period t the policy holds I_t on hand and X_t in transit, and $\mathrm{IP}_t = I_t + X_t$. Because the lead time is deterministic, an order placed at $t - j$ lands at $t - j + L$, so the orders still outstanding at the decision epoch of t are exactly those placed at $t - L, \dots, t - 1$ and they land in periods $t, \dots, t + L - 1$; the order placed at t lands at $t + L$, and every later order lands after $t + L$. Let A_j be arrivals in period j, $s_j = \min(d_j, I_j + A_j)$ the units sold and $\lambda_j = (d_j - (I_j + A_j))^+$ the demand lost, so that $s_j = d_j - \lambda_j$ by definition of the minimum. Write $\Lambda = \sum_{j=t}^{t+L-1} \lambda_j$ for the lost sales realized inside the protection interval, and let $\widehat{I}_{t+L} = I_{t+L}$ be the on-hand stock at the decision epoch of $t + L$, that is, before the order placed at t lands.

Proposition 8 (Lost-sales over-ordering of the textbook rule)**.** *With the notation above, put* $U^{\mathrm{naive}} = \sum_{j=t}^{t+L} d_j - \mathrm{IP}_t$ *and* $U^{\mathrm{proj}} = d_{t+L} - \widehat{I}_{t+L}$*. Then*

$$U^{\mathrm{naive}} \;=\; U^{\mathrm{proj}} + \Lambda \qquad \textit{pathwise.} \tag{8}$$

Consequently $U^{\mathrm{naive}} \ge U^{\mathrm{proj}}$ *pathwise, hence* U^{naive} *dominates* U^{proj} *stochastically;* $T(U^{\mathrm{naive}}) \ge T(U^{\mathrm{proj}})$ *for every functional* T *with the monotonicity property that* $T(U + \Lambda) \ge T(U)$ *whenever* $\Lambda \ge 0$ *almost surely, a class that contains every quantile and the mean; the two shortfalls agree almost surely if and only if* $\Lambda = 0$ *almost surely, that is if and only if the shelf never empties inside the protection interval;* $\mathbb{E}U^{\mathrm{naive}} - \mathbb{E}U^{\mathrm{proj}} = \mathbb{E}\Lambda$ *when both are integrable; and* Λ *is non-decreasing in* L *along a fixed arrival stream.*

Proof. The on-hand recursion is $I_{j+1} = I_j + A_j - s_j$, which is an identity because holding is charged on the ending inventory and that same quantity is carried forward. Summing it over $j = t, \dots, t + L - 1$ telescopes to

$$I_{t+L} = I_t + \sum_{j=t}^{t+L-1} A_j - \sum_{j=t}^{t+L-1} s_j.$$

Every unit counted in X_t arrives in one of the periods $t, \dots, t + L - 1$, and no other unit does, because the order placed at t or later arrives at $t + L$ or later; therefore $\sum_{j=t}^{t+L-1} A_j = X_t$ exactly, and

$$\widehat{I}_{t+L} = I_{t+L} = \mathrm{IP}_t - \sum_{j=t}^{t+L-1} s_j.$$

Substituting $s_j = d_j - \lambda_j$ and collecting,

$$\widehat{I}_{t+L} = \mathrm{IP}_t - \sum_{j=t}^{t+L-1} d_j + \Lambda.$$

Hence

$$U^{\mathrm{proj}} = d_{t+L} - \widehat{I}_{t+L} = d_{t+L} + \sum_{j=t}^{t+L-1} d_j - \mathrm{IP}_t - \Lambda = U^{\mathrm{naive}} - \Lambda,$$

which is (8). Every step is an identity between realized numbers: the recursion, the fact that the in-transit total lands inside the window, the decomposition of sales into demand minus lost demand, and the cancellation. These operations preserve exact equality on the fixed path.

The consequences follow. Since $\lambda_j \geq 0$ we have $\Lambda \geq 0$ pathwise, so $U^{\text{naive}} \geq U^{\text{proj}}$ pathwise and $\mathbb{P}(U^{\text{naive}} > x) \geq \mathbb{P}(U^{\text{proj}} > x)$ for every x, which is first-order stochastic dominance. If T has the stated monotonicity then $T(U^{\text{naive}}) = T(U^{\text{proj}} + \Lambda) \geq T(U^{\text{proj}})$; the mean has it by linearity and monotonicity of the integral, and the ρ-quantile has it because $U^{\text{proj}} \leq U^{\text{naive}}$ pathwise implies $F_{U^{\text{naive}}} \leq F_{U^{\text{proj}}}$ pointwise and hence the corresponding generalized inverses are ordered. Equality almost surely holds if and only if $\Lambda = 0$ almost surely, and $\Lambda = 0$ if and only if $\lambda_j = 0$ for every j in the interval, which is exactly the statement that demand is fully served, so the shelf never empties inside $t, \ldots, t + L - 1$. Taking expectations in (8) gives $\mathbb{E}U^{\text{naive}} - \mathbb{E}U^{\text{proj}} = \mathbb{E}\Lambda$. Finally $\Lambda = \Lambda(L) = \sum_{j=t}^{t+L-1} \lambda_j$ is a sum of nonnegative terms, so $\Lambda(L+1) = \Lambda(L) + \lambda_{t+L} \geq \Lambda(L)$: the quantity is non-decreasing in the length of the interval along a fixed realized trajectory, that is, holding the arrivals and the demands fixed and lengthening the window. This monotonicity concerns the window length on a fixed trajectory as its window length L increases. A change in L in the closed-loop system also changes the policy, the orders and hence the λ_j themselves. □

Remark 3 (order-quantity gaps and $\mathbb{E}\Lambda$). Proposition 8 says the naive shortfall exceeds the projected one by Λ pathwise and by $\mathbb{E}\Lambda$ in mean. The order-quantity gap depends on the quantile or crossing functional used in §3.4; a gap of $\mathbb{E}\Lambda$ applies to the means. For random variables with $U^{\text{naive}} = U^{\text{proj}} + \Lambda$ and $\Lambda \geq 0$, all that holds in general is $0 \leq q_\rho(U^{\text{naive}}) - q_\rho(U^{\text{proj}}) \leq \operatorname{esssup} \Lambda$. Both bounds can be attained: with $U^{\text{proj}} \equiv 0$ and $\Lambda = 2 \cdot \text{Bernoulli}(1/2)$ one has $\mathbb{E}\Lambda = 1$ while the medians of the two shortfalls are both 0, so the difference of the targets is 0 while the mean gap is $\mathbb{E}\Lambda$. Order quantities retain the unconditional ordering; the gap size depends on the functional.

B.5 Proposition 9: exact identification of destroyed orders

The accounting. The benchmark's runner maintains a schedule $\text{sched}[\cdot]$ of units due to land in each future period, and reports to the policy the in-transit total $X_t = \sum_{u \geq t} \text{sched}_t[u]$, evaluated at the decision epoch of t, that is before anything in period t has happened. Within period t the evaluator performs, in this order: (a) the policy places q_t; (b) if $L_t < \infty$, $\text{sched}[t+L_t]$ is incremented by q_t, and if $L_t = \infty$ nothing is scheduled and the order is destroyed; (c) $A_t := \text{sched}[t]$ is delivered to on-hand stock and $\text{sched}[t]$ is set to zero; (d) sales and holding are booked, which do not touch the schedule. Orders scheduled to land beyond the end of the horizon are never delivered but remain in the schedule and hence in X.

Proposition 9 (Exact supply identification)**.** *Under the accounting above, for every period t,*

$$q_t \mathbf{1}\{L_t < \infty\} \;=\; X_{t+1} - X_t + A_t. \tag{9}$$

Hence whenever $q_t > 0$ the indicator $\mathbf{1}\{L_t < \infty\}$ is a function of quantities the policy has observed by the decision epoch of $t+1$, and is identified exactly, with no inference. For $t = T$ there is no such epoch and the last order's fate is never available to the policy, though the identity itself still holds. Under an accounting in which an order with $L_t = \infty$ enters the reported in-transit total and is never removed from it, the same combination equals q_t identically and the indicator is not identified at all.

Proof. Write $S_t(\cdot)$ for the schedule at the decision epoch of t, so $X_t = \sum_{u \geq t} S_t[u]$. Step (b) adds $q_t \mathbf{1}\{L_t < \infty\}$ to the entry at index $t + L_t \geq t$, which is in the range of the sum, including the case $L_t = 0$ where the index is t itself, and including the case $t + L_t > T$ where the entry is never delivered but is still counted. Therefore after (b)

$$\sum_{u \geq t} S[u] = X_t + q_t \mathbf{1}\{L_t < \infty\}\,.$$

Step (c) reads $A_t = S[t]$ at that moment and then zeroes that entry, which removes exactly A_t from the sum and leaves the entries at indices $u \geq t+1$ untouched. Step (d) does not modify the schedule, so the schedule at the decision epoch of $t+1$ satisfies

$$X_{t+1} = \sum_{u \geq t+1} S_{t+1}[u] = X_t + q_t \mathbf{1}\{L_t < \infty\} - A_t,$$

which rearranges to (9).

