Is Escalation Worth It? A Decision-Theoretic Characterization of LLM Cascades

Anonymous Authors[1]

Abstract

Model cascades, in which a cheap LLM defers to an expensive one on low-confidence queries, are widely used to navigate the cost-quality tradeoff at deployment. Existing approaches largely treat the deferral threshold as an empirical hyperparameter, giving limited guidance on the geometry of the resulting cost-quality frontier over a model pool. We develop a decision-theoretic characterization of deterministic threshold cascades over a model pool: two-model frontiers are piecewise concave on decreasing-benefit regions, the two-model pool frontier is the pointwise envelope over pairwise cascades, and fixed k-model cascades satisfy stagewise first-order conditions that equalize marginal quality-per-cost. Across five benchmarks and eight models, full fixed chains underperform the pairwise envelope and optimized subsequence cascades add little on held-out data. A lightweight pre-generation router beats the best cascade policy on four of five datasets, suggesting that cascade performance is limited more by paying the cheap model before escalation than by a shortage of intermediate stages.

1. Introduction

Large language model (LLM) deployment is governed by a cost-quality tradeoff: the most capable models are expensive at scale, while cheaper models often miss application-specific quality targets. *Model cascades* address this tension by querying a cheap model first and escalating low-confidence responses to a more expensive model (Moslem & Kelleher, 2026). Existing methods often treat the deferral threshold as an empirical hyperparameter (Chen et al., 2023; Yue et al., 2024), leaving open the geometry of the resulting cost-quality frontier over a model pool.

We develop a decision-theoretic framework grounded in constrained optimization and duality. For two-model cascades, the budget- and quality-constrained formulations have reciprocal shadow prices, and the frontier is concave on decreasing-benefit regions of the confidence support. Given a pool of k models, the deterministic two-model threshold-cascade frontier is the pointwise envelope over $\binom{k}{2}$ pairwise frontiers. For fixed k-model threshold cascades, a single shadow price equates decision-boundary expected escalation benefit to expected downstream cost, equalizing marginal quality-per-cost across active stage boundaries.

We validate the framework on five benchmarks (MATH levels 3–5, MMLU, TriviaQA, SimpleQA, LiveCodeBench) across eight models from five providers. All model pools, valid pairs, thresholds, and routers are selected on calibration data only. Empirically, full fixed chains underperform the pairwise envelope, optimized subsequence cascades add little on held-out data, and a lightweight pre-generation router exceeds the best cascade policy on four of five datasets. Our contributions are as follows:

- **Pairwise envelope.** We characterize the two-model frontier over a pool as the envelope over $\binom{k}{2}$ single-threshold frontiers. At 90% of ceiling quality, the envelope reduces cost by up to 79.5% and matches or exceeds optimized subsequence cascades on all five benchmarks. Operationally, a deployed policy needs only the selected model pair and one threshold for a chosen budget, rather than a longer cascade chain.
- **Theory for threshold cascades.** We derive monotonicity, piecewise concavity, reciprocal shadow prices, and first-order conditions that allow expected quality to depend on prior confidence scores.
- **Cascading vs. diagnostic routing.** A lightweight pre-generation router exceeds the best cascade policy on four of five datasets, suggesting that cascade performance is limited more by the cheap model's generation cost before escalation than by a shortage of intermediate stages.

2. Related Work

Recent LLM deployment work studies both routing, which selects a model before inference, and cascading, which escalates after observing a generated response (Ong et al.,

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

2025; Chen et al., 2023; Gupta et al., 2024; Dekoninck et al., 2025). Several cascade papers formulate cost-quality tradeoffs as constrained optimization problems, but typically select thresholds by empirical search rather than characterizing the geometry of the attainable frontier. Our focus is complementary: for deterministic threshold cascades, we derive first-order conditions, concavity and shadow-price structure for two-model cascades, and the pairwise envelope over a model pool. Appendix B gives a more detailed comparison to routing, cascading, and unified routing-cascading formulations.

3. The k-Model Cascade Framework

3.1. Setup and Cascade Mechanism

Let $\mathcal{X}$ denote the space of queries and let $x \sim \mathcal{P}$ be drawn from the deployment distribution. A k-model cascade consists of an ordered model pool $\{\mathcal{M}_1, \ldots, \mathcal{M}_k\}$ together with a threshold vector $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_{k-1})$ that governs escalation between adjacent stages. When queried, model i incurs a random per-query cost $C_i(x)$ and produces a response $y_i(x)$ with quality $U_i(x) := u(y_i(x), y^*(x))$, where $y^*(x)$ is the ground-truth answer for x, available only offline. Let $c_i \equiv \mathbb{E}[C_i(x)]$ and $\mathbb{E}[U_i]$ denote the expected cost and quality, respectively. We assume the model pool is *non-dominated*: the models are ordered such that $c_1 < c_2 < \cdots < c_k$ and $\mathbb{E}[U_1] < \mathbb{E}[U_2] < \cdots < \mathbb{E}[U_k]$.[1] Each non-terminal model $i < k$ additionally produces a scalar confidence score $s_i(x)$, available at no additional model-call cost from the same call used to generate $y_i(x)$. Given $\boldsymbol{\tau}$, the cascade escalates from model i to model $i+1$ whenever $s_i(x) < \tau_i$, while the terminal model k always returns its response if stage k is reached.[2]

Stopping index, cost, and quality. Let $I(x;\boldsymbol{\tau}) := \min\big(\{j < k : s_j(x) \geq \tau_j\} \cup \{k\}\big)$ denote the stopping index on query x; including $\{k\}$ ensures that the terminal model always stops. The cascade returns $y_{I(x;\boldsymbol{\tau})}(x)$, with output quality and total cost

$$U(x;\boldsymbol{\tau}) := U_{I(x;\boldsymbol{\tau})}(x), \qquad C(x;\boldsymbol{\tau}) := \sum_{j=1}^{I(x;\boldsymbol{\tau})} C_j(x). \tag{1}$$

[1] Non-dominance is a simplifying assumption used to order the pool and reduce the number of candidate pairs. Average dominance does not preclude a model from being useful on particular conditional subpopulations; ruling this out would require a stronger conditional-dominance condition on all reachable score-prefix regions.

[2] Hereafter, we carry the query argument x explicitly in per-query random variables; all expectations and probabilities are over $x \sim \mathcal{P}$ unless otherwise noted.

Joint score distribution. Let $\mathbf{s}(x) = (s_1(x), \ldots, s_{k-1}(x))$ denote the vector of confidence scores that govern escalation decisions, and let $\mathbf{s}_{<i}(x) := (s_1(x), \ldots, s_{i-1}(x))$ denote the prefix through stage $i-1$. Under $x \sim \mathcal{P}$, $\mathbf{s}(x)$ is a random vector with joint density $f_{\mathbf{s}}$, which need not factorize due to shared dependence on query difficulty. We write $f_{s_i|\mathbf{s}_{<i}}$ for the conditional density of s_i given the prefix, and suppress the x-argument when no ambiguity results.[3]

Conditional quality. For non-terminal models $j < k$, let $m_j(\mathbf{s}_{1:j}) := \mathbb{E}[U_j(x) \mid s_1(x), \ldots, s_j(x)]$ denote the expected quality of model j conditional on the scores observed through stage j; for the terminal model, $m_k(\mathbf{s}_{<k}) := \mathbb{E}[U_k(x) \mid s_1(x), \ldots, s_{k-1}(x)]$.[4] Because confidence scores correlate with query difficulty, thresholds affect cascade quality through two channels: the probability that a query reaches each stage, and the difficulty composition of the queries reaching that stage.

Continuation values. Let $I^{>i}(x;\boldsymbol{\tau}) := \min\big(\{j : i < j < k,\ s_j(x) \geq \tau_j\} \cup \{k\}\big)$ denote the stopping index restricted to stages after i. The *quality continuation value* and *cost continuation value* at stage $i+1$ are functions of the prefix scores $\mathbf{s}_{1:i}$[5]:

$$V_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) := \mathbb{E}\big[U_{I^{>i}(x;\boldsymbol{\tau})}(x) \mid \mathbf{s}_{1:i}(x) = \mathbf{s}_{1:i}\big], \tag{2}$$

$$W_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) := \mathbb{E}\left[\sum_{\ell=i+1}^{I^{>i}(x;\boldsymbol{\tau})} C_\ell(x) \mid \mathbf{s}_{1:i}(x) = \mathbf{s}_{1:i}\right]. \tag{3}$$

The constrained optimization. The practitioner faces two dual problems:

$$\textbf{(P1)} \quad \min_{\boldsymbol{\tau}} \ \mathbb{E}[C(x;\boldsymbol{\tau})] \quad \text{s.t.} \quad \mathbb{E}[U(x;\boldsymbol{\tau})] \geq Q, \tag{4}$$

$$\textbf{(P2)} \quad \max_{\boldsymbol{\tau}} \ \mathbb{E}[U(x;\boldsymbol{\tau})] \quad \text{s.t.} \quad \mathbb{E}[C(x;\boldsymbol{\tau})] \leq B. \tag{5}$$

Both describe the same efficient tradeoff geometrically; on concave segments, standard Lagrangian duality applies with the shadow-price interpretations developed in Section 4. We focus on **(P2)** below.

[3] We assume scores are continuously distributed. Discrete signals (e.g., self-consistency vote counts) can be handled by replacing densities with mass functions and integrals with sums.

[4] We assume that all conditional expectations (e.g., continuation probabilities and conditional accuracies) are well-defined and continuous in the threshold τ. This ensures existence of optimal thresholds and justifies differentiation under the expectation.

[5] V_{i+1} and W_{i+1} are the expected quality and expected additional cost of the downstream cascade given the prefix $\mathbf{s}_{1:i}$; both are well-defined on the support of $\mathbf{s}_{1:i}$ and can be evaluated at $s_i = \tau_i$.

Pareto frontier. The *Pareto frontier* of the cascade is the value function of **(P2)** as the budget varies:

$$U^\dagger(B) := \sup_{\boldsymbol{\tau}\in[0,1]^{k-1}} \big\{\mathbb{E}[U(x;\boldsymbol{\tau})] : \mathbb{E}[C(x;\boldsymbol{\tau})] \le B\big\}. \tag{6}$$

Integral form over the confidence support. The expectations $\mathbb{E}[U(x;\boldsymbol{\tau})]$ and $\mathbb{E}[C(x;\boldsymbol{\tau})]$ admit integral representations over the confidence support that make the mechanism transparent. Define the stopping regions

$$R_i(\boldsymbol{\tau}) = \begin{cases} \{\mathbf{s} : s_1 \ge \tau_1\}, & i = 1, \\ \{\mathbf{s} : \mathbf{s}_{<i} < \boldsymbol{\tau}_{<i},\ s_i \ge \tau_i\}, & 1 < i < k, \\ \{\mathbf{s} : \mathbf{s}_{<k} < \boldsymbol{\tau}_{<k}\}, & i = k, \end{cases}$$

where vector inequalities are elementwise. Then expected quality decomposes stagewise as

$$\begin{aligned}\mathbb{E}[U(x;\boldsymbol{\tau})] = &\sum_{i=1}^{k-1} \int_{R_i(\boldsymbol{\tau})} m_i(\mathbf{s}_{1:i}) f_{\mathbf{s}_{1:i}}(\mathbf{s}_{1:i})\, d\mathbf{s}_{1:i} \\ &+ \int_{R_k(\boldsymbol{\tau})} m_k(\mathbf{s}_{<k}) f_{\mathbf{s}_{<k}}(\mathbf{s}_{<k})\, d\mathbf{s}_{<k}.\end{aligned} \tag{7}$$

and expected cost decomposes analogously as the sum of per-stage costs over reached regions:

$$\begin{aligned}\mathbb{E}[C(x;\boldsymbol{\tau})] = c_1 + \sum_{i=2}^{k} \int_{\{\mathbf{s}_{<i} < \boldsymbol{\tau}_{<i}\}} &\mathbb{E}[C_i(x) \mid \mathbf{s}_{<i}] \\ &\cdot f_{\mathbf{s}_{<i}}(\mathbf{s}_{<i})\, d\mathbf{s}_{<i}.\end{aligned} \tag{8}$$

Thus $\boldsymbol{\tau}$ affects expected quality by shifting probability mass across stopping regions and by changing the difficulty composition within those regions. The structural results below follow from how expected escalation benefit, $V_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) - m_i(\mathbf{s}_{1:i})$, varies with the boundary score s_i.

