STRATEGIC PRICING IN LLM ROUTING MARKETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Routing systems for large language models choose among models and collaboration modes to balance answer quality against cost. They take supplier prices as given. When suppliers set prices strategically, the router defines a game with two difficulties. Its equilibrium need not be pure, and a single trained router can reward a supplier for raising its own price. We compute equilibria by fictitious play and accept a profile only after checking every unilateral deviation on the bid grid. We learn answer quality once, independently of price, and add payments through an economic layer, so a supplier's demand cannot rise with its own price. We then study two platform controls. A committed outside option gives the customer side bargaining power. Under a router that picks the best mode, it raises customer surplus whenever its committed value exceeds baseline surplus, and repricing cannot remove that gain. In simulations with model accuracies calibrated to three open-weight models, its benefit rises with its capability and falls with its price. A capability-blind near-tie rule changes 9.1% of allocations at a moderate threshold and raises the least-used supplier's share in every run, at a bounded direct cost. Menu expansion buys contestability. Redistribution within the menu buys inclusion.

1 INTRODUCTION

Large language models are increasingly accessed through routing systems rather than as isolated services. Given a query, a router may select one model, invoke several models sequentially, or aggregate several responses. Systems such as MasRouter, RouteLLM, and FrugalGPT demonstrate that this demand-side choice can substantially improve the quality–cost frontier (Yue et al., 2025; Ong et al., 2025; Chen et al., 2024). Their usual input, however, is a set of supplier prices treated as fixed. Once model providers can adjust those prices strategically, routing and pricing become a coupled market. The router determines demand at each bid profile, and anticipated demand determines supplier bids.

Routers act for the customer. For each query they choose the model or collaboration mode that gives the best answer for the money. They cannot influence the money. Strategic suppliers price above cost, and better routing does not change that, because a router can only choose among the offers it receives. This leaves out the lever a buyer normally holds against its suppliers. A platform that aggregates many customers' demand can commit to an outside option: a model it operates or contracts at a fixed quality and price. The commitment gives the customer side bargaining power, because every incumbent supplier now competes against an option it cannot price away. We call this option the *anchor*.

A better and cheaper option helps at fixed prices. The open questions are whether the customer's gain survives the incumbents' strategic response, and how capable and how cheap the anchor must be. Answering them requires two tools that router research does not provide. The suppliers' equilibrium need not be pure, so it has to be computed and verified as a mixed profile. And a learned router can reward a supplier for raising its own price, so price counterfactuals need an allocation rule whose own-price response is signed. We also study a second, narrower control that keeps the menu fixed: a capability-blind near-tie rule that spreads traffic among nearly equivalent modes.

Our contributions are fourfold.

- **A supply-side model of LLM routing.** Privately informed suppliers submit sealed first-price bids to a fixed router that picks a collaboration mode (model, chain or ensemble) by customer

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.

surplus. Unlike Stackelberg pricing (Guo et al., 2026) or full-information API-module competition (Bhawalkar et al., 2025), we add private costs, quality-aware mode routing and in-router platform controls (Section 3).

- **Certified equilibria and readable price effects.** We accept a mixed equilibrium only if no supplier gains by deviating on the bid grid (Section 4). A structured router, where a supplier's demand cannot rise with its own price, makes price counterfactuals interpretable (Proposition 1).
- **Customer-side bargaining power with a guarantee.** Under the hard rule, an outside option (the *anchor*) whose committed value exceeds baseline surplus raises customer surplus at every post-entry equilibrium, so incumbents cannot reprice the gain away (Theorem 2). A capability-blind near-tie rule spreads traffic among near-equivalent modes at a threshold-bounded direct cost (Proposition 2).
- **Findings in simulation.** With accuracies calibrated to three open-weight models, equilibria are mixed and best-response dynamics rarely converge. At the strong-anchor endpoint, the anchor raises customer surplus under every allocation rule we test, almost entirely through the new option under price-coherent rules, and lowers incumbent surplus. In the learned-cache sweep, the benefit rises with capability and falls with price. The near-tie rule raises the least-used supplier's share in every run at under 1% of its cost bound (Section 6).

2 Related Work

Demand-side routers choose models or collaboration modes at given prices (Yue et al., 2025; Ong et al., 2025; Chen et al., 2024; Feng et al., 2025). Two lines make prices endogenous. Guo et al. (2026) model LLM pricing as a Stackelberg game led by providers' public prices. Bhawalkar et al. (2025) cast API modules' posted prices as a first-price procurement auction and show that a bang-per-buck rule makes full-information equilibria approximately efficient. We instead fix a quality-aware router, give suppliers private costs, and study their mixed equilibria and the platform's commitments. We draw on procurement auctions (Krishna, 2010; Che, 1993; Asker & Cantillon, 2008), reserve prices (Myerson, 1981), and mixed-strategy price dispersion (Varian, 1980; Burdett & Judd, 1983; Tirole, 1988), not asynchronous or collusive cycles (Maskin & Tirole, 1988; Green & Porter, 1984). The anchor is a committed outside good with a quality and a price (Salop, 1979; Mussa & Rosen, 1978), a buyer-side reserve price, since competition substitutes for bargaining power (Bulow & Klemperer, 1996). We use fictitious play (Brown, 1951; Robinson, 1951) on a discretized game (Athey, 2001), finite-game existence (Nash, 1950; Harsanyi, 1967), and monotone comparative statics (Topkis, 1978).

3 Strategic LLM Routing Market

We formalize a sealed-bid first-price reverse auction layered over a multi-agent LLM router. The platform implements the routing rule, strategic suppliers submit type-contingent price multipliers, and the customer chooses a collaboration mode using quality and total payment. The formulation separates the customer allocation problem from suppliers' private costs, enabling the cached equilibrium computation in Section 4.