For the identification claim, the policy knows q_t, which it placed, and observes X_t at the decision epoch of t and both X_{t+1} and A_t at the decision epoch of $t+1$, since the previous period's arrivals are part of the reported state. So the right-hand side of (9) is measurable with respect to the information $\mathcal{H}_{t+1}$ available at that epoch, and when $q_t > 0$ the indicator is recovered as the ratio of the right-hand side to q_t. When $q_t = 0$ the identity reads $0 = 0$ and the indicator is not recovered, which is immaterial: no order was placed, so no trial took place.

The deployed rule tests the ratio against $\frac{1}{2}$ instead of against 1. The two tests agree exactly here, because orders are floored before being placed, so the delivered amount is either the full order or zero and no intermediate value can arise. The threshold is a guard against a fractional order reaching the comparison, not a relaxation of the identity.

For the retaining accounting, suppose instead that step (b) adds q_t to a pending list regardless of L_t and that only finite-lead orders are ever removed, on delivery. Then the reported total X' obeys $X'_{t+1} = X'_t + q_t - A_t$ for every t, so $X'_{t+1} - X'_t + A_t = q_t$ identically. The map from the unobserved indicator to the observed triple (X'_t, X'_{t+1}, A_t) is constant in the indicator, so no function of the observables can distinguish a surviving order from a destroyed one at $t+1$, so the parameter is unidentified, not merely hard to estimate. That accounting also leaves destroyed orders in X' forever, so a rule of the form $q = (S - \mathrm{IP})^+$ is throttled by stock that will never arrive, its subtrahend growing with every destroyed order until the rule falls silent. □

Corollary 10 (the survival probability is a posterior mean over observed trials)**.** *Let $\mathcal{T} = \{t : q_t > 0\}$ and $B_t = \mathbf{1}\{L_t < \infty\}$ for $t \in \mathcal{T}$. By Proposition 9 each B_t is $\mathcal{H}_{t+1}$-measurable. Modeling $\{B_t\}_{t\in\mathcal{T}}$ as independent* $\mathrm{Bernoulli}(\pi)$ *with a* $\mathrm{Beta}(1,1)$ *prior on π, the posterior after the trials completed by period t is* $\mathrm{Beta}(1{+}n_t^{\mathrm{surv}},\ 1{+}n_t^{\mathrm{ord}}{-}n_t^{\mathrm{surv}})$ *with posterior mean $(1{+}n_t^{\mathrm{surv}})/(2{+}n_t^{\mathrm{ord}})$, where n_t^{ord} counts the completed trials and n_t^{surv} the survivals among them.*

Proof. Measurability is Proposition 9. The trial set is chosen by the policy, so the factorization needs one more step: the order is placed before L_t is realized, so $\{t \in \mathcal{T}\} = \{q_t > 0\}$ is $\mathcal{H}_t$-measurable and the trials are selected by a predictable rule, under which the likelihood of $(B_t)_{t\in\mathcal{T}}$ still factorizes. Beta-Bernoulli conjugacy then gives the posterior and its mean. The content of the corollary lies in the observable input: the trials are *observed*, so the estimator applies conjugacy directly to the recovered survival data. Under the retaining accounting of Proposition 9 the B_t are not observable and no such estimator exists. □

B.6 Proposition 11: containment and conditional numerical separation

Proposition 11 (Containment and conditional numerical separation)**.** *Consider decision rules mapping a state with local forecast rate $r > 0$ and inventory position* $\mathrm{IP} \geq 0$ *to an order, and let*

$$\mathcal{B} = \{q = (\theta r - \mathrm{IP})^+ : \theta \geq 0\}, \qquad \mathcal{R} = \{q = cr : c \geq 0\},$$
$$\mathcal{K} = \{q = \min((\theta r - \mathrm{IP})^+,\, cr) : \theta \geq 0,\ c \geq 0\}.$$

Then $\mathcal{B} \subseteq \mathcal{K}$ exactly, by the choice $c = \theta$. For every fixed state, the capped rule with parameters (θ, c) equals the rate rule with parameter c as soon as $\theta \geq c + \mathrm{IP}/r$, and in particular $\min((\theta r - \mathrm{IP})^+, cr) \to cr$ as $\theta \to \infty$. If the inventory positions reachable on an instance are bounded and the local rate is bounded below by $r_{\min} > 0$, which holds on any finite-horizon instance with bounded demand and bounded orders, then $\mathcal{R} \subseteq \mathcal{K}$ on that instance for finite θ. In all cases $\mathcal{B} \cup \mathcal{R} \subseteq \overline{\mathcal{K}}$, and on every instance with $\sum_t |R_t|$ integrable

$$\sup_{\pi\in\mathcal{K}} \mathbb{E}[\textstyle\sum_t R_t(\pi)] \ \geq \ \sup_{\pi\in\mathcal{B}\cup\mathcal{R}} \mathbb{E}[\textstyle\sum_t R_t(\pi)]\,.$$

For a constructed lost-sales instance with $\mathbb{P}(L = \infty) > 0$, the numerical certificate in Appendix C establishes strict reward separation conditional on the validity of its estimated slope bounds. The containment claims are proved below; the construction and certificate are described under Numerical separation.

Proof. For the first claim, take $c = \theta$. Since $\mathrm{IP} \geq 0$ we have $(\theta r - \mathrm{IP})^+ \leq \theta r = cr$, so the minimum is attained by the first argument and the capped rule reproduces the base-stock rule state

by state, not merely in a limit. For the second, if $\theta r - \mathrm{IP} \geq cr$, equivalently $\theta \geq c + \mathrm{IP}/r$, then $(\theta r - \mathrm{IP})^+ \geq cr$ and the minimum is cr; for fixed $r > 0$ and finite IP the left-hand side $(\theta r - \mathrm{IP})^+$ increases without bound as $\theta \to \infty$, so the threshold is eventually met and the pointwise limit is cr. If the reachable positions satisfy $\mathrm{IP} \leq B < \infty$ and $r \geq r_{\min} > 0$, then a single finite $\theta \geq c + B/r_{\min}$ makes the two rules agree at every reachable state, so they generate the same trajectory on every sample path and the same expected reward. The inequality between suprema then holds term by term: every member of $\mathcal{B}$ is a member of $\mathcal{K}$, and every member of $\mathcal{R}$ is either a member of $\mathcal{K}$ under the boundedness condition or a pointwise limit of members of $\mathcal{K}$ whose values agree with it for all large θ. That agreement is pathwise, the threshold θ depending on the path through its bound on IP, so dominated convergence under the integrability hypothesis carries it from trajectories to expected rewards. □

Corollary 12. *On every instance the best member of $\mathcal{K}$ is at least as good as the best member of $\mathcal{B} \cup \mathcal{R}$. The family the search of §3.5 ranges over therefore has a best member at least as good as the first pass's. This comparison concerns the best members of the families. The shipped search resolves each period and returns a softmax-weighted blend of candidate actions, with value differences scaled by the median standard error; its realized performance is evaluated empirically.*

Proof. Immediate from Proposition 11. □

Numerical separation. A constructed instance examines the reward advantage of the larger family. The witness is the constructed instance of §C: constant demand 10, horizon 5, $p = h = 1$, empty initial inventory, lead times drawn independently and uniformly from $\{1, 2, 3, \infty\}$ per period, and the local rate held at the mean demand. The expectation over lead-time sequences is a finite sum over its $4^5 = 1024$ atoms, each evaluated in closed form, so the expected rewards at the three grid optima are exact up to floating-point arithmetic, and the capped grid optimum strictly beats both pure grid optima. Appendix C gives the construction, the enumeration method and the numbers for six configurations, and bounds the two pure families from above on the witness instance to account for grid resolution. Those bounds use slopes measured on a grid. Strict reward separation therefore has a numerical certificate for this construction, conditional on the validity of the slope bounds estimated on the grid.