3.2. First-Order Optimality Conditions

Applying standard Lagrangian methods for constrained resource allocation to the LLM threshold-cascade problem, the following theorem gives the stationarity condition in terms of conditional expectations that are estimable from calibration data (the continuation values V_{i+1} and W_{i+1} as functions of the prefix $\mathbf{s}_{1:i}$). Appendix A.3 gives the complete KKT conditions for both constrained formulations.

Theorem 3.1 (First-Order Optimality Conditions)**.** *Consider problem equation 5 with Lagrangian*

$$\mathcal{L}(\boldsymbol{\tau}, \lambda) = \mathbb{E}[U(x;\boldsymbol{\tau})] - \lambda(\mathbb{E}[C(x;\boldsymbol{\tau})] - B),$$

so that $\lambda \ge 0$ has units of quality per unit cost. Under regularity conditions on $f_{\mathbf{s}}$ (stated precisely in Appendix A), at an interior optimum each threshold τ_i satisfies:

$$\begin{aligned}&\mathbb{E}\Big[V_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) - m_i(\mathbf{s}_{1:i}) \,\Big|\, \mathbf{s}_{<i} < \boldsymbol{\tau}_{<i},\ s_i = \tau_i\Big] \\ &\quad = \lambda\, \mathbb{E}\Big[W_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) \,\Big|\, \mathbf{s}_{<i} < \boldsymbol{\tau}_{<i},\ s_i = \tau_i\Big].\end{aligned} \tag{9}$$

Proof sketch. The threshold τ_i enters $\mathbb{E}[U(x;\boldsymbol{\tau})]$ and $\mathbb{E}[C(x;\boldsymbol{\tau})]$ only through the indicator $\mathbf{1}[s_i(x) < \tau_i]$ in the stopping regions $R_i, \dots, R_k$.[6] Let $\mathcal{S}_{<i}(\boldsymbol{\tau}) = \{\mathbf{s}_{<i} : s_1 < \tau_1, \dots, s_{i-1} < \tau_{i-1}\}$ denote the prefix region of score vectors whose associated queries reach stage i. Applying the Leibniz integral rule:

$$\begin{aligned}\frac{\partial \mathbb{E}[U(x;\boldsymbol{\tau})]}{\partial \tau_i} = &\int_{\mathcal{S}_{<i}} \big[V_{i+1}(\mathbf{s}_{<i}, \tau_i;\boldsymbol{\tau}) - m_i(\mathbf{s}_{<i}, \tau_i)\big] \\ &\cdot f_{\mathbf{s}_{1:i}}(\mathbf{s}_{<i}, \tau_i)\, d\mathbf{s}_{<i},\end{aligned} \tag{10}$$

$$\begin{aligned}\frac{\partial \mathbb{E}[C(x;\boldsymbol{\tau})]}{\partial \tau_i} = &\int_{\mathcal{S}_{<i}} W_{i+1}(\mathbf{s}_{<i}, \tau_i;\boldsymbol{\tau}) \\ &\cdot f_{\mathbf{s}_{1:i}}(\mathbf{s}_{<i}, \tau_i)\, d\mathbf{s}_{<i}.\end{aligned} \tag{11}$$

where $f_{\mathbf{s}_{1:i}}$ is the marginal density of $(s_1(x), \dots, s_i(x))$.[7] Setting $\partial\mathcal{L}/\partial\tau_i = 0$ and dividing both sides by $\Pr(\mathbf{s}_{<i}(x) < \boldsymbol{\tau}_{<i}) \cdot f_{s_i|\mathbf{s}_{<i}<\boldsymbol{\tau}_{<i}}(\tau_i)$, which appears as a common factor, yields equation 9. □

Economic interpretation. The FOC equation 9 admits a clean economic reading: *at the optimum, the ratio of decision-boundary expected escalation benefit to decision-boundary expected downstream cost is equalized across all stages and equal to the shadow price λ.* Equivalently, at each decision boundary $s_i = \tau_i$, marginal quality-per-cost $(V_{i+1} - m_i)/W_{i+1} = \lambda$ is the same regardless of stage. This is the standard optimality condition for constrained resource allocation applied to LLM threshold cascades.

3.3. Monotonicity

The FOC characterizes interior optima but does not by itself imply that the cost-quality curve is monotone. The following *decision-boundary dominance* condition is a local sufficient condition: at each active stage, marginally escalated queries must benefit in expectation from continuation.

Proposition 3.2 (Stagewise Monotonicity)**.** *Suppose that, at a threshold vector $\boldsymbol{\tau}$,*

$$\mathbb{E}\big[V_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) - m_i(\mathbf{s}_{1:i}) \,\big|\, \mathbf{s}_{<i} < \boldsymbol{\tau}_{<i},\ s_i = \tau_i\big] > 0, \tag{12}$$

for every $i = 1, \dots, k-1$. Then increasing any single threshold τ_i locally, with the others fixed, strictly increases both $\mathbb{E}[C(x;\boldsymbol{\tau})]$ and $\mathbb{E}[U(x;\boldsymbol{\tau})]$. Hence, on any connected region of threshold space where equation 12 holds for all

[6] The conditioning event $\{s_i(x) = \tau_i\}$ has measure zero; conditional expectations at the boundary are defined using the regular conditional distribution induced by the joint density of $(\mathbf{s}_{<i}, s_i)$, equivalently by disintegration via the conditional density of s_i given the prefix-reaching event $\mathbf{s}_{<i} < \boldsymbol{\tau}_{<i}$. The required densities are assumed continuous and positive at the boundary under the regularity conditions of Appendix A.

[7] The optimal policy may lie at the boundary (i.e., always or never escalate). In such cases, the first-order condition is replaced by a one-sided condition. For clarity, we focus on interior optima.

stages, any local coordinate increase in an active threshold increases both expected quality and expected cost.

Proof sketch. Condition equation 12 is the quality-gain term in the FOC equation 9: at the stage-i boundary, continuing the cascade has strictly higher expected quality than stopping at model i. By the gradient expressions equation 10–equation 11, both expected quality and expected cost therefore increase locally with τ_i. □

4. Specialization to Two-Model Cascades

The two-model cascade is the basic building block of the pairwise envelope and admits a sharper characterization than the general case. When $k = 2$, write $\mathcal{M}_L := \mathcal{M}_1$, $\mathcal{M}_H := \mathcal{M}_2$, $s_L := s_1$, $c_L := c_1$, $c_H := c_2$, and $\tau := \tau_1$. Escalating past the cheap model deterministically invokes the expensive model, so the continuation values simplify to $V_2(s;\tau) = \mathbb{E}[U_H(x) \mid s_L(x) = s]$ and $W_2(s;\tau) = \mathbb{E}[C_H(x) \mid s_L(x) = s]$. Define:

$$\begin{aligned} m_L(s) &:= \mathbb{E}[U_L(x) \mid s_L(x) = s], \\ m_H(s) &:= \mathbb{E}[U_H(x) \mid s_L(x) = s]. \end{aligned} \tag{13}$$

The stagewise integral form equation 7–equation 8 reduces to a pair of one-dimensional integrals over the confidence support $\mathcal{T} = \{\tau \in (0,1) : f_{s_L}(\tau) > 0\}$:

$$\mathbb{E}[C(x;\tau)] = c_L + \int_0^\tau \mathbb{E}[C_H(x) \mid s_L(x) = s] f_{s_L}(s)\, ds, \tag{14}$$

$$\mathbb{E}[U(x;\tau)] = \int_0^\tau m_H(s) f_{s_L}(s)\, ds + \int_\tau^1 m_L(s) f_{s_L}(s)\, ds. \tag{15}$$

If expected escalation cost is score-independent, $\mathbb{E}[C_H(x) \mid s_L(x) = s] = c_H$ for all $s \in \mathcal{T}$, specializing Theorem 3.1 yields the two-model first-order condition

$$m_H(\tau) - m_L(\tau) = \lambda\, c_H, \tag{16}$$

and the stagewise dominance condition equation 12 reduces to the standard expected-dominance condition $m_H(\tau) - m_L(\tau) > 0$ for $\tau \in \mathcal{T}^\circ$.

Definition 4.1 (Decreasing-Benefit Region)**.** An interval $I \subseteq \mathcal{T}^\circ$ is a *decreasing-benefit region* if the escalation benefit is weakly decreasing on I:

$$\begin{aligned} m_H(s') - m_L(s') \le m_H(s'') - m_L(s''), \\ \text{for all } s', s'' \in I \text{ with } s' > s''. \end{aligned} \tag{17}$$

This condition is an ordinal informativeness requirement on s_L: lower confidence scores must identify queries for which escalation has weakly larger expected benefit. The support $\mathcal{T}^\circ$ need not be a single decreasing-benefit region.[8]

Proposition 4.2 (Piecewise Concavity and Reciprocal Shadow Prices)**.** *Suppose expected escalation cost is score-independent, $\mathbb{E}[C_H(x) \mid s_L(x) = s] = c_H$ for all $s \in I$, and let $I \subseteq \mathcal{T}^\circ$ be a decreasing-benefit region. Then the Pareto frontier $U^\dagger$ is concave on the cost interval $\{\mathbb{E}[C(x;\tau)] : \tau \in I\}$. If the optimum of **(P1)** or **(P2)** for a given target lies in the interior of I, it is characterized by equation 16, and the optimal multipliers at $\tau^* \in I^\circ$ are reciprocals:*

$$\begin{aligned} \lambda^*_{P1} &= \frac{c_H}{m_H(\tau^*) - m_L(\tau^*)}, \\ \lambda^*_{P2} &= \frac{m_H(\tau^*) - m_L(\tau^*)}{c_H} = \frac{1}{\lambda^*_{P1}}. \end{aligned} \tag{18}$$

*The P1 multiplier λ^*_{P1} has units of cost per unit quality and equals the local marginal cost of tightening the quality constraint by one unit. The P2 multiplier λ^*_{P2} has units of quality per unit cost and equals the local marginal quality obtainable per additional unit of budget. If $\mathcal{T}^\circ$ is itself a single decreasing-benefit region, $U^\dagger$ is globally concave, the Lagrangian relaxations of **(P1)** and **(P2)** have zero duality gap, and the local marginals are global shadow prices.*

Proof sketch. Under score-independent expected escalation cost on I, $d\mathbb{E}[C(x;\tau)]/d\tau = c_H f_{s_L}(\tau)$ and $d\mathbb{E}[U(x;\tau)]/d\tau = (m_H(\tau) - m_L(\tau)) f_{s_L}(\tau)$. Thus, at any interior $\tau \in I$ with $f_{s_L}(\tau) > 0$, the frontier slope is $dU^\dagger/dC = (m_H(\tau) - m_L(\tau))/c_H$. On a decreasing-benefit region this slope is non-increasing in τ; since cost is strictly increasing in τ, the slope is non-increasing in cost, so $U^\dagger$ is concave on the corresponding cost interval. The reciprocal relationship follows from the envelope theorem. Full proof in Appendix A.4. □

The score-independence assumption isolates the economic geometry: thresholds change the probability and composition of escalation without systematically changing the expected downstream cost conditional. Appendices A.2 and C.7 discuss and empirically check this condition.

5. The Pairwise Envelope Over a Model Pool

Given a pool of k models, a practitioner can deploy any two-model cascade formed from a pair (i, j) with $i < j$. This section characterizes the frontier achievable over all such two-model cascades and identifies the budget levels at which the optimal pair transitions. Let $\mathcal{Q} = \{(i,j) : 1 \le i < j \le$

[8] Section 6 evaluates the condition empirically on representative envelope pairs. Outside any decreasing-benefit region the frontier is locally non-concave; see Appendix A.5 for a discussion of randomized threshold policies in this regime.

$k\}$ denote the set of $\binom{k}{2}$ cost-ordered pairs, and for each $(i,j) \in \mathcal{Q}$ let $U^{\dagger,(i,j)}$ denote the Pareto frontier equation 6 of the two-model cascade with $\mathcal{M}_L = \mathcal{M}_i$ and $\mathcal{M}_H = \mathcal{M}_j$. By non-dominance of the pool, each pair satisfies $c_i < c_j$ and $\mathbb{E}[U_j] > \mathbb{E}[U_i]$, so every pair defines a non-trivial two-model cascade. The *pairwise envelope* is the pointwise supremum of per-pair Pareto frontiers:

$$U^*(B) := \sup_{\substack{(i,j)\in\mathcal{Q}:\\ B\in[c_i,\, c_i+c_j]}} U^{\dagger,(i,j)}(B), \qquad B \in [c_1, c_k]. \tag{19}$$

Each pairwise frontier $U^{\dagger,(i,j)}$ has domain $[c_i,\, c_i + c_j]$. We restrict attention to budgets $B \le c_k$, i.e. budgets no higher than the most expensive standalone model.[9]

Remark 5.1 (Switching points). Since U^* is the pointwise supremum of the piecewise smooth per-pair frontiers $\{U^{\dagger,(i,j)} : (i,j) \in \mathcal{Q}\}$, it is smooth on each region where a single pair (i,j) attains the supremum and has a corner at the *switching points* where the optimal pair changes. Generically, at a switching point B^* the left- and right-derivatives of U^* differ, so the shadow price of quality jumps discontinuously; tangential intersections are non-generic but possible. Practically, these are the budget levels at which the optimal cascade recipe changes.