3.1 Modes and Payments

Each query is served by one collaboration mode, such as a single supplier, a chain, or an ensemble. The mode determines which suppliers are invoked and how many billable tokens each contributes. Each invoked supplier is paid its bid times its billable tokens at the list price, so the customer's total payment adds up across the suppliers in the mode. Appendix A gives the mode set and the payment accounting. All suppliers have the same list price in every run reported here.

Terms. A *supplier* is a model provider that sets its price. The *customer* is the party whose queries the router serves, and the *platform* runs the router. We assume the platform acts for the customer. Its router maximizes customer surplus and it keeps no margin. A *collaboration mode* is one way to

answer a query (one model, a chain of two, or an ensemble), and the *menu* is the set of modes the router may choose from. A supplier's *bid* is a multiplier on its list price. A *mixed strategy* randomises over bids. A profile of such strategies is a *mixed equilibrium* if no supplier gains by deviating, and its *certificate* is the largest gain found by checking every deviation on the bid grid. The *anchor* is a non-strategic option that the platform adds to the menu at a committed quality and price. Once it is present, the strategic suppliers are called *incumbents*. *Contestability* means that incumbents lose traffic to an option the customer can actually use. We measure it by the anchor's routing mass and the fall in incumbent surplus. *Inclusion* is the least-used supplier's share of calls.

Notation. $\mathcal{S}_0$ is the baseline menu and $\mathcal{S}^D$ the menu after the anchor is added. $\mathcal{T}_\epsilon$ is the near-tie subset of $\mathcal{S}_0$. A bid profile is $\mathbf{d} = (d_1, \ldots, d_n)$, and d_D is the anchor's committed price. $v^0(q, S; \mathbf{d})$ is the customer's value of serving query q by mode S at those bids, v^D is the same on the augmented menu, and $\overline{v}_D$ is the expected value of the anchor's solo mode. $\alpha(q, S)$ is the correctness of mode S on query q in the simulator, r is the reward for a correct answer (set to 1), and λ converts payment into accuracy units. η is the temperature of the soft (entropy-regularized) rule, π_η its allocation, and V_η its value. $F_i(S)$ is supplier i's base fee in mode S, that is, its billable tokens at the list price, and X_i is its expected base fee under the router's allocation. κ_ϵ is the fraction of queries that the near-tie rule changes.

Surplus accounting. We use *welfare* for the joint accounting of the market's outcomes and name the specific quantity whenever we report a number. *Customer surplus* $\mathrm{CS} = \mathbb{E}[r\alpha - \lambda P]$ is correctness minus λ times payment, in accuracy units. Its monetary equivalent is CS/λ. *Supplier surplus* $PS_i = \mathbb{E}[d_i F_i - c_i \,\mathrm{tok}_i]$ is in money, and so is *anchor revenue*, the anchor's fee times its routing mass. *Total surplus* $W = \mathrm{CS}/\lambda + \sum_i PS_i +$ anchor revenue is formed in money, where payments cancel as transfers. We price the anchor's own cost at zero, so W overstates total surplus by that cost. Every theorem and every reported effect below concerns CS unless it says otherwise.

A running example. Take suppliers A, B and C with list price 10^{-5} per token and base token counts 800, 850 and 820 (Appendix E). For illustration only, let every invoked supplier contribute its base tokens once. The base fees are then $F_A = 0.0080$, $F_B = 0.0085$ and $F_C = 0.0082$. At bids $\mathbf{d} = (0.9, 1.0, 0.8)$, the customer pays 0.0072 for Solo_A, 0.0157 for the chain A→B, and 0.0223 for the ensemble of all three. With $r = 1$ and $\lambda = 35$, a mode with accuracy α is worth $\alpha - 35P$. The chain is therefore chosen over Solo_A only if it is at least $35 \times 0.0085 \approx 0.30$ more accurate. Supplier A has private cost 4×10^{-6} per token, so it earns 0.0040 each time a mode that invokes it is chosen. Its bid trades that margin against the chance of being chosen, given its rivals' bids.

3.2 Timing

Suppliers submit type-contingent bids without observing rivals' private costs. The router then observes the query and the submitted profile and allocates by its rule. The information structure separates the customer's per-query problem from the suppliers' pricing problem, which is what makes the cached representation of Section 4 possible. Full timing and the customer's mode choice are in Appendix A.4.

3.3 Supplier Utility

Given its rivals' strategies, a supplier's interim utility at its private cost is its expected revenue minus its cost under the router's allocation. A Bayesian Nash equilibrium (BNE) is a profile of type-contingent bid functions from which no supplier gains by deviating. Appendix A gives the full statement.

Lemma 1 (Additive separability)**.** *For every supplier i,*

$$U_i(\mathbf{d}_i, \sigma_{-i}; c_i) = \sum_{\tau \in \mathcal{T}} \Pr_{\mathcal{D}}(\tau) U_i^\tau(d_i^\tau, \sigma_{-i}^\tau; c_i), \tag{1}$$

where U_i^τ depends on strategies only through their type-τ components.

Separability holds because no constraint couples query types. A capacity constraint on suppliers would break it first.

Lemma 1 allows each query type to be solved as a one-dimensional Bayesian bidding subgame and the resulting strategies to be reassembled. It is a computational property, not a claim that task types share the same equilibrium.

3.4 Discretized Game

The experiments replace the continuous type and action spaces with finite cost and bid grids and precompute the customer's allocation for every bid profile into a router cache. Theorem 1 applies to this finite game. The grid construction and the cache's structure are in Appendix A.

Theorem 1 (Mixed equilibrium in the discretized game)**.** *For finite $\mathcal{C}_d$, D_d, N, and $\mathcal{T}$, the discretized strategic LLM routing game admits at least one mixed-strategy BNE.*

The proof is in Appendix B.