The mechanism. The construction isolates three ingredients of the arrival-lumping mechanism relevant to the benchmark. Orders are destroyed with positive probability, so a rule that reads the inventory position sees a gap open that no arrival will close. A pure base-stock member responds by re-ordering the entire gap, because that is all the rule can do: its action is a function of the position error alone. When that catch-up order and its successor both survive and their lead times differ by exactly the gap between their placement dates, they land in the same period, and the shelf receives in one period what was meant to cover several. Holding is charged on ending inventory every period until the lump burns off at the demand rate, and at $\rho = 0.5$ one extra period of holding cancels a unit's entire margin, so the lump is expensive in proportion to its size. A pure rate member bounds each period's order by cr independently of inventory position, avoiding position-driven catch-up bursts while continuing to order after a lump arrives for another reason. The capped base-stock rule is the smallest family we know of with both properties, since it is a function of the position gap and of the rate at once and its action is the smaller of the two, so it tracks the level while refusing to place a catch-up lump.

The control configuration isolates the effect of lead-time support. Restrict the lead-time support to $\{1, \infty\}$ and no two surviving orders can ever land together, so the lumping mechanism is unavailable by construction; the separation gap there is exactly zero. Where two distinct finite lead times are available the gap opens in four of the five configurations tested and closes in the fifth, at $\rho = 0.95$ with $T = 5$, where holding is cheap enough that suppressing the catch-up lump costs more than it saves. One further period restores it, so that closure belongs to the horizon at the same ρ. These controlled comparisons support the mechanism within the tested configurations.

B.7 Proposition 13: log-loss bounds for the shape layer

Set-up. Index the shape variants by $g \in \{1, \ldots, G\}$, so that every sum over g below has exactly G terms and the $\log G$ in the statement is exact. The flat variant $\alpha = 0$ is one of these G and $g = 0$

is an alias for it, not an extra member. Assume at every period some variant assigns the observation positive density, $\min_g \mathsf{l}_{t,g} < \infty$, without which the mixture is undefined. Each variant emits, at period t, a scoring density $p_{t,g}$ for the *raw* observation d_t; as specified in §3.3 this includes the Jacobian $-\log \eta_g[t]$ of the deseasonalization, which makes the G numbers densities of one and the same observable and therefore comparable, for the reason given in Lemma 6. This subsection works in log-loss units, which are the negative of the log-weight increments of (11); the two therefore carry different symbols and are never interchanged. Write $\mathsf{l}_{t,g} = -\log p_{t,g}(d_t)$ for the log loss of variant g at t, $L_{t,g} = \sum_{s\le t} \mathsf{l}_{s,g}$ for its cumulative log loss, and $\widetilde{L}_{t,g} = \sum_{s\le t} \gamma^{t-s} \mathsf{l}_{s,g}$ for the discounted version. The policy maintains $\log V_g \leftarrow \gamma \log V_g + \log p_{t,g}(d_t)$ and then subtracts the row maximum, which is a shift common to all variants and therefore invisible to the softmax, so the weights it uses at t are

$$w_{t,g} = \frac{\exp(-\widetilde{L}_{t-1,g})}{\sum_{g'} \exp(-\widetilde{L}_{t-1,g'})}. \tag{10}$$

The scoring mixture density is $\bar{p}_t = \sum_g w_{t,g} p_{t,g}$, with log loss $\mathsf{l}^{\mathrm{mix}}_t = -\log \bar{p}_t(d_t)$. The path sampler uses the same variant weights; its within-variant sampling distribution is specified separately below.

The bound below applies to the scoring density. Within a variant, members are scored under a single Student-t of scale $\sqrt{\tau_m^2 + \sigma_m^2}$, whereas paths are drawn as a convolution of two independent t variates and then truncated at zero and at the cap. The scoring density and path-sampling distribution are therefore distinct. The bound controls the scoring mixture's log loss; it does not directly bound the quantiles of the sampled demand paths.

Proposition 13 (Log-loss bounds for the shape layer)**.** (i) *Assume* $\min_g \mathsf{l}_{t,g} < \infty$ *at every* t*. With undiscounted weights,* $\gamma = 1$*, so that* $w_{t,g} \propto \exp(-L_{t-1,g})$ *with* $w_{1,g} = 1/G$*, the mixture is the Bayesian mixture under log loss and for every horizon* T *and every data sequence*

$$\sum_{t=1}^{T} \mathsf{l}^{\mathrm{mix}}_t \;\le\; \min_g L_{T,g} + \log G.$$

In particular, since the flat variant is a member of the set and its scoring density is the core forecaster's, the layer's cumulative log loss exceeds the core's by at most $\log G$ *nats, a constant independent of* T*.*

(ii) *With the discounted update actually used,* $\gamma < 1$*, suppose the flat variant's single-period excess log loss is bounded,* $\mathsf{l}_{s,0} - \min_g \mathsf{l}_{s,g} \le \delta$ *for all* s*. Then for every* t

$$w_{t,0} \;\ge\; G^{-1} \exp\big(-\delta/(1-\gamma)\big), \qquad \mathsf{l}^{\mathrm{mix}}_t - \mathsf{l}_{t,0} \;\le\; \log G + \frac{\delta}{1-\gamma}.$$

Proof. (i) Let $W_t = \sum_g \exp(-L_{t,g})$ with $W_0 = G$. Then

$$\frac{W_t}{W_{t-1}} = \frac{\sum_g \exp(-L_{t-1,g}) \exp(-\mathsf{l}_{t,g})}{\sum_{g'} \exp(-L_{t-1,g'})} = \sum_g w_{t,g}\, p_{t,g}(d_t) = \bar{p}_t(d_t) = \exp(-\mathsf{l}^{\mathrm{mix}}_t),$$

where the second equality is the definition of $w_{t,g}$ at $\gamma = 1$ together with $\exp(-\mathsf{l}_{t,g}) = p_{t,g}(d_t)$. Telescoping over $t = 1, \dots, T$, $\log W_T = \log G - \sum_t \mathsf{l}^{\mathrm{mix}}_t$. On the other hand $W_T = \sum_g \exp(-L_{T,g}) \ge \exp(-\min_g L_{T,g})$, because the sum dominates each of its nonnegative terms, so $\log W_T \ge -\min_g L_{T,g}$. Combining the two displays gives $\sum_t \mathsf{l}^{\mathrm{mix}}_t \le \min_g L_{T,g} + \log G$. The bound holds for every sequence, with no assumption on how the data are generated, and the constant $\log G$ is exact, not an order of magnitude. Applying it with the flat variant in place of the minimizer, and using that variant $\alpha = 0$ deseasonalizes by the constant shape $\eta^0 \equiv 1$ and therefore has exactly the core forecaster's scoring density, gives the second sentence.

(ii) Fix t. For the first step, by definition of the mixture,

$$\mathsf{l}^{\mathrm{mix}}_t - \mathsf{l}_{t,0} = -\log\Big(\sum_g w_{t,g} \exp(-\mathsf{l}_{t,g})\Big) + \mathsf{l}_{t,0} = -\log \sum_g w_{t,g} \exp(\mathsf{l}_{t,0} - \mathsf{l}_{t,g}) \;\le\; -\log w_{t,0},$$

where the inequality keeps only the $g = 0$ term of the sum, whose exponential factor is 1, and uses that $-\log$ is decreasing.

For the second step, from (10),

$$-\log w_{t,0} = \widetilde{L}_{t-1,0} + \log \sum_g \exp(-\widetilde{L}_{t-1,g}) \;\le\; \widetilde{L}_{t-1,0} + \log\Big(G \exp(-\min_g \widetilde{L}_{t-1,g})\Big)$$
$$= \log G + \Big(\widetilde{L}_{t-1,0} - \min_g \widetilde{L}_{t-1,g}\Big),$$

bounding each of the G terms of the sum by the largest. For the discounted excess, exchange the minimum and the sum using $\min_g \sum_s a_{s,g} \ge \sum_s \min_g a_{s,g}$, valid because each summand is at least its own minimum over g:

$$\widetilde{L}_{t-1,0} - \min_g \widetilde{L}_{t-1,g} \;\le\; \sum_{s\le t-1} \gamma^{t-1-s} \mathsf{l}_{s,0} - \sum_{s\le t-1} \gamma^{t-1-s} \min_g \mathsf{l}_{s,g}$$
$$= \sum_{s\le t-1} \gamma^{t-1-s} \big(\mathsf{l}_{s,0} - \min_g \mathsf{l}_{s,g}\big) \;\le\; \frac{\delta}{1-\gamma},$$

the last step using the hypothesis termwise and $\sum_{k\ge 0} \gamma^k = 1/(1-\gamma)$. Substituting into the previous display gives $-\log w_{t,0} \le \log G + \delta/(1-\gamma)$, which with the first step is the per-period bound. Exponentiating the same inequality gives the weight floor $w_{t,0} \ge G^{-1} \exp(-\delta/(1-\gamma))$. □

Remark 4 (adaptation and per-period guarantees under discounting). Part (ii) is a per-period bound and summing it over the horizon yields $T(\log G + \delta/(1-\gamma))$, which grows linearly in T. Discounting reduces the influence of older observations so that variants can gain weight as their fit improves. Alternating which variant fits best can incur a switching cost every period, so excess over a fixed comparator can grow with the horizon. For $\gamma < 1$, the result established here is the conditional per-period bound and the accompanying weight floor, which keeps the flat variant available after a run of bad periods.