6. Experiments

6.1. Experimental Setup

We evaluate eight models from five providers: Llama 3.1-8B, Qwen2.5-7B, GPT-4o mini, GPT-oss-20B, DeepSeek-V3, Llama 3.3-70B, GPT-4o, and MiniMax-M2.7. Benchmarks cover MATH levels 3–5 (Hendrycks et al., 2021b), MMLU (Hendrycks et al., 2021a), TriviaQA (Joshi et al., 2017), SimpleQA (Wei et al., 2024), and LiveCodeBench (Jain et al., 2024); grading details are in Appendix C.2. Single-model operating points are reported in Appendix C.3: Llama 3.1-8B is the lowest-cost model throughout, while the highest-accuracy model varies by dataset. Descriptive pairwise frontiers are in Appendix C.4. Confidence scores are white-box log-probability signals (Bouchard et al., 2026); main-text cascades use mean token negentropy, with alternative signals and a learned scorer compared in Appendix C.1.

We compare the pairwise envelope against the highest-accuracy single-model baseline, a full fixed cost-ordered cascade chain, a FrugalGPT-style cascade that selects a cost-ordered model sequence and thresholds (Chen et al., 2023), and a diagnostic frozen-embedding router, a lightweight version of the learned routing approach in RouteLLM (Ong et al., 2025) (see Appendix C.11). The envelope is built from deterministic two-model uncertainty-threshold cascades, closely related to token-level uncertainty cascades (Gupta et al., 2024); candidates are restricted to calibration-admissible model pairs and evaluated under the same cost-quality criterion.[10] Per-query cost uses actual token counts and the token prices in Appendix C.7, which also checks the cost–score independence approximation. We report test-set median curves with pointwise 10th–90th percentile bands over 50 random 50/50 calibration-test splits.

6.2. Held-Out Frontier Comparisons

Figure 1 compares three deterministic threshold-cascade policy classes. The pairwise envelope selects the best two-model cascade at each budget. The full fixed chain uses the calibration-selected non-dominated pool in cost order and optimizes thresholds only, matching the fixed-chain object in the theory. The optimal subsequence baseline is broader: it jointly selects a cost-ordered subsequence and thresholds on calibration data. Table 1 summarizes these frontiers using two deployment-oriented metrics: normalized gain over random escalation between the cheapest and highest-accuracy models, which measures area above the no-signal baseline across budgets, and cost reduction at 90% of ceiling quality relative to always using the highest-accuracy model. On both metrics, the pairwise envelope is competitive with optimized subsequence cascades across all datasets: normalized gain differs by at most 0.014, and CR@90 is identical on MMLU, TriviaQA, and MATH, within one point on SimpleQA, and higher for the envelope on LiveCodeBench. By contrast, the full fixed chain has lower normalized gain and lower CR@90 in every dataset. Thus, within the deterministic threshold-cascade class studied here, additional mandatory stages can hurt, while optional intermediate stages add little beyond the best pairwise policy. The practical implication is that, for the evaluated model pools and tasks, a practitioner can sweep one threshold per calibration-valid pair, take the envelope, and deploy only the selected pair and threshold for a chosen budget, obtaining performance comparable to joint subsequence optimization while avoiding a higher-dimensional search. Appendices C.5, C.8, C.9, and C.10 verify the two-model sweep against k=2 NSGA-II and show stability across calibration size, grid resolution, and optimizer choice.

[9] The pointwise supremum of locally concave pairwise frontiers need not itself be globally concave; switches between pairs can introduce non-concavities. If randomized mixtures over cascade policies are allowed, the achievable set is convexified and the relevant frontier becomes the concave envelope of these deterministic frontiers.

[10] Admissible edges, model pools, model sequences, thresholds, and router classifiers are selected on calibration data only, before held-out evaluation. An escalation edge is included only when the downstream model is more costly and more accurate on the calibration split. The router is a diagnostic baseline, not a state-of-the-art learned-routing claim. Monetary cost is not incurred by the open-source sentence-transformer embedding used by the router; the reported costs count LLM token costs.

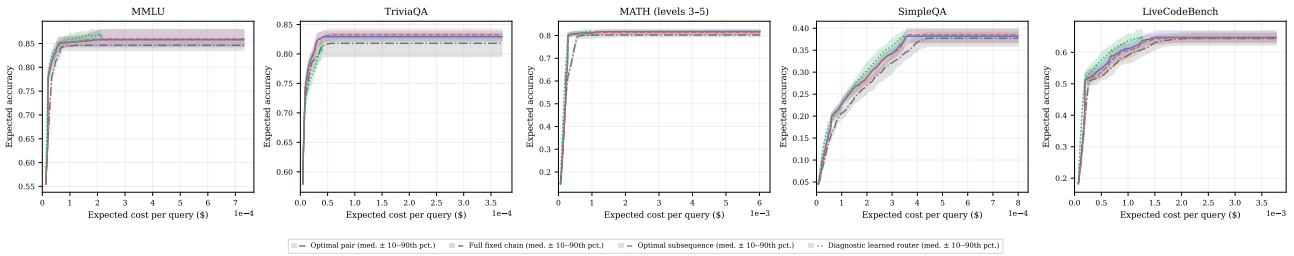

Figure 1. Optimal pair, full fixed chain, optimal subsequence, and diagnostic router across five datasets. Curves are medians across 50 random 50/50 splits; shaded bands are 10th–90th percentiles. Thresholds/sequences fit on calibration data and evaluated held out.

Table 1. Method comparison across five datasets. Gain = normalized area above the random-escalation baseline connecting the lowest-cost and highest-accuracy single-model endpoints, computed on the median held-out frontier. CR % = cost reduction at 90% of ceiling quality vs. the highest-accuracy single model. Computed over 50 random 50/50 calibration-test splits.

	MMLU		TriviaQA		MATH (levels 3–5)		SimpleQA		LiveCodeBench	
Method	Gain	CR %	Gain	CR %	Gain	CR %	Gain	CR %	Gain	CR %
Always-expensive	—	0.0	—	0.0	—	0.0	—	0.0	—	0.0
Optimal pair	0.360	**73.7**	**0.316**	**74.5**	0.393	79.5	0.158	15.1	0.329	56.1
Full fixed chain	0.259	59.3	0.249	67.2	0.344	66.7	0.093	1.3	0.277	41.4
Optimal subsequence	0.360	**73.7**	0.306	**74.5**	0.390	79.5	0.156	14.2	0.315	51.2
Diagnostic learned router	**0.393**	**73.7**	0.219	59.9	**0.394**	**81.2**	**0.193**	**26.7**	**0.365**	**64.9**

Diagnostic learned router comparison. The embedding router has higher normalized gain than the cascade envelope on 4/5 datasets, with the largest gains on SimpleQA and LiveCodeBench (Table 1). A same-signal comparison shows that this advantage is primarily structural: pre-generation dispatch avoids paying the cheap model's cost c_L on queries sent elsewhere, while an embedding cascade using the same signal remains below the UQ cascade on 4/5 datasets. TriviaQA is the exception, where query embeddings are near-uninformative (AUROC ≈ 0.49). This diagnostic comparison is intended to separate structural costs from signal quality rather than to benchmark the strongest possible learned router. It should therefore be read as evidence about the structural cost of post-generation escalation, not as a claim that this simple router is optimal among learned routing policies.

Structural condition diagnostics. Appendix C.6 reports escalation-benefit diagnostics for the conditions in Section 4. Representative envelope pairs satisfy expected dominance on most of the score support and satisfy the decreasing-benefit condition on 90–100% of the support. The main deviations occur in high-confidence regions, where the cheap model is already reliable and escalation can add little or negative expected benefit.

7. Conclusion

We developed a decision-theoretic framework for LLM cascades that characterizes cost-quality tradeoffs via piecewise concavity, reciprocal shadow prices, and first-order conditions equalizing marginal quality-per-cost across stage boundaries. For a pool of k models, we characterize the frontier achievable by deterministic two-model threshold cascades as the pointwise envelope over $\binom{k}{2}$ pairwise frontiers. Empirically, within the deterministic threshold-cascade class studied here, optimized subsequence cascades do not deliver practically meaningful held-out gains over the pairwise envelope, while full fixed chains underperform it across five benchmarks, eight models, and five providers. A diagnostic learned k-model router that dispatches pre-generation, however, exceeds the best cascade policy on four of five datasets, including three where its embedding signal is weaker than the cascade's white-box uncertainty signal, because it avoids paying the cheap model's generation cost on queries routed elsewhere. The exception is TriviaQA, where query embeddings are uninformative and post-generation confidence remains the only viable signal. Together, these results suggest a narrower practical diagnostic: when inexpensive pre-generation features contain usable difficulty information, even simple routing can expose the structural cost paid by post-generation cascades; when such features are uninformative, confidence-based cascading remains competitive. This structural-cost conclusion is not an exhaustive claim about all learned routing systems, including richer routers or route-then-cascade hybrids. Extending the theoretical framework to jointly characterize routing and cascading under a common cost-quality formulation is a natural next step. The empirical scope is also limited to the evaluated model pool, short-form and code correctness benchmarks, and monetary token-cost objectives; dedicated reasoning models, long-form generation, and latency-aware objectives may change the relative value of intermediate stages.

References

Aggarwal, P., Madaan, A., Anand, A., Potharaju, S. P., Mishra, S., Zhou, P., Gupta, A., Rajagopal, D., Kappaganthu, K., Yang, Y., Upadhyay, S., Faruqui, M., and Mausam. Automix: Automatically mixing language models, 2025. URL https://arxiv.org/abs/2310.12963.

Bouchard, D. and Chauhan, M. S. Uncertainty quantification for language models: A suite of black-box, white-box, LLM judge, and ensemble scorers. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=WOFspd4lq5.

Bouchard, D., Chauhan, M. S., Skarbrevik, D., Ra, H.-K., Bajaj, V., and Ahmad, Z. Uqlm: A python package for uncertainty quantification in large language models. *Journal of Machine Learning Research*, 27(13):1–10, 2026. URL http://jmlr.org/papers/v27/25-1557.html.

Chen, L., Zaharia, M., and Zou, J. Frugalgpt: How to use large language models while reducing cost and improving performance, 2023. URL https://arxiv.org/abs/2305.05176.

Dekoninck, J., Baader, M., and Vechev, M. A unified approach to routing and cascading for llms, 2025. URL https://arxiv.org/abs/2410.10347.

Ding, D., Mallick, A., Zhang, S., Wang, C., Madrigal, D., Garcia, M. D. C. H., Xia, M., Lakshmanan, L. V. S., Wu, Q., and Rühle, V. Best-route: Adaptive llm routing with test-time optimal compute, 2025. URL https://arxiv.org/abs/2506.22716.

Farr, D., Manzonelli, N., Cruickshank, I., and West, J. Redct: A systems design methodology for using llm-labeled data to train and deploy edge classifiers for computational social science, 2024. URL https://arxiv.org/abs/2408.08217.

Feng, T., Shen, Y., and You, J. Graphrouter: A graph-based router for llm selections, 2025. URL https://arxiv.org/abs/2410.03834.

Gupta, N., Narasimhan, H., Jitkrittum, W., Rawat, A. S., Menon, A. K., and Kumar, S. Language model cascades: Token-level uncertainty and beyond, 2024. URL https://arxiv.org/abs/2404.10136.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding, 2021a. URL https://arxiv.org/abs/2009.03300.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., and Steinhardt, J. Measuring mathematical problem solving with the math dataset, 2021b. URL https://arxiv.org/abs/2103.03874.

Jain, N., Han, K., Gu, A., Li, W.-D., Yan, F., Zhang, T., Wang, S., Solar-Lezama, A., Sen, K., and Stoica, I. Livecodebench: Holistic and contamination free evaluation of large language models for code, 2024. URL https://arxiv.org/abs/2403.07974.

Jitkrittum, W., Gupta, N., Menon, A. K., Narasimhan, H., Rawat, A. S., and Kumar, S. When does confidence-based cascade deferral suffice?, 2024. URL https://arxiv.org/abs/2307.02764.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension, 2017. URL https://arxiv.org/abs/1705.03551.