4 Mixed-Equilibrium Recovery

Iterated best response (IBR), which answers the opponents' latest pure actions, cannot settle on a mixed equilibrium. We therefore compute equilibria by fictitious play (FP), which answers the opponents' empirical action distributions. The market, the information structure and the router are unchanged.

4.1 Cached Routing Games

By additive separability, equilibrium computation runs independently for each query type. For each type the customer's allocation is precomputed at every bid profile on the finite grid, so the supplier game reduces to a finite normal-form game over a cached allocation table. The construction is in Appendix C.

4.2 Fictitious Play

Fictitious play replaces the latest pure action with the empirical distribution of each opponent's past bids. Every round, each supplier best-responds to those beliefs, and the beliefs are updated with the realized action. The tail of the trajectory is time-averaged into a mixed profile. The recursions, hyperparameters, and convergence diagnostics are in Appendix C.

4.3 Certification

A recovered profile is accepted only after enumerating, for every supplier and every grid point, the gain from deviating unilaterally while opponents hold their recovered mixed strategies. The maximum such gain is the profile's ϵ-BNE residual. A profile is accepted on that certificate, not on the dynamic that produced it. We claim no convergence theorem for fictitious play in this game class, and the rounds are a computation, never repeated play. Section 6 compares the certified profile with a time-averaged best-response cycle on the same game. The enumeration procedure and stability treatment are in Appendix C.

4.4 Interpretation

The recovered market therefore does not fail to settle. It settles on dispersion, with suppliers randomizing over a range of prices. On the tested grid every feasible bid exceeds every private cost. The lowest bid is 8×10^{-6} per token and the highest cost is 6×10^{-6}, so positive margins hold by construction. The equilibrium determines how prices spread over that range. The router takes these prices as given, and this price-taking is the premise the rest of the paper responds to.

5 STRUCTURE-PRESERVING PLATFORM INTERVENTIONS

We study two platform interventions. The anchor adds a non-strategic provider to the menu, after which the suppliers' equilibrium is recomputed. The near-tie rule keeps the menu and redistributes probability among nearly equivalent modes. Both are evaluated under hard, entropy-regularized and structured allocation rules with the same direct-versus-response decomposition.

5.1 THE ANCHOR

The platform adds a non-strategic provider D with committed capability α_D and committed price d_D, expanding the menu with an anchor solo mode and anchor-directed chain modes. The anchor's fee enters the customer's payment on the same terms as an incumbent's. The hard rule selects the menu-maximising mode at the observed prices. The entropy-regularized rule replaces that argmax with a temperature-weighted allocation over the same menu, so the customer's choice is smooth in value differences. Both are defined on the augmented menu. The hard rule's definition is in Appendix C. The log-sum-exp form of the regularized objective and the identity separating raw customer value from the entropy term are in Appendix B.

Each formal result below holds for one allocation rule. Proposition 3, Theorem 2 and its corollary concern the hard rule. Propositions 4 and 1 concern the soft rule, and Proposition 2 concerns the near-tie rule. We make no claim about routers in general. At fixed incumbent bids, adding the anchor cannot lower the hard rule's optimized value on any query, because the augmented menu contains every baseline mode with its original value (Proposition 3, Appendix B).

A stronger statement holds after incumbents respond, because the anchor's solo mode has a value that does not depend on incumbent bids. Let b^0 denote a baseline equilibrium and b^D an equilibrium after the anchor is introduced. Let $\mathrm{CS}^0(b)$ and $\mathrm{CS}^D(b)$ denote expected raw customer surplus under the baseline and augmented hard-allocation environments, respectively. Define the committed solo-anchor value

$$\overline{v}_D \triangleq \mathbb{E}_Q[r\alpha(Q, \mathrm{Solo}_D) - \lambda d_D F_D(\mathrm{Solo}_D)]\,. \tag{2}$$

In the running example an anchor with 820 tokens at $d_D = 0.9$ costs the customer 0.0074 per Solo_D query, or 0.26 in accuracy units.

Theorem 2 (Sufficient strong-anchor condition)**.** *Suppose the augmented router uses the hard allocation in Eq. equation 37, every baseline mode is retained, and the anchor's capability and bid are fixed. If $\overline{v}_D > \mathrm{CS}^0(b^0)$, then the anchor raises equilibrium customer surplus. More precisely, $\mathrm{CS}^D(b^D) - \mathrm{CS}^0(b^0) \geq \overline{v}_D - \mathrm{CS}^0(b^0) > 0$.*

Proof sketch. At every incumbent bid profile, the hard rule can select Solo_D, whose value does not depend on incumbent bids, so $\mathrm{CS}^D(b^D) \geq \overline{v}_D$ at any post-entry equilibrium, subtracting $\mathrm{CS}^0(b^0)$ gives the bound (Appendix B). □

Because the bound holds at every incumbent bid profile, it holds at every post-entry equilibrium, mixed or pure. The condition refers to the baseline equilibrium that is reached. If several can be reached (solver realizations can reach profiles with different surplus), requiring $\overline{v}_D$ to exceed the largest baseline surplus makes the condition independent of which one is reached.

With a solo anchor of constant accuracy α_D and fee $\overline{F}_D$, the condition becomes $\alpha_D > (\mathrm{CS}^0(b^0) + \lambda d_D \overline{F}_D)/r$, a sufficient capability threshold that rises with the anchor's price (Corollary 1, Appendix B).

The condition is sufficient rather than necessary and can be conservative. It uses only the solo anchor and ignores any additional value created by anchor–incumbent chains. Its advantage is robustness to strategic response. Once the committed anchor alone exceeds baseline equilibrium surplus, incumbent repricing cannot remove the customer's option to use it. The capability–price sweep in Section 6.3 examines the corresponding empirical boundary.