Remark 5 (scoring-density equivalence and sample-path reproducibility). The flat variant reproduces the core forecaster's scoring density exactly, as required by Proposition 13. Its random draws follow a separate sampling stream: selecting a variant consumes randomness before the bank is sampled, and each variant runs a private copy of the bank with its own seed, so even when the flat variant carries essentially all of the weight the realized sample paths differ from those of the unlayered forecaster. Bit-identity holds only where the layer is absent entirely, which is the case on the synthetic instances: they carry no product description, no shape is requested, the shape axis is never instantiated, and the forecaster object is left untouched. Bit-identity in this case follows directly from the construction.

C Numerical verification of Proposition 11

Proposition 11 establishes containment, $\mathcal{B} \cup \mathcal{R} \subseteq \overline{\mathcal{K}}$, with $\mathcal{B} \subseteq \mathcal{K}$ exactly. The finite-horizon instance below examines strict reward separation through exhaustive enumeration and a numerical certificate conditional on the validity of estimated slope bounds. Every reported value describes this construction; benchmark performance is evaluated in the experiments.

C.1 The instance

A single product at a single location, starting empty, over a horizon of T periods. Demand is constant at 10 units per period, so the three families use the same known demand rate: the local rate r used by all families is fixed at the mean demand, $r = 10$, in every period and for every candidate. The unit profit and unit holding cost are stated per configuration below. The lead time L_t is drawn independently in each period from a stated support that includes ∞, uniformly over the support; a draw of ∞ destroys the order, which never arrives and never enters the in-transit total.

The period runs in the benchmark's order: the policy places its order; if the lead time is finite the order is scheduled to land at $t + L_t$; arrivals scheduled for t are added to on-hand stock; demand is served from stock and unmet demand is lost; holding is charged on the ending inventory. An order

Table 3: Expected reward at each family's best grid point, by configuration. The gap is the best capped value minus the better of the two pure values, and the last column expresses it as a fraction of the perfect-foresight revenue $p \sum_t d_t$ of the same instance. All values are exact sums over the enumerated lead-time sequences.

lead support	(p, h)	T	paths	pure $\mathcal{B}$	pure $\mathcal{R}$	capped $\mathcal{K}$	gap
$\{1, \infty\}$	$(1, 1)$	6	64	17.812500	25.000000	25.000000	0.000000
				$\theta = 1.0$	$c = 1.0$	$\theta = 2.0,\ c = 1.0$	0.00%
$\{1, 2, \infty\}$	$(1, 1)$	6	729	17.325103	22.510288	23.168724	0.658436
				$\theta = 1.0$	$c = 1.0$	$\theta = 2.0,\ c = 1.0$	1.10%
$\{1, 2, \infty\}$	$(4, 1)$	6	729	113.415638	107.572016	120.123457	6.707819
				$\theta = 3.0$	$c = 1.0$	$\theta = 3.0,\ c = 2.0$	2.79%
$\{1, 2, 3, \infty\}$	$(1, 1)$	5	1024	11.914062	14.921875	16.523438	1.601562
				$\theta = 1.0$	$c = 1.0$	$\theta = 2.0,\ c = 1.0$	3.20%
$\{1, 2, \infty\}$	$(19, 1)$	5	243	525.802469	500.246914	525.802469	0.000000
				$\theta = 3.0$	$c = 2.0$	$\theta = 3.0,\ c = 3.0$	0.00%
$\{1, 2, \infty\}$	$(19, 1)$	6	729	682.139918	654.074074	688.230453	6.090535
				$\theta = 4.0$	$c = 2.0$	$\theta = 4.0,\ c = 3.0$	0.53%

scheduled beyond the end of the horizon is never delivered. The three families are exactly the pure base-stock, pure rate and capped base-stock families of §3.5, with the pure limits realized by setting the inactive parameter to 10^6. Orders are real-valued here, so all three families are evaluated as continua under the same convention, with integer rounding disabled.

C.2 How the expectation is computed

The expectation is over the lead-time sequence alone, and the horizon is short enough that this expectation is a finite sum. With a support of size S and horizon T there are S^T sequences; we enumerate all of them, weight each by its probability, and evaluate the resulting trajectory in closed form. The reported values and their differences are finite sums, exact up to floating-point arithmetic and free of Monte Carlo error.

Each family's optimum is taken over a grid of step 0.1 on $[0, 6]$ in each of its parameters, so the capped family is searched over a 61×61 grid and each pure family over 61 points; the two configurations at $(p, h) = (19, 1)$ use step 0.01 on $[0, 12]$, the gap there being the small quantity the counterexample turns on.

A grid maximum lower-bounds the supremum of a family. Assessing separation also requires upper bounds for the pure families to account for search resolution. On the witness instance we compute these bounds by re-evaluating each pure family on a grid of step 0.002 over $[0, 40]$, fifty times finer and more than six times wider, which leaves both optima unchanged in value and in location. Since the value is piecewise linear in the parameter, we estimate slope bounds by the largest slopes on that grid: 26.76 for the base-stock family and 47.50 for the rate family. If these slope bounds are valid, each supremum on $[0, 40]$ is bounded by its grid maximum plus its estimated slope bound times half a step, giving 11.940820 and 14.969375. Past the interval both curves fall away, being already below -990 at 40 and below -2.6×10^7 at 10^6, as they must once ordering beyond demand buys only holding cost. The best capped member attains 16.523438, giving a separation margin of 1.554062 against the bound. This numerical certificate uses slopes measured on the stated grid; the separation is conditional on the validity of those estimated slope bounds. Its scope is numerical verification of the witness instance.

C.3 Results

Table 3 reports six configurations. The fourth row is the benchmark's own lead-time support and is the witness cited in Proposition 11: the best pure base-stock grid point attains 11.914062, the best pure rate grid point 14.921875, and the capped member with $\theta = 2$ and $c = 1$ attains 16.523438, so the best grid point of $\mathcal{K}$ beats the best grid point of $\mathcal{B} \cup \mathcal{R}$ by 1.601562, which is 3.20% of the

Slot	Shipped	What it decides	Authored by
Forecaster	two-space bank	The model bank and how its members are weighted (§3.2)	Human
Shape	on	Whether a shape read from the product text defines extra forecast variants (§3.3)	LLM, design time
Search	capped base-stock	Which family of policies the stochastic cells search over (§3.5)	LLM, design time

Table 4: The three slots. The frozen core of §3.1 is one setting of all three and appears as a row in the same ablation.

perfect-foresight revenue of 50 on that instance. This is the numerical strict-separation witness for Proposition 11, under the slope-bound condition stated above.

The other five rows examine the conditions behind the gap. The first is a control: with support $\{1, \infty\}$ every surviving order lands exactly one period after it is placed, so two surviving orders can never land in the same period and the lumping mechanism the design argument invokes is unavailable by construction. The separation gap there is exactly zero and the best capped member only matches the pure rate optimum. The second row differs from it in the lead-time support alone, holding the costs and the horizon fixed; admitting a second finite lead time opens the gap. This controlled change provides evidence for the arrival-lumping mechanism. Across the five configurations whose support contains two distinct finite lead times, so that a catch-up order and its successor can arrive together, the gap is strictly positive in four and exactly zero in the fifth, the last two rows being that case: at $\rho = 0.95$ it closes exactly at $T = 5$ and one further period restores it. Across the six constant-demand configurations examined here, positive separation occurs only when the lead-time support permits surviving orders to arrive together. The configuration at $\rho = 0.95$ and $T = 5$ shows that this possibility alone does not ensure a positive gap.

These constructed instances have constant demand, a five- or six-period horizon and a known rate. The witness provides the conditional numerical certificate of strict reward separation in Proposition 11. Performance on the benchmark is evaluated separately in §4.

D Construction details of the policy layers

This appendix gives the construction details needed to reproduce the layers in §3, following their order in the main text.

D.1 The three slots

Table 4 names the three slots, gives the shipped setting of each, and identifies its author. We designed the forecaster and the per-period arithmetic of the decision rules. The language model contributed a mixture axis over the bank and a policy-family choice for the stochastic branch.