Malinin, A. and Gales, M. Uncertainty estimation in autoregressive structured prediction, 2021. URL https://arxiv.org/abs/2002.07650.

Manakul, P., Liusie, A., and Gales, M. J. F. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models, 2023. URL https://arxiv.org/abs/2303.08896.

Markovic-Voronov, J., Behdin, K., Xu, Y., Zhou, Z., Wang, Z., and Mazumder, R. Robust batch-level query routing for large language models under cost and capacity constraints, 2026. URL https://arxiv.org/abs/2603.26796.

Mei, K., Xu, W., Guo, M., Lin, S., and Zhang, Y. Omnirouter: Budget and performance controllable multi-llm routing, 2025. URL https://arxiv.org/abs/2502.20576.

Moslem, Y. and Kelleher, J. D. Dynamic model routing and cascading for efficient llm inference: A survey, 2026. URL https://arxiv.org/abs/2603.04445.

Ong, I., Almahairi, A., Wu, V., Chiang, W.-L., Wu, T., Gonzalez, J. E., Kadous, M. W., and Stoica, I. Routellm: Learning to route llms with preference data, 2025. URL https://arxiv.org/abs/2406.18665.

Scalena, D., Zotos, L., Fersini, E., Nissim, M., and Üstün, A. Eager: Entropy-aware generation for adaptive inference-time scaling, 2025. URL https://arxiv.org/abs/2510.11170.

Su, J., Lin, F., Feng, Z., Zheng, H., Wang, T., Xiao, Z., Zhao, X., Liu, Z., Cheng, L., and Wang, H. Cp-router: An uncertainty-aware router between llm and lrm, 2025. URL https://arxiv.org/abs/2505.19970.

Valkanas, A., Pal, S., Rumiantsev, P., Zhang, Y., and Coates, M. C3po: Optimized large language model cascades with probabilistic cost constraints for reasoning, 2025. URL https://arxiv.org/abs/2511.07396.

Wang, C., Liu, X., Liu, Y., Zhu, Y., Mo, X., Jiang, J., and Chen, H. When to reason: Semantic router for vllm, 2025. URL https://arxiv.org/abs/2510.08731.

Wei, J., Karina, N., Chung, H. W., Jiao, Y. J., Papay, S., Glaese, A., Schulman, J., and Fedus, W. Measuring short-form factuality in large language models, 2024. URL https://arxiv.org/abs/2411.04368.

Yue, M., Zhao, J., Zhang, M., Du, L., and Yao, Z. Large language model cascades with mixture of thoughts representations for cost-efficient reasoning, 2024. URL https://arxiv.org/abs/2310.03094.

Zhang, K., Wang, C., Peng, L., Go, A., and Liu, X. Privacy-preserved llm cascade via cot-enhanced policy learning, 2025. URL https://arxiv.org/abs/2410.08014.

Zhang, X., Huang, Z., Taga, E. O., Joe-Wong, C., Oymak, S., and Chen, J. Efficient contextual llm cascades through budget-constrained policy learning, 2024. URL https://arxiv.org/abs/2404.13082.

Zhuang, R., Wu, T., Wen, Z., Li, A., Jiao, J., and Ramchandran, K. Embedllm: Learning compact representations of large language models, 2024. URL https://arxiv.org/abs/2410.02223.

A. Supplementary Theoretical Analysis

A.1. Regularity conditions for Theorem 3.1.

We assume the joint density $f_{\mathbf{s}}$ is continuous and strictly positive on the interior of its support, and that m_i, V_{i+1}, and W_{i+1} are continuous in $(s_{1:i}, \boldsymbol{\tau})$. The proof follows the sketch in Section 3.2; boundary optima require replacing stationarity with one-sided inequalities.

A.2. Score-Dependent Per-Query Cost

Proposition 4.2 does not require constant per-query costs. It requires the weaker score-independence condition $\mathbb{E}[C_H(x) \mid s_L(x) = s] = c_H$ on the relevant confidence region, which makes expected cost linear in the escalation probability. If expected token cost varies systematically with the cheap model's confidence score, then $\mathbb{E}[C_H(x) \mid s_L(x) < \tau]$ can depend on τ and the realized cost curve need not be linear in $F_{s_L}(\tau)$. In that case, decreasing escalation benefit alone no longer guarantees concavity of the realized token-cost frontier, though Theorem 3.1 and Proposition 3.2 still apply. Appendix C.7 reports the corresponding cost–score correlation diagnostic.

A.3. Complete First-Order Conditions for the Constrained Problems

Section 3.2 states the stationarity component of the FONC for the budget-constrained problem **(P2)**. For completeness, the KKT conditions for an interior optimum of **(P2)** are

$$\mathbb{E}[C(x;\boldsymbol{\tau})] \le B, \quad \text{(primal feasibility)} \tag{20}$$

$$\lambda \ge 0, \quad \text{(dual feasibility)} \tag{21}$$

$$\lambda\,(\mathbb{E}[C(x;\boldsymbol{\tau})] - B) = 0, \quad \text{(complementary slackness)} \tag{22}$$

and, for each active threshold,

$$\mathbb{E}\Big[V_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) - m_i(\mathbf{s}_{1:i}) \;\Big|\; {}^{\mathbf{s}_{<i}<\boldsymbol{\tau}_{<i}}_{s_i=\tau_i}\Big] = \lambda\,\mathbb{E}\Big[W_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) \;\Big|\; {}^{\mathbf{s}_{<i}<\boldsymbol{\tau}_{<i}}_{s_i=\tau_i}\Big]. \tag{23}$$

Boundary optima replace stationarity with the appropriate one-sided inequalities.

The quality-constrained problem **(P1)** uses the same marginal objects with the reciprocal shadow price. Write its Lagrangian as

$$\mathcal{L}_{P1}(\boldsymbol{\tau}, \mu) = \mathbb{E}[C(x;\boldsymbol{\tau})] + \mu\,(Q - \mathbb{E}[U(x;\boldsymbol{\tau})])\,,$$

where $\mu \ge 0$ has units of cost per unit quality. Its KKT conditions are

$$\mathbb{E}[U(x;\boldsymbol{\tau})] \ge Q, \quad \text{(primal feasibility)} \tag{24}$$

$$\mu \ge 0, \quad \text{(dual feasibility)} \tag{25}$$

$$\mu\,(Q - \mathbb{E}[U(x;\boldsymbol{\tau})]) = 0, \quad \text{(complementary slackness)} \tag{26}$$

and, for each active threshold,

$$\mathbb{E}\Big[W_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) \;\Big|\; {}^{\mathbf{s}_{<i}<\boldsymbol{\tau}_{<i}}_{s_i=\tau_i}\Big] = \mu\,\mathbb{E}\Big[V_{i+1}(\mathbf{s}_{1:i};\boldsymbol{\tau}) - m_i(\mathbf{s}_{1:i}) \;\Big|\; {}^{\mathbf{s}_{<i}<\boldsymbol{\tau}_{<i}}_{s_i=\tau_i}\Big]. \tag{27}$$

Thus the marginal cost-per-quality ratio is equalized across active stage boundaries. Whenever the same interior frontier point solves both constrained formulations, comparison with equation 9 gives $\mu = 1/\lambda$. In the two-model score-independent case, equation 27 reduces to

$$c_H = \mu\,(m_H(\tau) - m_L(\tau)), \tag{28}$$

which is the P1 counterpart to equation 16.

A.4. Proof of Proposition 4.2 (Piecewise Concavity and Reciprocal Shadow Prices)

Proof. Fix a decreasing-benefit region I. Under score-independent expected escalation cost on I, differentiating the two-model integral expressions gives

$$\frac{d}{d\tau}\mathbb{E}[U(x;\tau)] = \big(m_H(\tau) - m_L(\tau)\big) f_{s_L}(\tau),$$
$$\frac{d}{d\tau}\mathbb{E}[C(x;\tau)] = c_H f_{s_L}(\tau).$$

The slope of the cost-quality frontier at an interior point parameterized by τ is therefore

$$\frac{d\mathbb{E}[U(x;\tau)]}{d\mathbb{E}[C(x;\tau)]} = \frac{m_H(\tau) - m_L(\tau)}{c_H}.$$

On a decreasing-benefit region this slope is non-increasing in τ and, because expected cost is increasing in τ, non-increasing in cost. Hence $U^\dagger$ restricted to the corresponding cost interval is concave.

For a target whose unrestricted optimum τ^* lies in $\mathrm{int}(I)$, stationarity of the Lagrangian on I is necessary and sufficient. The envelope theorem applied to the value functions $V_{P1}(Q) = \min_\tau \mathbb{E}[C(x;\tau)]$ s.t. $\mathbb{E}[U(x;\tau)] \ge Q$ and $V_{P2}(B) = U^\dagger(B)$ yields $\lambda^*_{P1} = V'_{P1}(Q)$ and $\lambda^*_{P2} = U^{\dagger\prime}(B)$. Since $V'_{P1}(Q) = c_H/(m_H(\tau^*) - m_L(\tau^*))$ and $U^{\dagger\prime}(B) = (m_H(\tau^*) - m_L(\tau^*))/c_H$, these are reciprocals.

When $I = \mathcal{T}^\circ$, the value function $U^\dagger$ is globally concave. For any non-degenerate budget $B \in (c_L, c_L + c_H)$, Slater's condition for (P2) follows from strict budget feasibility: integrating $d\mathbb{E}[C(x;\tau)]/d\tau = c_H f_{s_L}(\tau)$ gives $\mathbb{E}[C(x;\tau_0)] = c_L + c_H F_{s_L}(\tau_0)$, so any τ_0 with escalation probability $p_0 = F_{s_L}(\tau_0) < (B - c_L)/c_H$ satisfies $\mathbb{E}[C(x;\tau_0)] < B$, which exists by continuity of F_{s_L}. Standard Lagrangian duality for the resulting concave program then gives zero duality gap, and the local marginals are global shadow prices at interior optima. Boundary budgets

$B \le c_L$ and $B \ge c_L + c_H$ reduce to one-sided derivative cases.

If the optimum lies at the boundary of I, the stationarity condition is replaced by a one-sided inequality and the reciprocal relationship holds as an inequality between the left- and right-derivatives of the value functions. □

A.5. Randomized Threshold Policies Outside Decreasing-Benefit Regions

Outside any decreasing-benefit region, the map $\tau \mapsto m_H(\tau) - m_L(\tau)$ is locally increasing, so the slope of the Pareto frontier $U^\dagger$ is locally increasing in cost. In this regime, the frontier is locally non-concave and a randomized mixture of two thresholds (i.e., deploying threshold τ_a with probability α and τ_b with probability $1-\alpha$) can achieve a cost-quality pair strictly above the deterministic frontier. Specifically, for any $\tau_a < \tau_b$ in a locally increasing region, the convex combination $(\alpha \cdot \mathbb{E}[C(x;\tau_a)] + (1-\alpha) \cdot \mathbb{E}[C(x;\tau_b)],\ \alpha \cdot \mathbb{E}[U(x;\tau_a)] + (1-\alpha) \cdot \mathbb{E}[U(x;\tau_b)])$ lies above the deterministic curve whenever the curve is locally convex.

However, local non-concavity does not imply that deterministic cascades are empirically uncompetitive, but rather that randomizing between two thresholds can convexify a locally convex segment of the deterministic frontier. Whether the deterministic cascade lies above the single-model endpoint chord depends on the average escalation benefit among escalated queries relative to non-escalated queries. In practice, the magnitude of local non-concavities (reversals in $m_H - m_L$) is small relative to the overall frontier curvature on the datasets studied in Section 6, so deterministic threshold cascades remain near the relevant empirical frontier.

B. Extended Related Work

LLM routing and cascading. A large body of recent work studies cost-efficient LLM deployment via model selection. Routing methods select a single model per query prior to inference, typically using learned routers trained on query representations or preference data (Ong et al., 2025; Zhuang et al., 2024; Feng et al., 2025; Wang et al., 2025). Several works incorporate explicit constraints such as budget, latency, or capacity into routing decisions (Mei et al., 2025; Markovic-Voronov et al., 2026; Ding et al., 2025). Cascading methods instead query models sequentially and decide whether to accept or escalate a response based on signals observed *after* generation, including learned confidence estimators, token-level probabilities, and self-consistency signals (Chen et al., 2023; Yue et al., 2024; Aggarwal et al., 2025; Jitkrittum et al., 2024; Gupta et al., 2024). Extensions consider reasoning-intensive settings (Valkanas et al., 2025), uncertainty-aware routing with statistical guarantees (Su et al., 2025), deployment constraints such as privacy (Zhang et al., 2025), and unified routing-cascading formulations (Dekoninck et al., 2025). We focus on the cascading setting.