The hard rule can switch abruptly when two modes have similar values. We therefore also use a soft, entropy-regularized rule. Its temperature is $\eta > 0$, a symbol chosen to avoid confusion with the

query type τ. The soft allocation is

$$\pi_\eta(S \mid q, \mathbf{d}) = \frac{\exp\{v(q, S; \mathbf{d})/\eta\}}{\sum_{S' \in \mathcal{S}} \exp\{v(q, S'; \mathbf{d})/\eta\}}. \tag{3}$$

Adding the anchor weakly raises this rule's regularized value (Proposition 4, Appendix B). Its raw customer surplus can still move slightly in either direction, because smoothing puts probability on modes that are not the best. We therefore report raw customer surplus.

5.2 The Structured Router

The structured router learns answer quality once and adds payments exactly. A predictor of mode quality is trained once from query features and mode identities and then held fixed across all price profiles. The router scores each mode by predicted quality minus its payment at the observed bids and applies the soft allocation to that score. Appendices C.5 and E describe training, and Appendix I.1 held-out evaluation.

Enforcing coherence rather than assuming it. A flexible learned router can predict query–mode quality accurately at every bid profile it sees and still, across profiles, send more billable demand to a supplier that raised its price. Nothing in its training objective forbids that. One response is to assume monotone demand and audit how often it fails. We instead make the audit unnecessary. Quality is learned once, independently of price, and payments enter through an economic layer whose own-price derivative is signed by construction.

We report the price-conditioned learned cache as a baseline, not as independent confirmation, because its price response is the object that the next proposition makes interpretable. On the discretized grid, the hard rule, both soft-rule temperatures and the structured router pass all 144 adjacent own-price comparisons in both markets. The learned cache passes 101 of 144 without the anchor and 108 of 144 with it. These counts describe one trained cache. Averaged over retrained caches, the violations are consistent with noise. Monotone demand is therefore a statistical property of a learned cache and a guaranteed property of the structured router. The four rules also differ in what they need to run. The hard and soft rules use the simulator's true per-query quality, so they are reference allocations rather than deployable routers. The learned cache is deployable but has no guarantee. The structured router is deployable and carries the guarantee of the next proposition.

External routers such as RouteLLM, FrugalGPT and GraphRouter choose a model at given prices, without collaboration modes or strategic suppliers. They answer a different question from ours: which model to pick, rather than how suppliers price into a fixed rule. Any of them could adopt the economic layer. Its cost is whatever routing value the price-independent quality head loses. In the anchor experiment, the structured router reproduces the effect of the reference rules (Table 7)

Proposition 1 (Own-price coherence)**.** *For the allocation in Eq. equation 38, for every query type τ and holding all other bids fixed,*

$$\frac{\partial X_i^\tau(\mathbf{d})}{\partial d_i^\tau} = -\frac{\lambda}{\eta}\, \mathbb{E}_{q|\tau}\big[\mathrm{Var}_{S \sim \widehat{\pi}_{\psi,\eta}(\cdot|q,\mathbf{d})}[F_i(S)]\big] \le 0. \tag{4}$$

Proof. For a fixed query q of type τ, differentiating the softmax gives $\partial \widehat{\pi}(S \mid q)/\partial d_i^\tau = -(\lambda/\eta)\widehat{\pi}(S \mid q)[F_i(S) - X_i(q)]$. Multiplying by $F_i(S)$ and summing over modes yields minus the conditional variance. Averaging over queries of type τ gives Eq. equation 4. □

This result concerns aggregate billable-resource demand, the quantity entering supplier revenue and the monotonicity audit. The aggregate derivative is the query-average of conditional variances. A higher own bid cannot increase a supplier's expected allocation under the structured economic layer. Prediction error may affect which modes are attractive, but it does not overturn this price-response structure.

5.3 Direct and Response Effects

The anchor's effect on customer surplus splits into a direct component (the gain from the expanded menu at unchanged incumbent bids) and a strategic-response component, the further change once suppliers re-solve. We report the two separately.

5.4 Near-Tie Allocation

The second lever changes how the platform distributes traffic when several existing modes are close in calibrated customer value, without adding a supplier or altering mode quality. For a query and bid profile, the near-tie set collects the modes within ϵ_{tie} of the best available value. The policy-conditioned rule intervenes only where the learned cache's own choice already lies in that set, redistributing its probability uniformly across the set and leaving every other allocation unchanged.

For query q and bid profile $\mathbf{d}$, let $v^*(q, \mathbf{d}) = \max_{S \in \mathcal{S}_0} v(q, S; \mathbf{d})$ and define the near-tie set

$$\mathcal{T}_\epsilon(q, \mathbf{d}) = \{S \in \mathcal{S}_0 : v^*(q, \mathbf{d}) - v(q, S; \mathbf{d}) \le \epsilon_{\text{tie}}\}. \tag{5}$$

Proposition 2 (Bounded direct cost of near-tie redistribution)**.** *At fixed supplier bids, the pointwise reduction in calibrated customer value caused by the policy-conditioned rule is at most ϵ_{tie} on an affected query and is zero on an unaffected query. Consequently,*

$$\text{CS}^{\text{tie}}(\mathbf{d}) - \text{CS}^0(\mathbf{d}) \ge -\kappa_\epsilon \epsilon_{\text{tie}}. \tag{6}$$

This guarantee is local and direct, and Appendix B proves it. It controls the allocation cost while supplier bids are fixed, but it does not sign the response term after suppliers recompute equilibrium strategies. We therefore evaluate tie-breaking with the same decomposition as in Eqs. equation 16–equation 18, together with the minimum supplier share, supplier concentration, and mode concentration. The threshold ϵ_{tie} is a platform design parameter. A larger value makes intervention more frequent and broadens the set of eligible modes, but also relaxes the direct value guarantee.