D.2 The predictive model

One instantiation of the bank $\mathcal{M}$ holds nine members: the mean of the whole history; three exponentially weighted moving averages with different smoothing constants; a Holt linear-trend member; two rolling-window means; an AR(1) fitted by the sample lag-one autocorrelation; and a seasonal member whose period is picked online by autocorrelation and re-picked as history accumulates. Each member uses fixed configuration constants and is refit on the full history each period. Member m is scored using a Student-t predictive density with ν degrees of freedom, centered on its point forecast $\mu_{m,t}(j)$ at forecast offset j. Its scale combines parameter uncertainty τ_m with innovation noise $\sigma_m\sqrt{1 + g_m j}$, growing at the rate implied by the member's own state-space form, $g_m = 0$ for the stationary members and a_m^2 for a smoothing member with constant a_m. That growth is what keeps the tails of the $L + 1$-period cumulative from coming out too thin. The member constants, the degrees of freedom in each space and the seasonal period detection are in Appendix E.1.

Members are scored on the raw observation, so a member fitted on $z = \phi_m(d)$ contributes its transformed log-density together with the Jacobian of ϕ_m, which is the correction §3.2 discusses

and Lemma 6 proves. The discounted update and the softmax that turns it into weights are

$$\ell_{t,m} \;\leftarrow\; \gamma\,\ell_{t-1,m} + \log f_m\big(\phi_m(d_t) \mid \mathcal{H}_t\big) + \log \phi'_m(d_t), \qquad w_{t,m} = \frac{\exp(\ell_{t,m})}{\sum_{m'} \exp(\ell_{t,m'})}, \tag{11}$$

with $\log \phi'_m(u) = 0$ for identity members and $-\log(1+u)$ for log1p members.

The forgetting factor γ of (11) is shared by the bank and by the shape-variant weights of (13), so a single constant sets the memory of both levels of the hierarchy; its value and the running renormalization of the log-weights are in Appendix E.2. The bank is instantiated in both transform spaces on the deterministic cells and in identity space alone on the stochastic cells, where the search layer consumes a local demand rate.

The path-sampling distribution uses the same mixture weights as the scoring density. A sample path is generated by drawing a member from these weights, one level perturbation shared by the whole path and an independent innovation per step, and then inverting that member's transform,

$$d^{(k)}_{t+j} = \phi_m^{-1}\Big(\mu_{m,t}(j) + \tau_m \xi^{(k)} + \sigma_m \sqrt{1 + g_m j}\;\varepsilon_j^{(k)}\Big), \qquad m \sim w_{t,\cdot}, \tag{12}$$

for $k \le K$ and $j < H$, with independent Student-t draws $\xi^{(k)}$ and $\varepsilon_j^{(k)}$. Values are clipped below at zero and above at a level far past anything the series has done, since inverting log1p on one heavy-tailed innovation can produce a draw that moves an upper quantile of the H-period sum by itself. The loose clip limits the influence of such extreme draws on the quantile. Appendix E.2 gives it.

D.3 The shape layer

Variant g deseasonalizes by $\eta_g = \eta^{\alpha_g}$, renormalized to mean one, and runs a private copy of the bank on $d_t/\eta_g[t]$. Its weight is updated by the discounted predictive likelihood of §3.3,

$$\log V_g \;\leftarrow\; \gamma \log V_g + \log p_g(d_t \mid \mathcal{H}_t), \qquad p_g(d_t \mid \mathcal{H}_t) \;=\; \frac{1}{\eta_g[t]}\,\tilde{p}_g\!\left(\frac{d_t}{\eta_g[t]} \;\middle|\; \mathcal{H}_t\right), \tag{13}$$

with $\tilde{p}_g$ the variant's own model-averaged scoring density in deseasonalized space and the factor $1/\eta_g[t]$ the Jacobian of Lemma 6, which puts variants that deseasonalize by different amounts on one scale.

The elicitation query goes to DeepSeek-V4-Flash, and its user message carries exactly what the official evaluation protocol gives a policy at initialization: the product text, the week dates and the five training demands. These fields constitute the entire instance-level input. The system prompt also names the retailer, the market and the calendar year, and it supplies priors we wrote by hand, a table of category seasonal peaks and the year's retail calendar. Those priors are shared across products and encode category and calendar information. The elicitation instructions require a normalized, nonnegative annual profile and exclude any use of observed evaluation demand.

Replies pass a validation gate before they are used. A reply is accepted only if it parses as strict JSON carrying one finite nonnegative entry per week of the year with a mean that is not degenerate, and an accepted shape is divided by its own mean. The accepted samples are aggregated by their per-week median, reducing sensitivity to an extreme weekly value in any one sample. The aggregate is then clipped, renormalized to mean one, and frozen before any policy runs. A product for which no sample survives the gate receives the all-ones shape, making a parse failure equivalent to switching the layer off for that product. The number of samples, the gate thresholds and the clips applied at build time and at run time are in Appendix E.3.

At run time a sampled path draws a variant from the normalized variant weights, a member from that variant's own bank, and scales by that variant's shape, so the hierarchy is resolved by sampling at both levels. The layer has constants of its own, the number of variants and the sampling and clipping choices among them, fixed at the values in Appendix E.3. Prior strength is handled by online competition among the variants; as §3.3 explains, selecting a strength on held-out data would require product text and calendar information absent from our validation set.

D.4 THE DETERMINISTIC RULE

The depletion multiplier κ of (3) is an empirical correction to the cost-based marginal value. In the shipped policy it is

$$\kappa \;=\; \min\{1 + \kappa_1 \, L \, \mathrm{cv}^{\,\kappa_2},\;\; \bar{\kappa}\}\,, \tag{14}$$

with cv the coefficient of variation of the policy's own observed demand history. Each of the two drivers carries one of the patterns the correction is meant to absorb. The lead time scales the increment above the additive term because the number of periods a surplus must be sat on before the next order can respond to it grows with the lead time; at $L = 0$ that increment vanishes and $\kappa = 1$, the constant base term. The coefficient of variation enters because the surplus a given target leaves behind grows with the dispersion of the predictive, so the wait the marginal unit is charged for grows with it too. The cap $\bar{\kappa}$ limits the multiplier's response to extreme observed dispersion.

Only κ_1 and κ_2 were fitted, on the validation set. The additive term is a constant base contribution. The cap $\bar{\kappa}$ and the clip applied to cv guard against extreme values. The functional form itself was chosen after seeing that a single global constant was mis-tuned in a patterned way across the test cells, which is the first of the two leakage channels §4.1 sets out. Thus test aggregates informed the functional form, while validation instances supplied the coefficient fit. Values are in Appendix E.4, which also gives the path count, the bisection bracket and the number of bisection steps the root search uses.

D.5 THE STOCHASTIC SEARCH

Searching policy parameters makes the rollout value consistent with each candidate's continuation rule. A one-step rollout assigns to a candidate action the value of that action plus the value of whatever continuation rule the simulation assumes, so the choice of continuation decides the ranking: a continuation that floods every simulated future with stock covers the very demand the candidate was meant to cover and drives its marginal effect below the sampling noise of the path set. Simulating the remaining horizon under the same rule the candidate defines removes that free choice, at the cost of restricting the search to a parametric family.

Rollouts use two quantities constructed from the reported state. The first is the order survival probability. We fit a Beta-Bernoulli model to the survival indicators identified by Proposition 9 and estimate the probability by its posterior mean. A positive floor keeps the survival estimate used in rollouts bounded away from zero. The second is the committed pipeline. The benchmark reports in transit as a scalar total with no age breakdown, while a rollout needs to know when those units land, so the total is split across the finite lead ages in proportion to the surviving order placed that many periods ago times the probability that its lead is at least that long, conditional on the lead being finite. Conditioning on finite lead time keeps the arrival-age calculation consistent with the survival accounting. Each age is then given a residual lead drawn uniformly from the lead values still consistent with it. Appendix E.5 gives the clip, the split and the coverage window the local rate is computed over.

The search is two passes. A coarse pass evaluates the pure candidates of the first pass's two families on a fixed grid, each realized inside $\mathcal{K}$ by setting the other parameter to a numerical stand-in for the pure limit of Proposition 11, which identifies a good θ and a good c separately. A refinement pass then brackets each of those two winners between its neighboring grid points and explores the interaction on a small product grid, carried together with the pure limits, so every action that earlier search could have played remains reachable at every period. The grids, their sizes and the number of sampled paths per candidate are in Appendix E.5.

The blend that turns candidate values into one order is a softmax over value gaps divided by a scale, with the scale set to the median standard error across the candidates, so the gaps are read in units of sampling noise and no tuning constant has to be rescaled per instance. The scale is floored away from zero so that the blend remains defined when every candidate returns the same estimate, and the executed order is the floor of the weighted mean of the candidate actions. Hedging across an expert set is a standard device (Cesa-Bianchi & Lugosi, 2006); its role here is empirical variance control, with performance assessed through the rollout search.