Analytical characterization of the cost-quality tradeoff. Several prior works formulate cascade design as constrained optimization (Zhang et al., 2024; Gupta et al., 2024; Chen et al., 2023; Jitkrittum et al., 2024), but do not characterize concavity conditions, dual shadow-price structure, or the geometry of threshold-cascade frontiers over a model pool. Chen et al. (2023) minimize expected cost subject to an accuracy constraint, selecting thresholds by empirical search rather than deriving first-order necessary conditions for optimality. Valkanas et al. (2025) introduces probabilistic cost constraints and provides generalization guarantees, but does not characterize the structure of the achievable cost-quality tradeoff. Dekoninck et al. (2025) derives optimal routing, cascading, and cascade-routing strategies from query- and output-dependent quality estimates. We address these gaps by modeling expected quality conditional on stage-wise confidence scores, and deriving piecewise concavity on decreasing-benefit regions, reciprocal shadow-price relationships, and the pairwise envelope as the frontier achievable by two-model threshold cascades. We also study whether k-model cascades ($k > 2$) improve on the best two-model cascade from the same pool, deriving stagewise first-order necessary conditions for multi-stage cascades and providing held-out evidence that optimized multi-stage threshold cascades do not materially improve on the pairwise envelope in our evaluated settings.

C. Additional Empirical Analyses

C.1. Scorer Choice Ablation

Figure 2 compares alternative UQ signals as cascade deferral scores. Within each calibration-test split, we first select the admissible model pairs using calibration data only. We then recompute the two-model threshold frontier for each scorer and each calibration-valid pair, summarizing performance by pair-level normalized gain over a no-signal random-escalation baseline. The plotted point is the median gain across pair cells, and the vertical bar gives the 10th–90th percentile range across those cells. The averages include calibration-valid pairs that never attain the envelope, so low pair-average gains need not reflect the performance of the selected envelope.

Scorers. We compute the UQ signals using the UQLM library (Bouchard et al., 2026; Bouchard & Chauhan, 2025). Six confidence signals are compared: five off-the-shelf log-probability scorers (mean token negentropy, min token negentropy, probability margin, min probability, sequence probability) and a learned `logreg_ensemble` that fits a

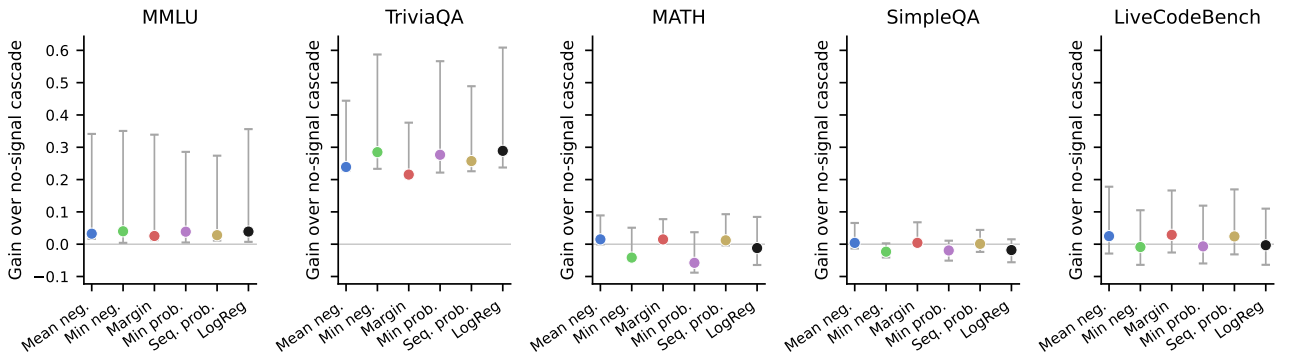

Figure 2. Scorer choice ablation. Points show median pair-level normalized gain over a no-signal random-escalation baseline; vertical bars show the 10th–90th percentile across calibration-valid pair cells, including pairs that need not attain the pairwise envelope. Values are computed from 50 calibration-test splits.

logistic regression on calibration data using all five base scorers as features, predicting cheap-model correctness $U_L(x)$. For a generated response $y = (t_1, \ldots, t_L)$, let p_j denote the probability assigned to the generated token t_j.

Length-Normalized Sequence Probability (LNSP). This score uses the geometric mean of the probabilities assigned to the generated tokens, which removes the mechanical penalty faced by longer responses (Malinin & Gales, 2021):

$$\mathrm{LNSP}(y) = \left(\prod_{j=1}^{L} p_j \right)^{1/L}.$$

Minimum Token Probability (MTP). This score records the least confident generated token in the response (Manakul et al., 2023):

$$\mathrm{MTP}(y) = \min_{j \in \{1,\ldots,L\}} p_j.$$

The remaining scorers use the top-K alternatives at each output position. Let $\{p_{j,1}, \ldots, p_{j,K}\}$ be the top-K probabilities at position j, sorted from largest to smallest, and define the renormalized top-K probabilities $p_{j,k}^{(K)} = p_{j,k} / \sum_{\ell=1}^{K} p_{j,\ell}$ for $k = 1, \ldots, K$. In all experiments, we request $K = 15$ top log-probabilities when available; for the Llama models served through the Together AI API, which exposes only $K = 5$ in our setup, the top-K scorers use $K = 5$.

Probability Margin (PM). This margin averages the probability gap between the most likely and second-most-likely token at each position (Farr et al., 2024):

$$\mathrm{PM}(y) = \frac{1}{L} \sum_{j=1}^{L} (p_{j,1}^{(K)} - p_{j,2}^{(K)}).$$

Average Token Negentropy (ATN@K). This score averages a normalized negentropy transformation over token positions (Scalena et al., 2025; Manakul et al., 2023). First define the top-K entropy at position j as

$$\mathrm{TE@}K(t_j) = -\sum_{k=1}^{K} p_{j,k}^{(K)} \log p_{j,k}^{(K)}.$$

We convert entropy to a confidence-oriented score in $[0, 1]$ by

$$\mathrm{TN@}K(t_j) = 1 - \frac{\mathrm{TE@}K(t_j)}{\log K},$$

$$\mathrm{ATN@}K(y) = \frac{1}{L} \sum_{j=1}^{L} \mathrm{TN@}K(t_j).$$

Minimum Token Negentropy (MTN@K). This score takes the least confident normalized-negentropy value across the generated positions (Scalena et al., 2025; Manakul et al., 2023):

$$\mathrm{MTN@}K(y) = \min_{j \in \{1,\ldots,L\}} \mathrm{TN@}K(t_j).$$

Gain metric. For a pair (L, H), the scorer-choice ablation uses a no-signal random-escalation baseline within the same cascade architecture: with escalation probability p, every query pays the cheap-model cost and a random fraction p also pays the expensive-model cost, giving $U_{\text{rand}}(p) = (1-p)a_L + pa_H$ and $C_{\text{rand}}(p) = c_L + pc_H$.

Benefit-AUROC diagnostic. As a secondary diagnostic, Table 2 reports benefit-AUROC in parentheses. For each pair, benefit-AUROC is the AUROC of $-s_L(x)$ for predicting positive realized escalation benefit, $\mathbf{1}\{U_H(x) > U_L(x)\}$; thus larger values mean low confidence better identifies queries on which escalation changes the answer from incorrect to correct.

Findings. Mean token negentropy is the most stable default: it achieves the highest average gain on MMLU, MATH, and LiveCodeBench, and is close to the best scorer on SimpleQA. TriviaQA is the exception, where the learned ensemble and min-token negentropy perform best. The benefit-AUROC correlations are positive in the pooled data on all five datasets and remain positive after pair demeaning on four of five datasets, but the ranking is not perfect. For example, the learned ensemble often has high benefit-AUROC without the highest gain. This reflects the fact that cascade value depends not only on ranking positive escalation-benefit cases, but also on where thresholds place mass along the score support and on the cost distribution of escalated queries. We therefore use mean token negentropy as the common main-text scorer because it is consistently competitive without dataset-specific scorer selection.

Table 2. Median pair-level gain (benefit-AUROC in parentheses) per scorer × dataset, averaged across valid pairs. Gain < 0 indicates that the cascade underperforms the no-signal random-escalation baseline on average. Bold marks the highest-gain scorer per dataset. The bottom rows report Spearman correlations between benefit-AUROC and gain before and after subtracting pair-specific means across scorers.

Scorer	MMLU	TriviaQA	MATH	SimpleQA	LiveCodeBench
Mean negentropy	**0.146** (0.68)	0.303 (0.76)	**0.035** (0.55)	0.019 (0.49)	**0.058** (0.56)
Min negentropy	0.139 (0.68)	0.368 (0.78)	−0.013 (0.60)	−0.021 (0.45)	0.011 (0.55)
Prob. margin	0.141 (0.67)	0.266 (0.73)	0.031 (0.55)	**0.023** (0.50)	0.056 (0.55)
Min probability	0.112 (0.66)	0.355 (0.77)	−0.036 (0.58)	−0.020 (0.45)	0.018 (0.55)
Seq. probability	0.111 (0.67)	0.324 (0.76)	0.034 (0.55)	0.007 (0.48)	0.054 (0.55)
LogReg ensemble	0.144 (0.68)	**0.378** (0.79)	0.003 (0.59)	−0.020 (0.45)	0.014 (0.55)
Pooled r_s	0.83	0.42	0.56	0.81	0.83
Pair-demeaned r_s	0.48	0.90	−0.09	0.94	0.40

C.2. Dataset Grading Details

We use dataset-specific binary correctness labels. MMLU responses are graded by exact match to the selected multiple-choice letter. TriviaQA uses normalized exact match after lowercasing and removing punctuation, articles, and excess whitespace. MATH is graded with a symbolic-equivalence checker based on SymPy after extracting the final answer. SimpleQA is graded with the official short-answer factuality rubric using an LLM judge, mapped to binary correctness. LiveCodeBench is graded by executing submitted solutions against the benchmark test cases. Grading is completed before calibration-test splitting and cascade optimization.

C.3. Model Operating Points

Table 3 reports single-model frontier endpoints for the lowest-cost and highest-accuracy models on each dataset.

C.4. Descriptive Pairwise Frontiers

Figure 3 visualizes the full-sample pairwise frontiers and switching points. This figure is descriptive: it shows the geometry of the pairwise-envelope object on the full sample, while the held-out comparison to full fixed chains, optimized subsequences, and routers is reported in Figure 1.

Table 3. Single-model frontier endpoints. Cost is mean dollars per query multiplied by 10^6; accuracy is the dataset-specific correctness metric.

Dataset	Lowest-cost model	Acc.	Cost	Highest-accuracy model	Acc.	Cost
MMLU	Llama 3.1-8B	0.554	12.4	GPT-oss-20B	0.843	79.9
TriviaQA	Llama 3.1-8B	0.579	4.4	Llama 3.3-70B	0.830	40.2
MATH	Llama 3.1-8B	0.145	60.1	DeepSeek-V3	0.818	1398.7
SimpleQA	Llama 3.1-8B	0.044	7.0	GPT-4o	0.382	356.4
LiveCodeBench	Llama 3.1-8B	0.179	70.8	DeepSeek-V3	0.645	1512.1

C.5. Two-Model Sweep Verification

We verify that the k=2 NSGA-II search recovers the same operating points as direct threshold enumeration. For each Pareto-optimal k=2 trial across 50 calibration-test splits, we interpolate the per-pair threshold-sweep frontier at the trial's test-evaluated cost and compare to the trial's test-evaluated quality. The median agreement gap $|U_{\text{trial}} - U_{\text{frontier}}(C_{\text{trial}})|$ is 0.000 on all five datasets, and the 90th percentile is ≤ 0.0014 (Table 4). This confirms that the two-model optimizer is numerically recovering the one-threshold frontier; the analytical FOC in Section 4 characterizes interior points of this frontier under the score-independent cost condition.

Table 4. Two-model sweep verification: frontier agreement gap $|U_{\text{trial}} - U_{\text{frontier}}(C_{\text{trial}})|$ between k=2 NSGA-II solutions and the threshold-sweep frontier, across all Pareto-optimal trials and 50 splits.