6 Mixed-Equilibrium and Welfare Results

We report three experiment blocks. The first is a diagnostic of the equilibrium computation of Section 4. It asks whether mixing remains after a long run and whether the recovered profile passes the certificate. The second is the anchor experiment of Section 5. It asks how a committed outside option changes customer surplus, equilibrium bidding and the allocation of demand. The third keeps the menu fixed and asks how the near-tie threshold changes inclusion, concentration and customer surplus. The diagnostic uses a different price sensitivity, so we report it separately rather than pool it with the other two.

What the simulations are for. The formal results of Sections 3–5 hold for any accuracy profile, list price, cost grid and price sensitivity, and the simulations do not prove them. The simulations serve three purposes. They show that the equilibrium computation and its certificate work on a realistic instance, they illustrate the size and structure of the effects, and they check consistency with the theory. Model accuracies are calibrated to public benchmarks of three open-weight models, so that capability gaps and ties resemble real ones. We set the list prices, token counts, cost grid, λ and the anchor's capability. Because model capabilities change with every release, we report structure (signs, orderings and slopes), not constants to carry to other markets.

6.1 Recovering the Equilibrium

The recovered equilibrium is mixed, and its certificate is small. On the diagnostic subgame (knowledge queries, a five-point bid grid, $\lambda = 350$, 2,000 rounds), fictitious play returns a profile with a certificate of 2.5×10^{-6}. Eleven of its 15 supplier–cost cells keep mixed support. Replacing each mixed strategy by its most frequent bid does not give a best-response profile, so the mixing matters for incentives.

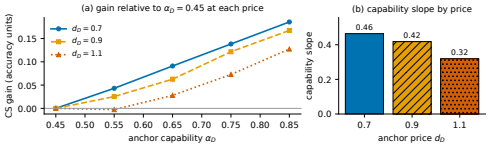

Figure 1: Structure of the anchor's effect in the learned-cache sweep. Panel (a) plots the customer-surplus gain at each committed capability *relative to the weakest anchor at the same committed price*, so a uniform shift in the baseline leaves the curves unchanged. The gain rises with capability at every price, except a small dip between the two weakest anchors at $d_D = 1.1$. Panel (b) reports the capability slope at each price, as printed in Table 5. Points are six-seed means in accuracy units. Appendix F reports levels and zero-crossings, which depend on the baseline calibration.

The certified profile also beats the alternative representation. On the seven-point configurations of the cycle comparison, the certificate is 8×10^{-5}, a different game from the diagnostic subgame. There, time-averaging a best-response cycle leaves a largest deviation gain of 1.6–2.7 $\times 10^{-4}$, a factor of 2.0 to 3.4 worse. Pure best-response dynamics rarely converge, and different solver realizations of the same game can reach profiles with different customer surplus. A profile is therefore accepted on its certificate, not on the dynamics that produced it.

Which cells mix depends on the grid. In the grid-refinement audit, the five-point grid yields six mixed cells, the seven-point grid none at the 0.95 threshold, and the nine-point grid eight (Table 3). We therefore claim that the calibrated game can require a mixed representation, not that mixing appears at every resolution. Each certificate is a statement about its own finite game. Appendix D reports per-cell results, the grid-refinement audit and the price-pressure sweep.

The diagnostic subgame is the $\lambda = 350$ point of a price-pressure sweep over eleven values of λ from 35 to 3500. Across the sweep, the certificate stays between 8.4×10^{-7} and 8.1×10^{-6} while the number of mixed cells varies non-monotonically. Fictitious-play drift has not clearly decayed at four of the eleven values, which is why acceptance rests on the certificate (Figure 2, Appendix D.1).

6.2 Anchor Setup

The anchor sweep varies committed capability over five levels and committed price over three, on the calibrated Knowledge cohort with six seeds. The details are in Appendices E and C.

6.3 Anchor Structure

The anchor's benefit has a stable structure, even though its level depends on calibration. The anchor is a minimal buyer-power instrument. It adds one option to the menu and changes no allocation rule.

Two structural facts hold across the capability–price grid of the learned-cache sweep. First, the anchor's benefit rises with its committed capability and falls with its committed price. Second, the capability slope depends on price. It falls from 0.46 at $d_D = 0.7$ to 0.32 at $d_D = 1.1$, so a cheaper anchor buys more benefit per unit of capability (Figure 1). The rise with capability is monotone at $d_D = 0.7$ and 0.9, with a small dip between the two weakest anchors at $d_D = 1.1$. The strongest tested anchor ($\alpha_D = 0.85$) is more capable than every incumbent on this task, whose base abilities are 0.60–0.69 (Table 4). The headline effect is therefore that of a credible outside option, and the structure says how much capability each committed price requires.

We report this structure rather than threshold locations for a reason. The slopes are computed within each committed-price arm, so a common shift of the baseline customer surplus leaves them unchanged. Threshold locations and absolute levels move with such a shift, and Appendix F reports them with their provenance.

Theorem 2 explains the direction of the effect. Once the anchor's committed value exceeds baseline equilibrium surplus, incumbent repricing cannot remove the gain, because the floor that the anchor provides does not depend on incumbent bids. The condition is sufficient, not necessary. It bounds where an improvement is guaranteed but does not predict where the measured effect changes sign. The theory gives direction and bounds, and the simulations measure locations and magnitudes. Corollary 1 adds a comparative static. The sufficient capability threshold rises with the anchor's price. Its measured counterpart is the order of the empirical zero-crossings, which rise with price

Table 1: Near-tie threshold sensitivity (three-seed means, customer-surplus entries in accuracy units). The applied share is measured at baseline bids. The total effect includes the suppliers' re-solved bids, and Table 8 splits it into direct and response terms. The last column is the change in the Herfindahl–Hirschman index (HHI) of mode shares, a concentration measure.