D.6 THE DESIGN-TIME LOOP

The method includes a diagnostic switch that substitutes true future demand for the forecast. The proposal screen enforces the online information boundary. Every proposal is inspected for that switch, and for any other route to a future observation, before it is run, and what the screen rejects is recorded in the research record alongside the proposals that ran.

The harness draws exclusively from the validation generator, keeping candidate scoring separate from benchmark test instances. Each proposal is scored against the current best configuration, so acceptance measures an incremental gain over the improvements already retained.

The second research pass started from the best composition found so far and received the first pass's full record, including every proposal and its score, whether accepted or rejected. The record also explained that four separately credited first-pass wins were four implementations of one idea in the same slot. Its context included the control that added per-period language-model order adjustments: estimated increments were negative on every deterministic cell and positive on both stochastic cells. This information let the second pass build on accepted ideas and focus on the cells where the earlier control suggested room for improvement.

D.7 COMPOSITION

The loop scores each candidate once against the configuration current when it was written. To assess additivity and attribute gains to individual components, we re-express the proposals as settings of independent slots and enumerate their product, which is possible because the accepted candidates modify disjoint parts of the policy, one group replacing the model bank and another the family the stochastic branch searches over. The original candidates and frozen core are preserved, with the core entering the ablation as a literal setting of the three slots. This construction supports the factorial interpretation of Figure 4.

A candidate takes over the parent policy's forecaster only after the parent has built and seeded its own, so a candidate that arrived with a forecaster alongside its search layer can still be used as a pure search layer, and the two contributions credited separately.

Three consistency checks precede the sweep. First, a variant's own mixture scoring density is recomputed from a per-member quantity the frozen forecaster also carries, so it has to be revalidated against every forecaster setting it is used with, including the ones that add members; the check is that the member set reconstructed there matches that setting's own. Second, the shape layer must be inert on the validation set, no validation instance carrying product text, and the orders emitted with and without the layer are required to agree entry by entry. Third, the two research passes number their rounds independently, so each search layer is identified by both its pass and round index. This identifies the intended candidate when the passes reuse a round number.

E CONFIGURATION DETAILS

This appendix specifies the shipped policy's configuration. Empirically chosen constants were selected on the validation set of §4 and then frozen. Displayed time indices follow the paper's convention, $t = 1, \ldots, T$. Writing the code's zero-based index as t_0, we have $t_0 = t - 1$.

E.1 THE MODEL BANK

One instantiation of the bank holds nine members. Each member is refit on the full history each period using the parameters prescribed by its model form, and emits a point forecast $\mu_{m,t}(j)$ at offset $j \geq 0$, a parameter-uncertainty scale τ_m and an innovation scale σ_m. Residual scales are floored at 10^{-6}. Table 5 lists the members with their constants and with the innovation growth rate g_m that sets how the predictive widens with the offset, the scale at offset j being $\sigma_m\sqrt{1 + g_m j}$.

The seasonal member's period is chosen online. Candidate lags run over $\{4, 5, \ldots, 14\}$; a candidate P is admissible only if the history is longer than $P + 3$, the criterion is the sample autocorrelation of the mean-centered history at lag P, and the default when no candidate is admissible is 7. Detection is repeated every fifth observation as history accumulates, and the seasonal member uses the median

Table 5: The nine members of one bank. m is the number of observations in the history, sd its sample standard deviation, σ_m the member's own one-step residual scale. The growth rate g_m inflates the j-step scale by $\sqrt{1+g_m j}$. For the local-level members it is the value implied by the state-space form, a^2 with smoothing constant a. The remaining entries are chosen approximations with the following errors: Holt's j-step variance grows superlinearly, which 0.25 cannot reproduce and at $j=1$ overstates; and AR(1) with ϕ near the clip does accumulate variance in j, which 0 ignores.

member	constants	τ_m	g_m
history mean	—	$\mathrm{sd}/\sqrt{m}$	0
EWMA $\times 3$	$a \in \{0.15, 0.35, 0.70\}$	$\sigma_m\sqrt{a/(2-a)}$	a^2
Holt linear trend	$\alpha_{\mathrm{H}} = 0.3$, $\beta_{\mathrm{H}} = 0.1$	$0.6\,\sigma_m$	0.25
rolling mean $\times 2$	window $w \in \{4, 8\}$, truncated to m	$\mathrm{sd}_w/\sqrt{w}$	0
AR(1)	ϕ = lag-1 autocorrelation, clipped to $[-0.95, 0.95]$	$\sigma_m/\sqrt{\max(m,2)}$	0
seasonal	period from Table 6	$\sigma_m/\sqrt{m}$	0

of the detected periods. The evaluator scores one instance at a time, so that median is taken over a single element and the period is the instance's own.

Member m's one-step scoring density is a Student-t centered at $\mu_{m,t}(0)$ with scale $\sqrt{\tau_m^2+\sigma_m^2}$ and ν degrees of freedom, with ν shared within each space: $\nu = 4$ in identity space and $\nu = 14$ in $\log(1+y)$ space. The two differ because the transform already supplies the right tail. Inverting $\log(1+y)$ with $\exp(\cdot) - 1$ turns a symmetric innovation into a right-skewed raw-space predictive, so stacking a $\nu = 4$ innovation on top of that inversion can produce extreme upper-tail draws. In identity space, the small degrees of freedom directly represent the heavy tail of weekly retail demand.

E.2 Weights and sample paths

The bank is instantiated in both spaces on the deterministic cells, $\mathcal{E} = \{\mathrm{id}, \mathrm{log1p}\}$ and $M = 18$, and in identity space alone on the stochastic cells, $M = 9$. The search layer of §3.5 uses the rate supplied by this identity-space bank. Log-weights are updated by the discounted log-likelihood recursion of §3.2 with forgetting factor $\gamma = 0.92$, an effective memory of $1/(1-\gamma) = 12.5$ periods, and are renormalized each period by subtracting the running maximum, which is a shift common to all members and therefore invisible to the softmax.

A sample path is drawn by selecting a member index from the current mixture weights, drawing one level perturbation ξ shared by the whole path, drawing an independent innovation ε_j at each step, and inverting the member's transform, as set out in §D.2. Both ξ and ε_j are Student-t on the selected member's own degrees of freedom. For a log1p member the exponent is clipped at 20 before inversion. Every path is then clipped below at zero and above at

$$6\left(\max_t d_t + 3\,\mathrm{sd}(d) + 10\right), \tag{15}$$

the maximum and standard deviation being taken over the observed history. The clip lies well above the observed range and limits the influence of a single inverted heavy-tailed draw on an upper quantile of the H-period sum.

E.3 The seasonal shape layer

At design time DeepSeek-V4-Flash is queried 3 times per product, at temperature 1.0 with a 2200-token limit. A reply is accepted only if it parses as strict JSON carrying exactly 52 finite nonnegative entries whose mean is at least 0.2, and an accepted shape is divided by its own mean. The three accepted samples are reduced by a per-week median to reduce sensitivity to an outlying draw, then clipped to $[0.05, 20]$ and renormalized to mean one. A product for which no sample passes the format checks receives the all-ones shape. The 52 weeks span the five training weeks and the 47 scored weeks of a real instance.

At run time the stored table is clipped to $[0.12, 8.0]$ and renormalized to mean one before use. The $G = 3$ variants use exponents $\alpha \in \{0, 1/2, 1\}$, and each tempered shape η^α is again renormalized to mean one and re-clipped to $[0.12, 8.0]$ to keep the tempered variant within the same range. Variant

weights use the same $\gamma = 0.92$ as the bank. Each variant runs a private copy of the bank with its own seed offset, and the variant sampler carries a further independent stream. The flat variant therefore shares the core forecaster's predictive law while using independent draws (Remark 5). Beyond week 52 the shape is padded with ones.

E.4 The deterministic rule

The closed-form rule of §3.4 draws $K = 128$ sample paths of length $H = L+1$, rolls the committed pipeline forward along each path under the evaluator's own lost-sales recursion to obtain $\widehat{I}_{t+L}$, and solves the marginal condition of §3.4 on the resulting sample of shortfalls. The root is found by bisection, 48 iterations on the initial bracket $[0,\ 1.05\max\{\max_k U^{(k)},\ 1\}]$, which contains the root by part (v) of Proposition 7 and leaves a residual bracket of 2^{-48} of its width. No order is placed once $t + L > T$, since the order would arrive after the last scored period.

The depletion multiplier is $\kappa = \min\{1+\kappa_1\, L\, \mathrm{cv}^{\,\kappa_2},\ \bar{\kappa}\}$ with $\kappa_1 = 1.5$, $\kappa_2 = 1.0$ and $\bar{\kappa} = 12$, where cv is the coefficient of variation of the policy's own observed demand history, clipped to $[0.05, 1.5]$. The coefficients κ_1 and κ_2 were fitted on validation; the floor, the cap and the clip range are fixed numerical guards that remain inactive on typical instances.