	MMLU	TriviaQA	MATH	SimpleQA	LiveCodeBench
Median	0.0000	0.0000	0.0000	0.0000	0.0000
90th pct	0.0008	0.0008	0.0011	0.0009	0.0014

C.6. Full Escalation Benefit Curves

Figure 4 tests the two structural conditions underlying Section 4 on the representative envelope pair per dataset. Expected dominance (Proposition 3.2) holds on 100% of the score support for MMLU, TriviaQA, and SimpleQA. MATH and LiveCodeBench show benefit reversal only in the highest-confidence regions, covering 16–19% of the plotted support, where the cheap model is already sufficiently reliable and escalation can add negligible or negative expected benefit. Decreasing benefit (Definition 4.1) holds on 90–100% of the support: it holds on all plotted support for TriviaQA, at least 97% for MMLU and LiveCodeBench, 94% for MATH, and 90% for SimpleQA. Because empirical costs vary by query and need not be score-independent, these diagnostics support but do not by themselves prove global concavity of the realized token-cost frontier.

Figures 5–9 show the escalation benefit $m_H(s) - m_L(s)$ for all pairs on each dataset. Each panel annotates the expected-dominance fraction (`dom`) and the decreasing-benefit fraction (`dec`). For LiveCodeBench, GPT-oss-20B is excluded from the cascade pool because it is cheaper and more accu-

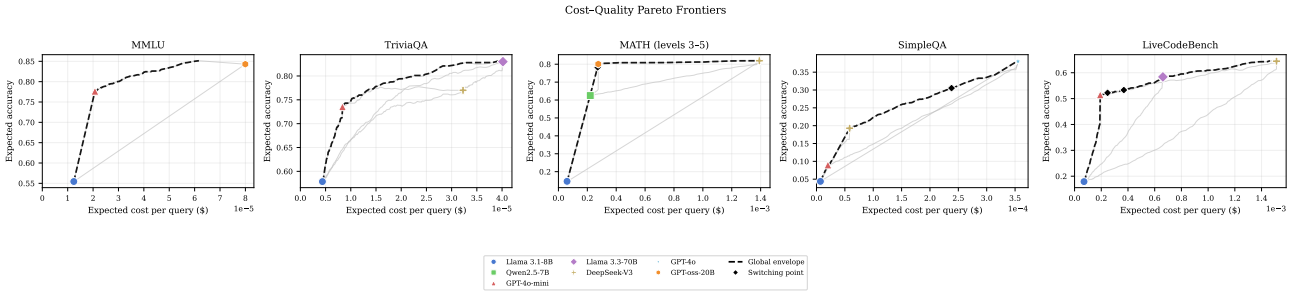

Figure 3. Descriptive cost-quality Pareto frontiers per dataset. Gray curves are per-pair frontiers $U^{\dagger,(i,j)}$ among non-dominated models; shapes mark the corresponding single-model endpoints. Black dashed: pairwise envelope U^*. Black diamonds: switching points. MiniMax-M2.7 is omitted because it is dominated on all datasets. Computed on the full dataset (2,000 examples for MATH, MMLU, TriviaQA, and SimpleQA; 1,055 for LiveCodeBench).

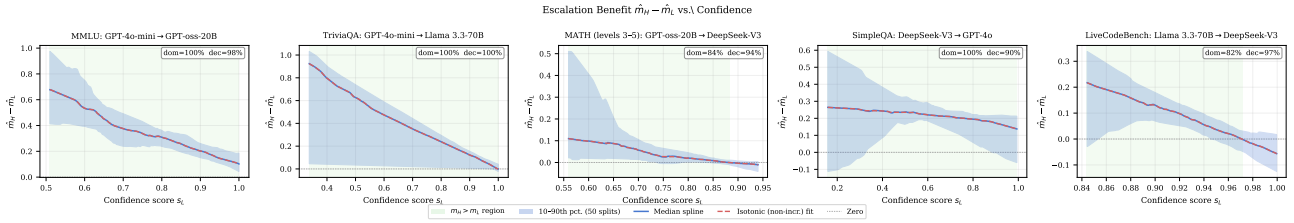

Figure 4. Escalation benefit $m_H(s) - m_L(s)$ as a function of the cheap model's confidence score s_L, for one representative pair per dataset (the pair dominating the largest cost range on the envelope). Solid curves are medians across 50 random 50/50 splits; shaded bands are 10th–90th percentiles. The dotted line marks zero. Annotations report the expected-dominance fraction (dom) and decreasing-benefit fraction (dec) of the median curve.

rate than the higher-cost models, making the high-cost side of the pool degenerate; the figure shows cascade dynamics among the remaining models.

C.7. Cost–Score Independence Check

Per-query costs are computed from realized input and output token counts using the prices in Table 5. Prices reflect public provider pricing at the time experiments were run (May 2026): OpenAI prices for GPT-4o and GPT-4o mini, and Together AI prices for all other models.

Table 5. Token prices used for cost computation, in dollars per million input and output tokens.

Model	Input	Output
Llama 3.1-8B	0.10	0.10
GPT-oss-20B	0.05	0.20
GPT-4o mini	0.15	0.60
Qwen2.5-7B	0.30	0.30
Llama 3.3-70B	0.88	0.88
MiniMax-M2.7	0.30	1.20
DeepSeek-V3	0.60	1.70
GPT-4o	2.50	10.00

Proposition 4.2 assumes that the expensive model's expected escalation cost is independent of the cheap model's confidence score: $\mathbb{E}[C_H(x) \mid s_L(x) = s] = c_H$. This condition permits per-query costs to vary, but rules out systematic variation in expected cost along the thresholding score. To assess this approximation, we compute the Spearman rank correlation between the cheap model's confidence score s_L and the expensive model's realized token cost C_H for every cost-ordered model pair in each dataset.

Table 6. Cost–score independence diagnostic. Each entry summarizes pair-level Spearman correlations between the cheap model confidence score s_L and the expensive model realized cost C_H across cost-ordered pairs. LiveCodeBench has fewer pairs because GPT-oss-20B is excluded from that dataset's cascade pool.

Dataset	Pairs	Median $\lvert\rho_s\rvert$	90th $\lvert\rho_s\rvert$	Max $\lvert\rho_s\rvert$	Share $\lvert\rho_s\rvert < 0.20$
MMLU	28	0.148	0.413	0.614	0.571
TriviaQA	28	0.142	0.390	0.534	0.714
MATH	28	0.150	0.487	0.506	0.607
SimpleQA	28	0.064	0.155	0.165	1.000
LiveCodeBench	21	0.321	0.496	0.526	0.238

Interpretation. The score-independence approximation is most plausible on SimpleQA, where all pair-level correlations satisfy $|\rho_s| < 0.20$. MMLU, TriviaQA, and MATH show weak-to-moderate median correlations but nontrivial tail cases, while LiveCodeBench exhibits the strongest cost–score dependence. These diagnostics explain why Figure 4 should be read as evidence about the benefit side of the concavity condition, not as a proof that the realized token-cost frontiers are globally concave. All empirical frontiers and tables use actual per-query token costs.

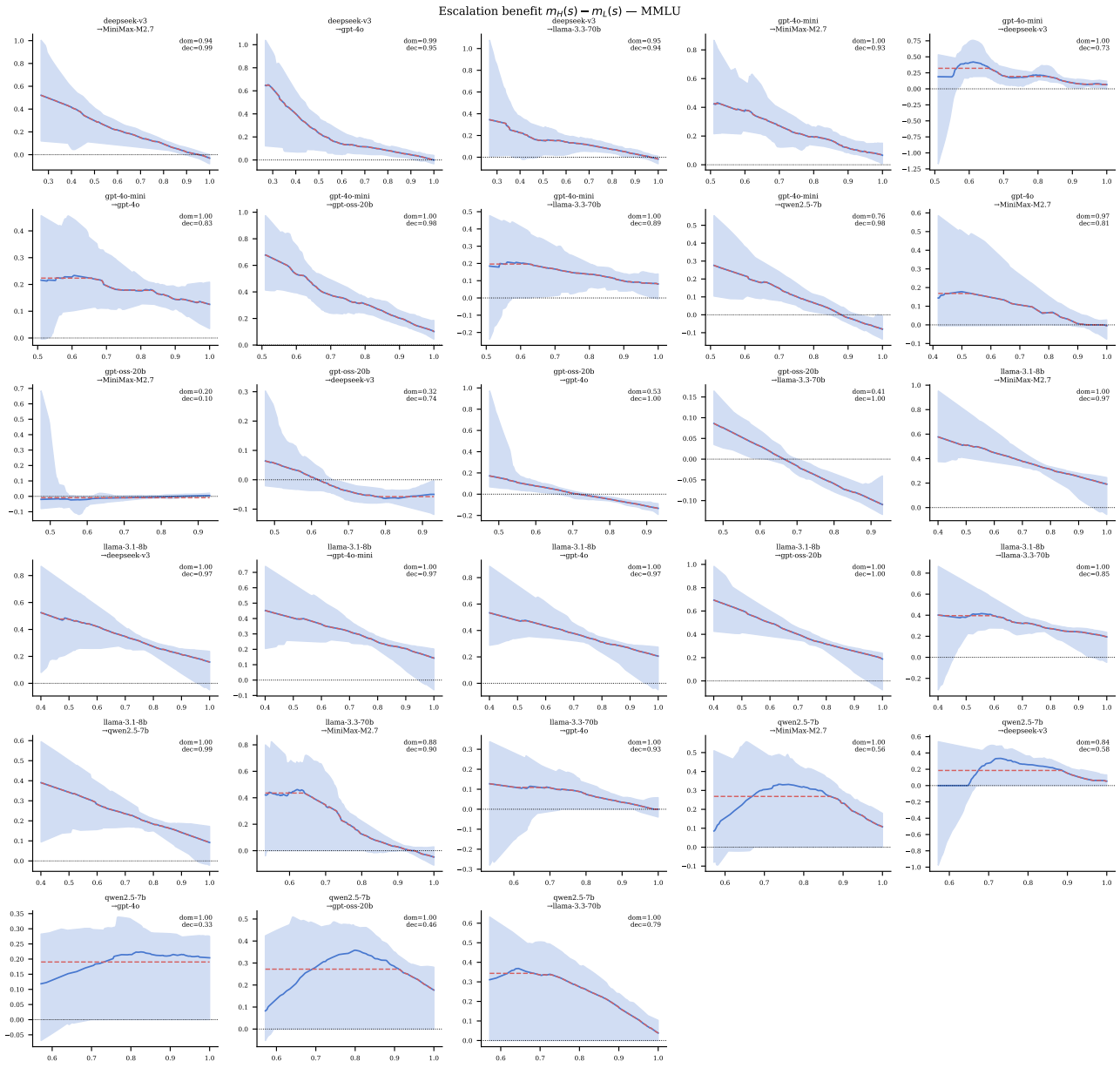

Figure 5. Escalation benefit curves for all 28 pairs on MMLU.

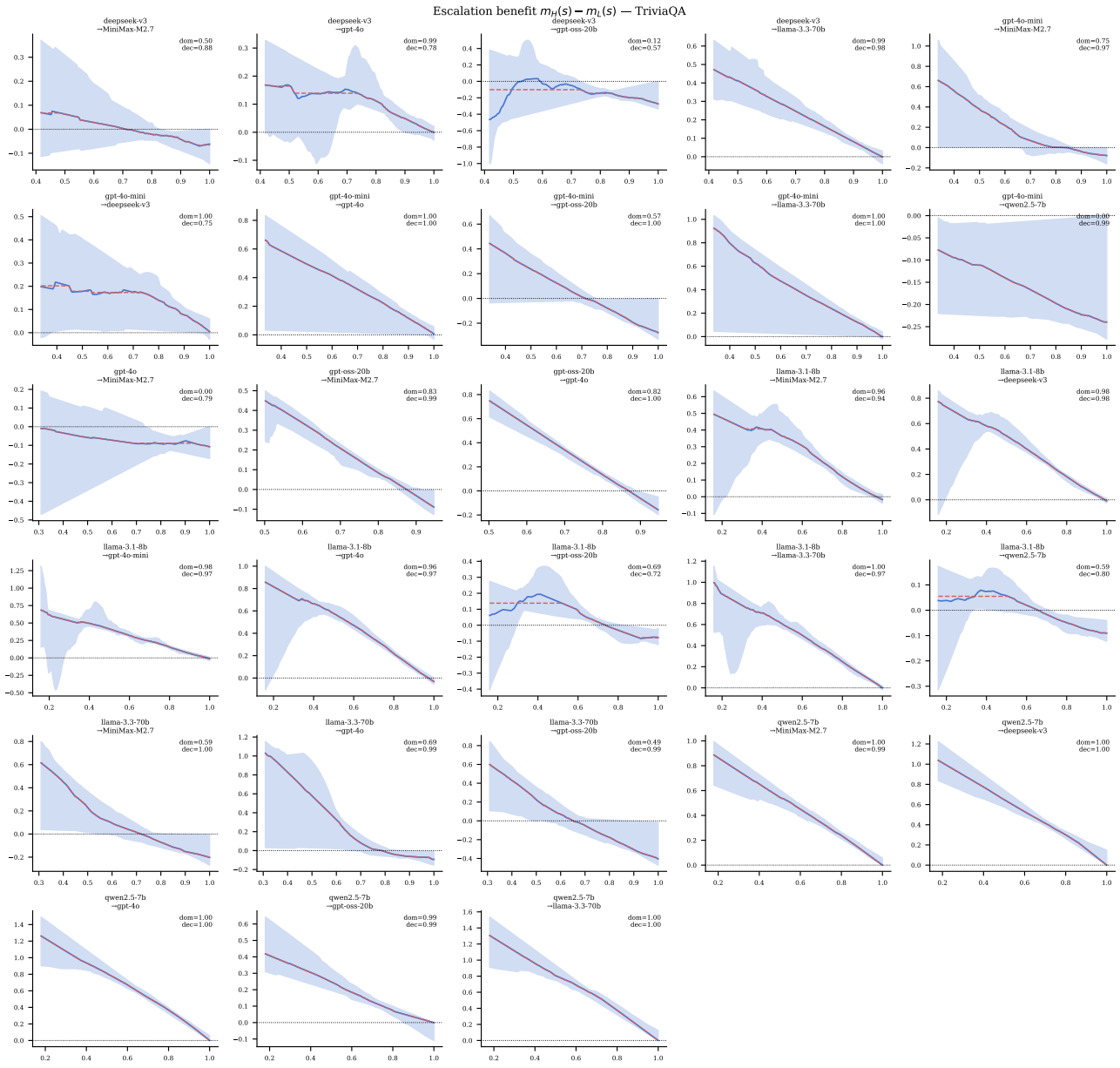

Figure 6. Escalation benefit curves for all 28 pairs on TriviaQA.