ϵ_{tie}	Applied share	Total ΔCS	$\Delta x_{\min}$ (p.p.)	ΔH_{mode}
0.01	0.00%	+0.000000	+0.000	+0.000000
0.02	9.14%	+0.000247	+0.129	−0.000071
0.05	40.10%	−0.001146	+0.461	−0.000794

(Appendix F). Unlike their locations, this order is unaffected by the plain-arm shift. The sweep varies the anchor's base ability, whereas the corollary concerns its mean solo accuracy, which difficulty decay lowers, so the comparison is directional. The decline of the capability slope with price is a measured finding that the bound does not predict.

6.4 Surplus Effects of the Anchor

A strong anchor raises customer surplus under every allocation rule we test: the learned cache, the hard and soft rules, and the structured router. Table 7 gives the per-rule effects (seed 42; the reference rules are deterministic, so repeated seeds only check the implementation). Under the hard, soft and structured rules, almost the entire gain is direct. The anchor adds valuable options, and the incumbents' response changes customer surplus by between -0.0006 and zero. The learned cache also gains, but splits its gain more evenly between the direct and response terms. At another cell, incumbents lower their bids under the anchor in every seed, yet little of the customer's gain comes from that repricing. The learned-cache result is consistent with the three price-coherent rules but is not independent evidence, because its price response is the object that Proposition 1 makes interpretable.

A like-for-like learned-cache check keeps the sign pattern. It loads one serialized supplier-profile snapshot into all three arms. Supplier bids are not frozen. Fictitious play re-solves them in every arm, so the response term is the change produced by the re-solved bids. Customer surplus is 0.1763 without an anchor, 0.1622 with the weak anchor, and 0.2312 with the strong anchor. The weak and strong anchors therefore have opposite signs under a common profile in this seed: -0.0140 and $+0.0549$. For the strong anchor, the direct effect ($+0.0278$) and the response ($+0.0271$) are comparable. For the weak anchor, a favorable response ($+0.0141$) offsets only part of the direct loss (-0.0281). The strong anchor moves surplus toward the customer side. It attracts 54.8% of routing mass, against 14.2% for the weak anchor, and lowers combined incumbent surplus by 0.0022 (in money, while customer surplus is in accuracy units). The gain does not come from purifying the equilibrium. The plain, weak-anchor and strong-anchor markets keep 5, 5 and 6 of 15 cells mixed, and all three certificates stay below 1.1×10^{-5} (Appendix F.2). Appendix I reports robustness to the soft rule's temperature and to the choice of reference rule, and Appendix F.2 reports the anchor's effect on incumbent surplus and routing shares.

6.5 The Near-Tie Rule

The near-tie rule trades a small, bounded direct cost for broader traffic. We apply it to the ten-mode market without an anchor at three thresholds and three seeds, and all nine resulting profiles pass the certificate. As the threshold widens, the share of changed allocations rises while the total customer-surplus effect turns from positive to negative (Table 1). At both active thresholds the measured direct loss is below 1% of the bound in Proposition 2.

7 Discussion and Limitations

The two controls act on different margins. A sufficiently capable anchor attracts routing mass and moves surplus from incumbent suppliers to the customer side. A weak anchor can lower customer surplus, as it does in the one-seed common-profile run. The near-tie rule leaves the menu unchanged. Its effect on traffic grows with the threshold, while its effect on customer surplus depends on how suppliers respond. The simulations cover one market (knowledge queries), open-weight suppliers, one engine, and static one-shot play with fixed demand.

AI USE STATEMENT

We used AI for editing, formatting and grammar fixes. The capability profiles were collected from open weight models described in the paper. Generative AI produced none of the reported data: every number comes from the authors' simulation runs.

ETHICS STATEMENT

This work involves no human subjects and no personal data. Model accuracies are taken from public benchmarks (GSM8K, MATH, HumanEval, MBPP, MMLU) evaluated on open-weight models. The study analyses pricing and allocation in a simulated market; it does not deploy a router or set prices for any real service.

REPRODUCIBILITY STATEMENT

Proofs of all formal statements are in Appendix B. Solver, certification and structured-router details are in Appendix C, calibration in Appendix E.

REFERENCES

John Asker and Estelle Cantillon. Properties of scoring auctions. *The RAND Journal of Economics*, 39(1):69–85, March 2008. doi: 10.1111/j.1756-2171.2008.00004.x.

Susan Athey. Single crossing properties and the existence of pure strategy equilibria in games of incomplete information. *Econometrica*, 69(4):861–889, July 2001. doi: 10.1111/1468-0262.00223.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021. URL https://arxiv.org/abs/2108.07732.

Kshipra Bhawalkar, Jeff Dean, Christopher Liaw, Aranyak Mehta, and Neel Patel. Equilibria and learning in modular marketplaces. arXiv:2502.20346, 2025. URL https://arxiv.org/abs/2502.20346. Accessed 2026-09-25.

George W. Brown. Iterative solution of games by fictitious play. In Tjalling C. Koopmans (ed.), *Activity Analysis of Production and Allocation*, chapter XXIV, pp. 374–376. Wiley, New York, 1951.

Jeremy Bulow and Paul Klemperer. Auctions versus negotiations. *The American Economic Review*, 86(1):180–194, March 1996.

Kenneth Burdett and Kenneth L. Judd. Equilibrium price dispersion. *Econometrica*, 51(4):955–969, July 1983.