E.5 The stochastic search

Each period the search evaluates candidate policies by simulating the whole remaining horizon, $H = T - t + 1$ periods including the current period, on $K = 40$ sampled demand paths. All candidates share the same paths, the same per-period survival draws and the same per-period lead draws, so candidates are compared on identical futures. Leftover stock at the end of the simulated horizon receives no credit. Orders inside the rollout are floored exactly as the evaluator floors them.

The pass is coarse then refined. The coarse pass evaluates the 24 pure candidates of the first pass's two families,

$$\theta \in \{0,\ 0.4,\ 0.8,\ 1.2,\ 1.6,\ 2.0,\ 2.5,\ 3.0,\ 3.6,\ 4.3,\ 5.2,\ 6.5\},$$
$$c \in \{0,\ 0.15,\ 0.3,\ 0.45,\ 0.6,\ 0.75,\ 0.9,\ 1.1,\ 1.35,\ 1.7,\ 2.1,\ 2.6\},$$

each realized inside the capped base-stock family of §3.5 by setting the other parameter to a sentinel of 10^5, which is the pure limit of Proposition 11 in numerical form. The refinement pass takes the best θ and the best c of the coarse pass and brackets each between its two neighboring grid points, sampling 4 points across that bracket. The refined candidate set is the 4×4 interaction grid together with the 4 pure base-stock and 4 pure rate limits, again 24 candidates, so every action the earlier search could have played remains reachable at every period.

Both θ and c multiply the local forecast rate. The rate at simulated step j is the forward mean of the predictive mean path over $[j,\ j + \mathrm{cov})$, truncated at the horizon, with the coverage window $\mathrm{cov} = \mathrm{round}(1 + \overline{L}) = 3$ periods for the finite lead support $\{1, 2, 3\}$, that is one period plus the mean finite lead.

Supply is handled by two devices. The survival probability is the Beta-Bernoulli posterior mean of Corollary 10, $(1 + n^{\mathrm{surv}})/(2 + n^{\mathrm{ord}})$, clipped to $[0.2, 1]$, and it drives the survival draws inside the rollout. The committed pipeline is not observable as an age profile, only as a scalar total, so it is split across ages $1, \ldots, 3$ in proportion to the surviving order placed a periods ago times $\mathbb{P}(L \geq a \mid L < \infty)$, the conditioning being required because destruction is already accounted for by counting only surviving orders. Each age is given a residual lead drawn uniformly from the lead values that are still consistent with it.

The executed order is a softmax-weighted blend of candidate actions, with value gaps scaled by the median candidate standard error. With v_g the estimated value of candidate g and $\mathrm{se}_g = \mathrm{sd}_g/\sqrt{K}$ its standard error, the weights are a softmax of $(v_g - \max_{g'} v_{g'})/(\varsigma\, \mathrm{se}^{\mathrm{med}})$ with temperature $\varsigma = 1$ and scale $\mathrm{se}^{\mathrm{med}} = \mathrm{median}_g\, \mathrm{se}_g$, floored at 10^{-9}; the order is the floor of the weighted mean of the candidate actions. The weights concentrate on the highest-valued candidates when their estimates are well separated relative to this scale, and spread across candidates with similar estimates. The standard-error scale adapts the blend to each instance. No order is placed once $t + \min L > T$.

Table 6: Numerical constants, collected. Values are read from the shipped configuration.

quantity	symbol	value
EWMA smoothing constants	a	$0.15,\ 0.35,\ 0.70$
Holt smoothing constants	—	0.3 (level), 0.1 (trend)
rolling window lengths	w	$4,\ 8$
seasonal period candidates	P	$4, \ldots, 14$; default 7; re-detected every 5 periods
degrees of freedom, identity	ν	4
degrees of freedom, log1p	ν	14
forgetting factor	γ	0.92
innovation growth	g_m	0; a^2 for EWMA; 0.25 for Holt
sample-path upper clip	—	$6(\max d + 3\,\mathrm{sd}(d) + 10)$
paths, deterministic rule	K	128
paths, stochastic search	K	40
bisection iterations	—	48
depletion coefficients	κ_1, κ_2	1.5, 1.0
depletion cap, cv clip	$\bar{\kappa}$	12; $\mathrm{cv} \in [0.05, 1.5]$
survival prior, clip	π	$\mathrm{Beta}(1,1)$; posterior mean clipped to $[0.2, 1]$
coverage window	cov	3
refinement points per axis	—	4 (4×4 interaction plus $4 + 4$ pure limits)
hedging temperature, scale	ς	1; median candidate standard error
shape variants	α	$0,\ 1/2,\ 1$
shape samples per product	—	3, reduced by per-week median
shape clips	—	$[0.05, 20]$ at build time; $[0.12, 8.0]$ at run time

	Synthetic			Real			
	$L = 0$	$L = 4$	stoch.	$L = 0$	$L = 4$	stoch.	Total
Test instances	240	240	240	200	200	200	1320

Table 7: The six test cells and their instance counts, read off the per-instance score files the benchmark ships. Every table and figure in §4 uses these six columns in this order. S averages over instances, so the two zero-lead-time cells carry a third of the weight between them and the two stochastic cells carry another third.

F Supporting measurements

F.1 Upper bound, validation set, and proposal screening

The upper-bound plan of §4.2 has a direct construction. In the paper's one-based indexing, arrivals can occur only at $A = \{t + L_t : L_t < \infty,\ t + L_t \le T\}$. A unit assigned to demand in period τ from an arrival slot $a \le \tau$ incurs $\tau - a$ periods of holding, so the latest feasible slot weakly dominates every earlier one. The unit is served exactly when $p > h(\tau - a)$; equality may be resolved either way. Orders are uncapacitated and costs are linear, hence these unit-wise choices do not compete for capacity. Their selected arrival slots are non-decreasing in τ, so the aggregate plan is feasible and creates no additional stockout. Demand with no feasible slot cannot be served by any policy. Scoring this plan under the benchmark dynamics gives the reported ex-post upper bound.

In the oracle diagnostic of §4.2, the projected base-stock rule with $\kappa = 1$ and realized future demand supplied as a point-mass forecast matches the upper bound in all four deterministic cells. The fixed rule solves (3) against this degenerate predictive. Evaluating the shipped adaptive-κ variant or a rule supplied with the true demand distribution requires separate experiments.

The validation generator behind §4.1 makes fresh-seed rebuilds of the ten documented synthetic families, plus a retail-shaped family calibrated only to the five training demands available to a policy. The composition sweep, both proposer passes and the freeze draw their instances from that generator and have no path to the benchmark's own. Candidates are inspected before they run, and one is rejected if it refers to an instance's realized demand, to the diagnostic switch that substitutes that demand for the forecast, to the benchmark's own instance files, or to the per-period lead-time sequence; each rejection is recorded. This static check targets accidental prohibited accesses; its resistance to deliberate evasion has not been evaluated.

Forecaster/search	seed 0	seed 1	seed 2	mean	sd	
core / core	0.61002	0.61009	0.60993	0.61001	0.00008	frozen OR core
core / first pass	0.61510	0.61483	0.61489	0.61494	0.00014	first pass only
core / shipped	0.61781	0.61739	0.61737	0.61752	0.00025	**shipped**
pooled / shipped	0.61839	0.61812	0.61813	0.61821	0.00015	pooled, not shipped

Table 8: Validation scores of the four configurations that decided the freeze, labelled by the forecaster and search settings. These scores support comparisons within the validation generator; §4 reports a separate test set. The second row is the loop's first pass and the third is what ships, so the two together show how the search component's gain divides between the loop's two passes. The fourth scores highest and is not shipped, because its forecaster pools across the instances of a cell and the official one-instance-at-a-time runner cannot reproduce it; §4.3 reports its test-set contribution. The standard deviation across seeds measures Monte Carlo noise from policy sampling on fixed data; uncertainty from sampling validation instances is outside this measure.

Contrast against the frozen core	n	ΔS	95% interval	moved	top-10
Shape layer, all cells	1320	$+0.0081$	$[+0.0058, +0.0105]$	0.44	0.20
Search layer, all cells	1320	$+0.0077$	$[+0.0061, +0.0093]$	0.33	0.17
Both authored layers, all cells	1320	$+0.0139$	$[+0.0115, +0.0164]$	0.62	0.14
shape, three real cells only	600	$+0.0179$	$[+0.0128, +0.0230]$	0.97	0.20
search, two stochastic cells only	440	$+0.0230$	$[+0.0185, +0.0277]$	0.99	0.17
Pooled forecaster, on top of the shipped pair	1320	$+0.0006$	$[-0.0000, +0.0013]$	0.98	0.08

Table 9: Paired bootstrap over the test instances. *moved* is the share of instances whose score changes at all, *top-10* the share of total absolute movement carried by the ten largest movers. The last row is measured against the shipped pair, the others against the frozen core. Instance-level scores come from a single seed, so a contrast here can differ slightly from the three-seed mean in Figure 4. For the pooled forecaster, the single-seed interval includes zero.