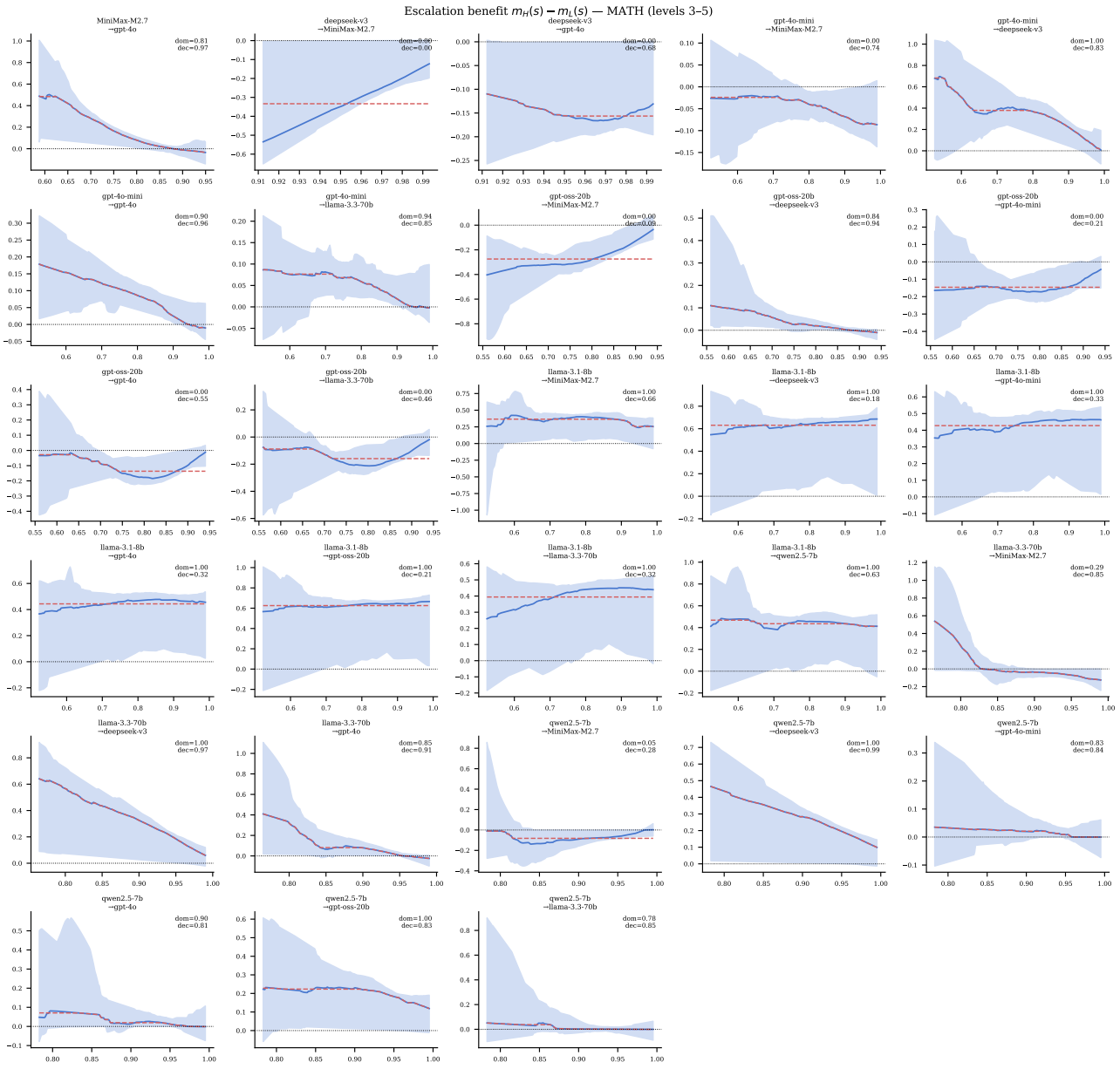

Figure 7. Escalation benefit curves for all 28 pairs on MATH (levels 3–5).

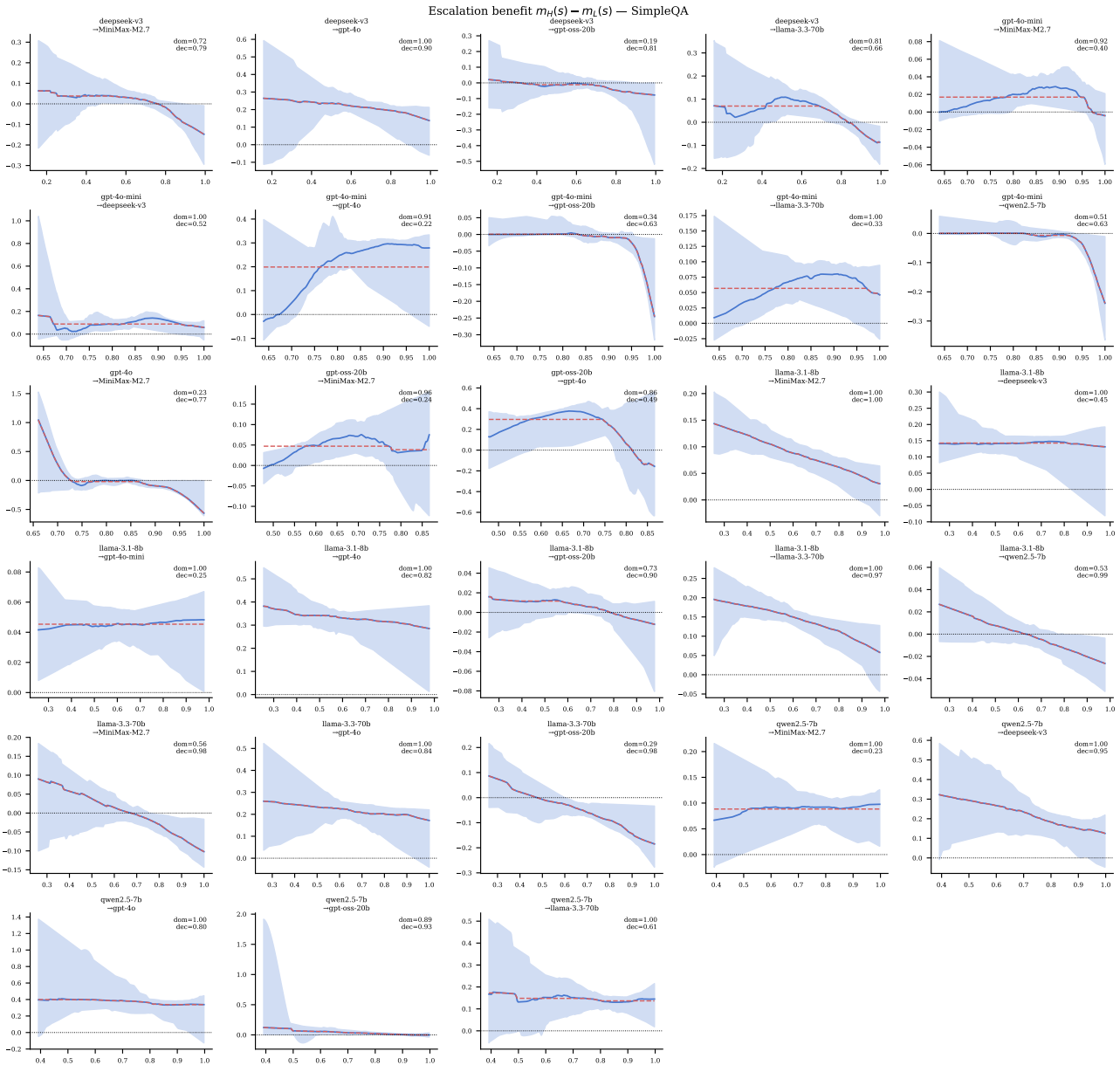

Figure 8. Escalation benefit curves for all 28 pairs on SimpleQA.

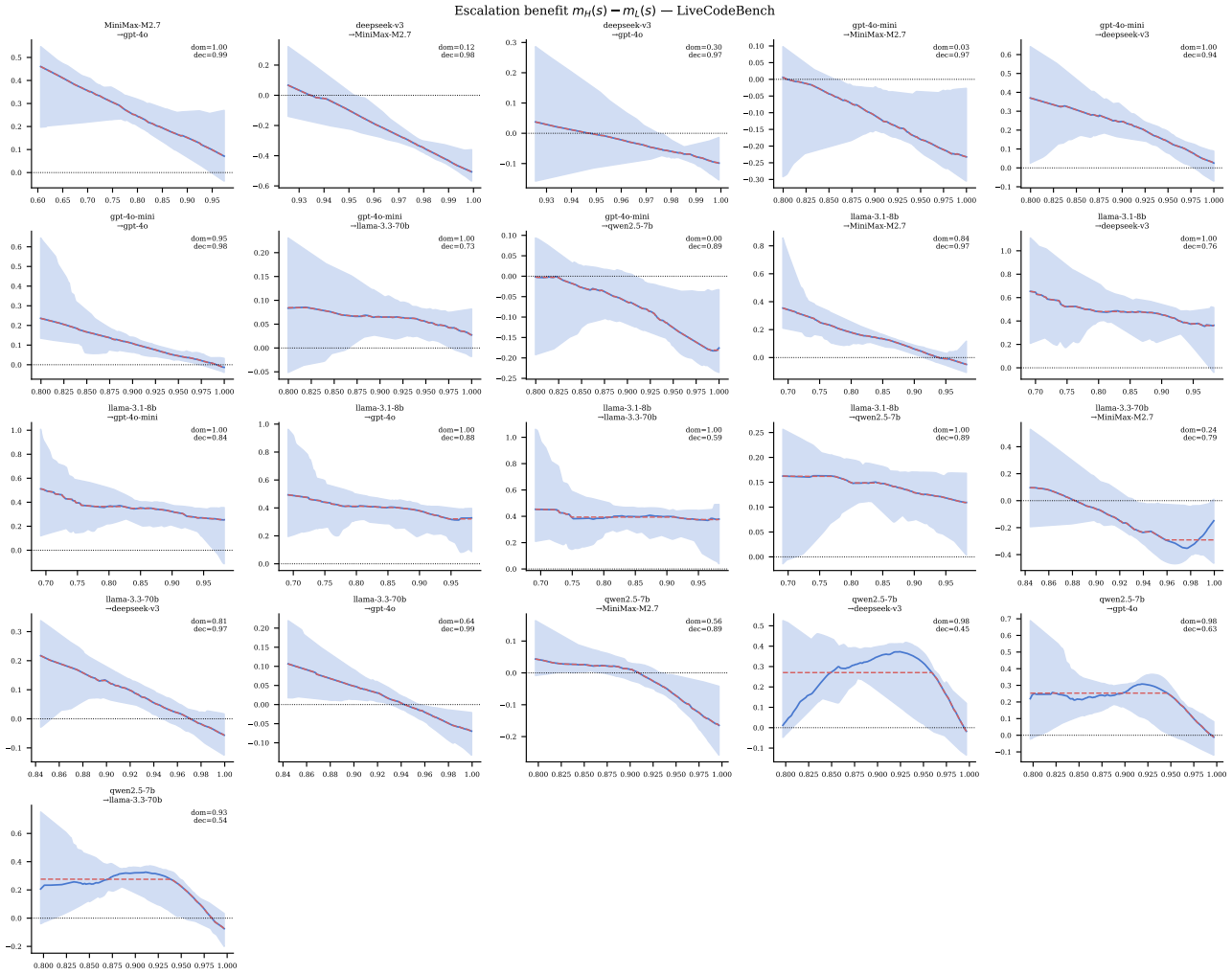

Figure 9. Escalation benefit curves for all 21 pairs on LiveCodeBench (GPT-oss-20B excluded; see text).