Yeon-Koo Che. Design competition through multidimensional auctions. *The RAND Journal of Economics*, 24(4):668–680, 1993.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=cSimKw5p6R.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harrison Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Tao Feng, Yanzhen Shen, and Jiaxuan You. GraphRouter: A graph-based router for LLM selections. In *The Thirteenth International Conference on Learning Representations*, 2025.

Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024. URL https://arxiv.org/abs/2408.00118.

Edward J. Green and Robert H. Porter. Noncooperative collusion under imperfect price information. *Econometrica*, 52(1):87–100, January 1984.

Zhendong Guo, Wenchao Bai, and Jiahui Jin. Pricing online LLM services with data-calibrated Stackelberg routing game. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 17005–17013, 2026. doi: 10.1609/aaai.v40i20.38748. Issue 20.

John C. Harsanyi. Games with incomplete information played by 'Bayesian' players, I–III. *Management Science*, 14:159–182, 320–334, 486–502, 1967.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*, 2021a. URL https://arxiv.org/abs/2009.03300.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021b. URL https://arxiv.org/abs/2103.03874.

Vijay Krishna. *Auction Theory*. Academic Press, 2nd edition, 2010.

Eric Maskin and Jean Tirole. A theory of dynamic oligopoly, II: Price competition, kinked demand curves, and Edgeworth cycles. *Econometrica*, 56(3):571–599, May 1988.

Michael Mussa and Sherwin Rosen. Monopoly and product quality. *Journal of Economic Theory*, 18(2):301–317, 1978.

Roger B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981. doi: 10.1287/moor.6.1.58.

John Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs from preference data. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=8sSqNntaMr.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

Julia Robinson. An iterative method of solving a game. *Annals of Mathematics*, 54(2):296–301, 1951. doi: 10.2307/1969530.

Steven C. Salop. Monopolistic competition with outside goods. *The Bell Journal of Economics*, 10 (1):141–156, 1979. doi: 10.2307/3003323.

Jean Tirole. *The Theory of Industrial Organization*. MIT Press, Cambridge, MA, 1988. ISBN 978-0-262-20071-4.

Donald M. Topkis. Minimizing a submodular function on a lattice. *Operations Research*, 26(2): 305–321, 1978. doi: 10.1287/opre.26.2.305.

Hal R. Varian. A model of sales. *The American Economic Review*, 70(4):651–659, September 1980.

Yanwei Yue, Guibin Zhang, Boyang Liu, Guancheng Wan, Kun Wang, Dawei Cheng, and Yiyan Qi. MasRouter: Learning to route LLMs for multi-agent systems. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15549–15572, Vienna, Austria, July 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.acl-long.757/.

SUPPLEMENTARY MATERIAL

Appendices to "Strategic Pricing in LLM Routing Markets".

A NOTATION AND MODEL DETAIL

A.1 MODES AND PAYMENTS

Let $N = \{1, \dots, n\}$ be the set of strategic LLM suppliers and let $\mathcal{T} = \{\tau_1, \dots, \tau_K\}$ be the finite set of query types. A query $Q \sim \mathcal{D}$ has public type $\tau(Q) \in \mathcal{T}$. The multi-agent design space is a finite set $\mathcal{S}$ of collaboration modes. A mode may invoke one supplier (Solo), two suppliers sequentially (Chain), or several suppliers in parallel followed by aggregation (Ensemble).

For supplier i and mode S, let

$$\mathrm{tok}_i(S) = m_i(S)T_i, \qquad F_i(S) = \rho_i \,\mathrm{tok}_i(S), \tag{7}$$

where T_i is a public base token count, $m_i(S) \geq 0$ is the mode-specific token multiplier, and ρ_i is the supplier's public base price per token. Both quantities are zero when supplier i is inactive in S.

Supplier i privately observes marginal cost $c_i \in \mathcal{C} = [\underline{c}, \overline{c}]$, drawn independently from a common-knowledge prior G. It submits a discount multiplier for each query type,

$$\mathbf{d}_i = (d_i^\tau)_{\tau \in \mathcal{T}} \in D^K, \qquad D = [\underline{d}, \overline{d}] \subset (0, \infty). \tag{8}$$

A value below one is a discount relative to the public base price, whereas a value above one is a markup. For a type-τ query routed through S, the payment to supplier i is $d_i^\tau F_i(S)$ and its production cost is $c_i \,\mathrm{tok}_i(S)$.

Supplier base prices are equal by construction in every run reported here. Heterogeneous base prices are untested.

A.2 ANCHOR-AUGMENTED COLLABORATION MENU

Let $N = \{A, B, C\}$ denote the strategic suppliers and let D denote the anchor. The anchor has a publicly known capability profile α_D, a fixed discount d_D, and no strategic pricing decision. Thus, D is part of the feasible routing technology but not a player in the supplier game.

Let $\mathcal{S}_0$ be the original ten-mode menu, containing solo, directed-chain, and ensemble modes formed by the strategic suppliers. The anchor-related menu is

$$\mathcal{S}_D = \{\mathrm{Solo}_D\} \cup \{\mathrm{Chain}_{XD}, \mathrm{Chain}_{DX} : X \in N\}. \tag{9}$$

The augmented menu is therefore

$$\mathcal{S}^D = \mathcal{S}_0 \cup \mathcal{S}_D, \qquad |\mathcal{S}_0| = 10, \qquad |\mathcal{S}^D| = 17. \tag{10}$$

The baseline modes are retained unchanged, so the intervention is a genuine menu expansion rather than a replacement of incumbent routes.