Table 8 reports the four configurations used to select the final settings of the three components. Selection scores are measured on the generator's instances, with the test set excluded from candidate scoring. Validation instances carry no product text, so the shape layer is inactive and its quality is outside this validation comparison. The harness checks this inactivity by requiring the orders emitted with and without the layer to agree entry by entry.

F.2 Control arm and paired intervals

The same-model online-adjustment control of Table 2 was run in the predecessor project. Its results are preserved in the standing-context block supplied at the start of every proposer round and retained with the experiment record. This control uses one arm and one model, with roughly twenty instances per cell. At this sample size, two of the six cells are more than two standard errors from zero: the real deterministic cell at lead time four has a negative estimate, and the real stochastic cell has a positive estimate.

The across-seed standard deviation of the shipped configuration is 0.0001 and measures rollout Monte Carlo noise on fixed data. We assess sensitivity to individual instances with the paired bootstrap. Every arm is scored on the same instances, so the contrasts are paired; we resample instances with replacement, keeping each instance's whole row of arms together, and report the percentile interval of the paired mean difference. These intervals are conditional on the fixed test set; their scope is sensitivity to its instance composition, rather than generalization across benchmarks. The measured contributions are distributed across instances: the shape layer moves nearly every real-cell instance and no synthetic one, the search layer nearly every stochastic-cell instance, and no contrast is dominated by its ten largest movers.

The composition sweep identified the four first-pass gains as variants of the same local-rate normalization in the stochastic search. The second pass added the cap; the final search component combines these two changes.

The alternative forecaster adds +0.0022 with the seasonal layer and +0.0007 with the search layer, indicating that its gain depends on the accompanying components. The negative seasonal-by-search interaction in §4.3 indicates overlapping gains from the two layers. Configurations with a pooled forecaster are excluded from submission to preserve the per-instance protocol.

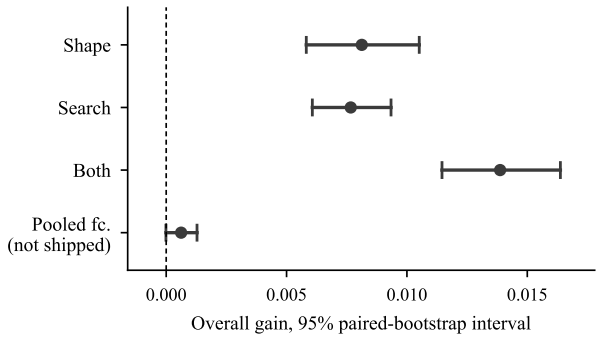


Figure 5: Paired bootstrap intervals for the overall contributions, the same quantities as Table 9. The pooled forecaster's interval includes zero; this configuration is retained as a diagnostic.

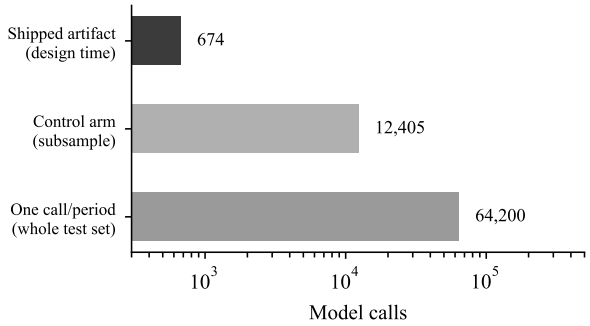


Figure 6: Model calls, log scale. The top bar counts all calls used to create the shipped artifact: shape elicitation and both proposer passes. The middle bar counts retained responses for the same-model online-adjustment control of §4.3, reported as a per-period model call on a 124-instance subsample. This experimental cost is shown separately from artifact construction. The bottom bar gives the calls required to run such a control once per period over the whole test set. The shipped policy makes no model calls during operation.

On the two cells with a deterministic lead time of four, the hindsight-tuned constant level of §4.2 exceeds every one of the ten public baselines. That section reports the strongest baseline's per-cell scores.

F.3 Design-time budget and reproducibility

The fifty-two proposals of §4.3 are all retained, and the second pass is recorded round by round: twelve of twenty-eight improved on the screening subset, four earned a full-validation re-score, one failed to run, and the best of the four was frozen. Those two passes and the shape elicitation of §4.3 are the whole design-time budget, elicitation being most of it; the products it covers appear in all three real cells, so three samples for each of the 200 products covers the benchmark's real half once. Rows of Figure 4 come from runs that take a cell's instances in one batch, the path the design-time search used; moving to the one-instance-at-a-time path a submission is scored on changes the shipped configuration by -0.0006 overall and reorders nothing.

The control arm's budget is based on 12,405 retained responses, roughly twice 124 instances times a horizon. This count indicates either coverage beyond the reported 124 instances or multiple calls in some periods. The predecessor project's logs leave these possibilities unresolved, so the budget uses the retained count. Per-cell deltas use the reported runs, while the control arm's cost remains uncertain by a factor of about two.

The reproduction of §4.3 produces the orders, scores them with the benchmark's own evaluator and replays them through the official policy protocol, one instance at a time exactly as that evaluator drives a policy. The replay reproduces the submitted orders with maximum deviation zero, so the shipped configuration carries no cross-instance dependence. Design-time replies are retained and a verification mode checks that every one a run needs is already present. This check verifies that a repeat run can use cached replies throughout, with no network calls.

The proposer context includes four test-set diagnostics: the score of a hindsight-tuned global order-up-to level; that the deterministic rule reaches the upper bound when fed the realized path; that the stochastic cells are furthest from it; and the per-cell deltas and standard errors of the control arm. The record carries no test score of any candidate, every one having been admitted or rejected on validation alone.

F.4 Numerical checks on the appendix results

We numerically check three results proved in Appendix B. For Lemma 6 we obtain the raw-space density by differencing the transformed CDF and match it to $q(\phi(d))\phi'(d)$, confirm that the uncorrected version integrates to 27.66 instead of one, exhibit an observation on which dropping the correction reverses which member wins, and check the closed form for $\mathbb{E}\Delta_t$ to four digits over two hundred periods. For Proposition 9 we run random trajectories under both accountings, including the degenerate ones in which every lead time is zero and in which every order is destroyed, and find no period where the identity fails or where the retaining accounting leaves the indicator identified. For Proposition 13 we check part (i) against random loss sequences, check the floor and the per-period bound of part (ii) against sequences built to satisfy the δ hypothesis, and then remove the hypothesis and drive the flat variant's weight below the smallest positive representable number, demonstrating the role of that hypothesis.

For the cost-structure results we solve the zero-lead-time dynamic program exactly by backward induction over a bounded integer state. The computed solutions confirm that V_t is flat on $[0, y_t^*]$ and non-increasing and that $\max\{I, y_t^*\}$ is optimal at every state under Assumption 2. The checks also cover the three falling-level cases discussed after Theorem 3 and the integer difference of Corollary 4 and its argmax over four hundred random demand laws. They also verify that Φ is non-increasing, that it crosses zero once and that the crossing is the least maximizer of its integral, at $\kappa = 0$ and at $\kappa > 0$.

The checks cover three boundary cases. On d_1 uniform on $\{2, 5\}$ and d_2 on $\{1, 4\}$ at $\rho = 1/2$ the smallest integer fractile selection is $(2, 1)$ and falls while the monotone real selection $(2, 2)$ exists, which is why Corollary 4 cannot be had by invoking the theorem with that selection; the conclusion nonetheless holds at every state there. On a shortfall supported below zero, the ordinary case of a pipeline that already covers the demand it is responsible for, $\Phi(0) < 0$, the maximizer of $\int_0^q \Phi$ over $q \geq 0$ is exactly $\{0\}$ and the crossing set of Proposition 7(iv) is empty, selecting the zero-order branch; the crossing on the whole line sits at -3 against $q^0 = 0$, which is the difference part (vi) records. Finally the pathwise identity $U^{\mathrm{naive}} = U^{\mathrm{proj}} + \Lambda$ is checked over four thousand random pipelines, with equality exactly on the paths where nothing was lost, and is then run past the end of the horizon, where in-transit units that never land leave a residual of -7 and the identity fails. That is the case $t + L \leq T$ excludes. In the paper's one-based indexing, the policy likewise places no order once $t + L > T$.