C.8. Calibration-Size Sensitivity

Figure 10 and Table 7 report the median quality gap $\Delta = \hat{U}_{\text{sub}} - \hat{U}_{\text{env}}$ at the median operating cost across 50 random splits, as the calibration fraction increases from 50% to 90%. The optimal subsequence cascade uses Optuna NSGA-II with 2,000 trials and population size 100 at every calibration fraction; the number of trials is held fixed so that the only variation is the amount of calibration data available to select candidate pools, valid pairs, thresholds, and model sequences.

Table 7. Calibration-size sensitivity diagnostics. Δ: median quality gap at the median cost (optimal subsequence cascade minus envelope). BWR: band-width ratio (subsequence cascade / envelope), values near 1 indicate similar generalization variance.

	MMLU		TriviaQA		MATH		SimpleQA		LiveCodeBench	
Split	Δ	BWR	Δ	BWR	Δ	BWR	Δ	BWR	Δ	BWR
50/50	−0.000	0.99	+0.003	0.94	−0.002	1.16	+0.002	0.97	−0.002	1.04
70/30	−0.001	0.93	+0.004	0.99	−0.003	1.06	+0.002	1.06	+0.000	1.10
80/20	+0.000	0.97	+0.003	1.03	−0.003	0.99	+0.001	1.02	+0.000	0.93
90/10	+0.000	1.09	+0.000	0.84	−0.005	0.99	+0.000	0.88	+0.000	1.09

Δ remains within about 0.005 accuracy of zero at every calibration fraction. Increasing the calibration fraction does not produce a monotone improvement: the largest positive gaps occur on TriviaQA at 50/50–80/20, while MATH remains slightly negative and LiveCodeBench alternates around zero. BWR shows no systematic decline with increasing calibration fraction, confirming that the optimal subsequence cascade does not generalize better as the calibration set grows. The result is therefore structural: additional calibration data does not reveal a practically meaningful subsequence-cascade improvement over the pairwise envelope.

C.9. Threshold Grid Sensitivity

Table 8 reports the mean and maximum pointwise accuracy difference between each grid resolution and the 500-point reference, across the median envelope over 50 splits. The common interpolation cost grid is fixed at 500 points in all conditions; only the number of threshold candidates swept within each pair changes.

Table 8. Pointwise accuracy difference vs. the 500-point reference ($n_\tau = 500$) for the median pairwise envelope. mean$|d|$ and max$|d|$ are computed over the 500-point cost grid.

	MMLU		TriviaQA		MATH		SimpleQA		LiveCodeBench	
n_τ	mean$\|d\|$	max$\|d\|$	mean$\|d\|$	max$\|d\|$	mean$\|d\|$	max$\|d\|$	mean$\|d\|$	max$\|d\|$	mean$\|d\|$	max$\|d\|$
50	0.0041	0.0130	0.0010	0.0065	0.0006	0.0072	0.0008	0.0081	0.0020	0.0089
100	0.0030	0.0099	0.0006	0.0048	0.0003	0.0042	0.0005	0.0051	0.0010	0.0078
200	0.0018	0.0050	0.0004	0.0040	0.0002	0.0029	0.0002	0.0039	0.0006	0.0046
500	(reference)		(reference)		(reference)		(reference)		(reference)	

The maximum accuracy deviation at the 200-point baseline is ≤ 0.005 on all five datasets, and the mean deviation is ≤ 0.002. Even the 50-point grid stays within 0.014 accuracy everywhere. The 200-point baseline is therefore adequate for all reported results.

C.10. Optimizer Sensitivity

Figure 11 and Table 9 compare two optimizer choices for the optimal subsequence cascade search: NSGA-II (the default used throughout the paper) and random search. Both share the same trial budget of 2,000 evaluations and population size of 100 per split, and results are reported over 50 calibration-test splits. The optimized subsequence search is capped at four models; the full fixed-chain baseline remains uncapped and uses the full calibration-selected non-dominated pool.

The point of this comparison is to test whether the near-zero subsequence-cascade gains depend on NSGA-II's specific search dynamics. If random search, which samples model sequences and threshold parameters uniformly at random, produces a similarly positioned Pareto frontier, the search space is not hiding a materially better subsequence configuration that NSGA-II fails to find. The absence of a practically meaningful improvement is then a structural property of the data rather than an artifact of the optimizer.

Table 9. Optimizer sensitivity diagnostics. Δ: gap between the optimizer's median frontier and the pairwise envelope, averaged over a 20-point window around the median operating cost (same computation as the main frontier comparison). BWR: band-width ratio (optimizer / envelope), measuring relative generalization variance.

	MMLU		TriviaQA		MATH		SimpleQA		LiveCodeBench	
Optimizer	Δ	BWR	Δ	BWR	Δ	BWR	Δ	BWR	Δ	BWR
NSGA-II	+0.0018	0.97	+0.0033	1.06	−0.0025	0.94	+0.0029	0.97	−0.0018	0.93
Random	+0.0031	0.93	+0.0048	1.16	−0.0020	0.92	+0.0029	0.95	+0.0010	1.03

NSGA-II and random search produce very similar Pareto frontiers, with median gaps no larger than about 0.005 accuracy in magnitude. The band-width ratios near 1.0 confirm that the two optimizers also exhibit similar generalization variance. This agreement implies that the search space contains no subsequence-cascade configuration that materially improves on the pairwise envelope: the near-zero result is robust to the optimizer and is not a consequence of NSGA-II failing to find such a configuration.

C.11. Diagnostic Learned Router: Signal Decomposition

The diagnostic learned router trains one logistic regression classifier per calibration-selected non-dominated model using frozen sentence-transformer embeddings (all-MiniLM-L6-v2, 384 dimensions), predicting $P(\text{model}_j \text{ correct} \mid \text{query embedding})$. Similar to the approach by Dekoninck et al. (2025), at inference, each query is dispatched pre-generation to $\arg\max_j [P_j(x) - w\,\bar{c}_j^{\text{cal}}]$ for a scalarization weight w swept over a log-uniform grid, where $\bar{c}_j^{\text{cal}}$ is the model's mean cost on the calibration split. The resulting cost-quality Pareto frontier is evaluated using realized held-out token costs after dispatch, over 50 calibration-test splits. Classifiers, non-dominated pools, and valid pair sets are fit

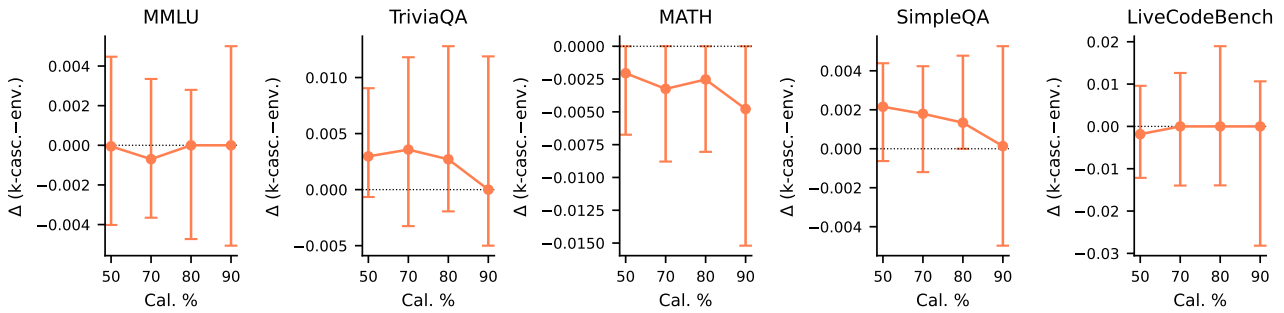

Figure 10. Median quality gap Δ (optimal subsequence cascade minus pairwise envelope) at the median operating cost, as a function of calibration fraction. Error bars span the 10th–90th percentile of per-split deltas across 50 random splits. The dotted line marks $\Delta = 0$. For LiveCodeBench, GPT-oss-20B is excluded from the model pool.

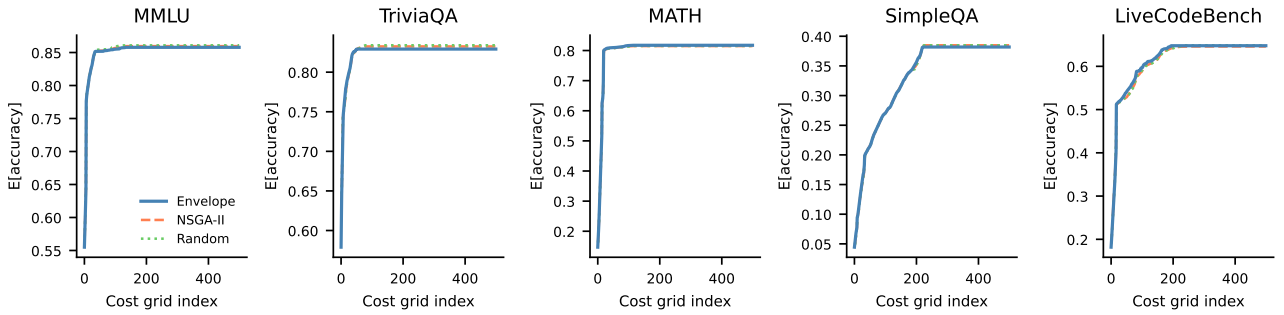

Figure 11. Median Pareto frontier (over 50 splits) for NSGA-II and random search versus the pairwise envelope. Both optimizers produce very similar frontiers across all five datasets. For LiveCodeBench, GPT-oss-20B is excluded from the model pool.

on the calibration half of each split using the same stratified 50/50 design as all other comparisons, then evaluated on the held-out half.

Table 10 decomposes the router's advantage by isolating signal quality from structural differences. We evaluate three methods, all using the same logistic-regression classifiers: (i) **UQ cascade**: the pairwise envelope with mean token negentropy as the deferral signal (post-generation); (ii) **Embedding cascade**: the pairwise cascade with $P(\text{cheap correct} \mid \text{embedding})$ as the deferral signal (pre-generation, pairwise structure); (iii) **Diagnostic learned router**: pre-generation dispatch to a single model (pre-generation, k-model structure).

The embedding cascade is a weaker deferral signal than mean token negentropy on 4/5 datasets (AUROC 0.49–0.73 vs. 0.65–0.77), yet the router (using the same embedding classifiers) exceeds the UQ cascade on 4/5 datasets. The same-signal comparison (router vs. embedding cascade) shows positive gaps on all five datasets. The advantage is therefore structural: the router avoids the cheap model's generation cost c_L on queries routed to other models, whereas any pairwise cascade always pays c_L first. TriviaQA is the exception (AUROC ≈ 0.49), where query embeddings carry near-zero information about correctness and no routing advantage can compensate. This baseline is intended to diagnose the structural difference between pre-generation dispatch and post-generation cascading, not to benchmark the state of the art in learned routing.

Table 10. Signal decomposition. Δ = median quality gap vs. UQ cascade (pairwise envelope); AUROC = mean over pool models of ROC-AUC for the routing signal against binary correctness labels, averaged over 50 splits.

	MMLU		TriviaQA		MATH (levels 3–5)		SimpleQA		LiveCodeBench	
Method	Δ	AUROC	Δ	AUROC	Δ	AUROC	Δ	AUROC	Δ	AUROC
UQ cascade (envelope)	+0.0000	0.714	+0.0000	0.746	+0.0000	0.647	+0.0000	0.699	+0.0000	0.773
Embedding cascade	−0.0079	0.657	−0.0200	0.490	+0.0000	0.715	−0.0036	0.608	−0.0082	0.725
Diagnostic learned router	+0.0027	0.657	−0.0319	0.490	+0.0022	0.715	+0.0142	0.608	+0.0181	0.725