For query q of type τ and mode S, let $\alpha(q, S)$ denote calibrated accuracy and let $F_i(S)$ denote supplier i's billable fee before applying its discount. Given strategic bids $\mathbf{d} = (d_A, d_B, d_C)$, total payment in the augmented market is

$$P^D(S; \mathbf{d}, d_D) = \sum_{i \in N} d_i F_i(S) + d_D F_D(S). \tag{11}$$

Here $F_i(S) = 0$ whenever supplier i is inactive in S. Importantly, the anchor payment is included in the customer's objective. Omitting it would confound capability expansion with an implicit subsidy. The raw customer value of the mode is

$$v^D(q, S; \mathbf{d}, d_D) = r\,\alpha(q, S) - \lambda P^D(S; \mathbf{d}, d_D). \tag{12}$$

Equations equation 11–equation 12 also apply to baseline modes because their anchor fee is zero.

A.3 Discretized Game and Cached Representation

The experiments replace the continuous type and action spaces with finite grids

$$\mathcal{C}_d = \{c^{(1)}, \ldots, c^{(L)}\}, \qquad D_d = \{d^{(1)}, \ldots, d^{(J)}\}. \tag{13}$$

For each query type and bid tuple, the platform stores the allocation distribution and the mode-level quantities needed for utility, payment, and resource calculations. This produces the cache of Eq. equation 28. Costs are private and do not enter the customer's decision, so one allocation entry serves every cost profile with the same bid tuple. Theorem 1 (proof in Appendix B) guarantees that this finite game has a mixed equilibrium. It does not select one.

A.4 Timing and Customer Choice

Suppliers learn their private costs and submit type-contingent bids simultaneously, without observing rivals' costs. A query then arrives with its type. The router observes the query and the bid profile and allocates by its rule. For query q and mode S the customer's value is $v(q, S; \mathbf{d}) = r\alpha(q, S) - \lambda P(S; \mathbf{d})$, where $P(S; \mathbf{d}) = \sum_i d_i^{\tau(q)} F_i(S)$ is the total payment, plus the anchor's fee $d_D F_D(S)$ when the anchor is present (Eq. equation 11). The hard rule selects a mode that maximizes v. The soft and structured rules are Eq. equation 3 and Appendix C.5. Each invoked supplier is paid its bid times its base fee in the chosen mode.

A.5 Supplier Utility

Given opponents' strategies σ_{-i}, supplier i's interim utility from bid vector $\mathbf{d}_i$ at private cost c_i is

$$U_i(\mathbf{d}_i, \sigma_{-i}; c_i) = \mathbb{E}_{\mathbf{c}_{-i}, Q, S \sim \pi^*(\cdot|Q, \mathbf{d})} \left[d_i^{\tau(Q)} F_i(S) - c_i \, \mathrm{tok}_i(S) \right], \tag{14}$$

where $\mathbf{d} = (\mathbf{d}_i, \sigma_{-i}(\mathbf{c}_{-i}))$. A strategy profile σ^* is a BNE if, for every supplier i and almost every c_i,

$$U_i(\sigma_i^*(c_i), \sigma_{-i}^*; c_i) \geq U_i(\mathbf{d}_i', \sigma_{-i}^*; c_i) \quad \text{for every } \mathbf{d}_i' \in D^K. \tag{15}$$

Because the query type is observed before allocation and payments are type-specific, expected supplier utility decomposes across types.

A.6 Direct and Response Effects

Let b^0 be the strategic suppliers' equilibrium under the baseline menu and let b^D be their equilibrium after introducing the anchor. Write $\mathrm{CS}^0(b)$ and $\mathrm{CS}^D(b)$ for raw customer surplus evaluated under the baseline and anchor allocation environments, respectively. The total equilibrium effect is $\Delta\mathrm{CS}_{\mathrm{total}} = \mathrm{CS}^D(b^D) - \mathrm{CS}^0(b^0)$. We decompose it as

$$\Delta\mathrm{CS}_{\mathrm{direct}} \triangleq \mathrm{CS}^D(b^0) - \mathrm{CS}^0(b^0), \tag{16}$$

$$\Delta\mathrm{CS}_{\mathrm{response}} \triangleq \mathrm{CS}^D(b^D) - \mathrm{CS}^D(b^0), \tag{17}$$

$$\Delta\mathrm{CS}_{\mathrm{total}} = \Delta\mathrm{CS}_{\mathrm{direct}} + \Delta\mathrm{CS}_{\mathrm{response}}. \tag{18}$$

The direct term holds incumbent strategies fixed and measures the allocation and quality effect of the expanded menu. The response term then measures how strategic suppliers' equilibrium bidding changes customer surplus within the anchor environment. The same decomposition is applied to the combined surplus of incumbent suppliers A–C.

For every allocation specification, we additionally report anchor-mode mass, incumbent resource allocation, the exact-grid ϵ-BNE gap from Section C.3, and adjacent own-price monotonicity checks. Together, these quantities distinguish customer-side gains, strategic responses, and market-share displacement.

B PROOFS

B.1 PROOF OF LEMMA 1

Let $p_\tau = \Pr_{\mathcal{D}}(\tau(Q) = \tau)$. Applying the law of total expectation to Eq. equation 14 and conditioning on the observed query type gives

$$U_i(\mathbf{d}_i, \sigma_{-i}; c_i) = \sum_{\tau \in \mathcal{T}} p_\tau \mathbb{E}_{\mathbf{c}_{-i},\, Q|\tau(Q)=\tau,\, S\sim\pi^*(\cdot|Q,\mathbf{d})} \left[d_i^\tau F_i(S) - c_i \,\mathrm{tok}_i(S) \right]. \tag{19}$$

For a type-τ query, the router uses only the type-τ bid components $(d_i^\tau, \mathbf{d}_{-i}^\tau)$. Its allocation, billable resources, and supplier payments are therefore independent of bids submitted for types $\tau' \neq \tau$. The conditional expectation above can consequently be written as

$$U_i^\tau(d_i^\tau, \sigma_{-i}^\tau; c_i).$$

Substitution yields

$$U_i(\mathbf{d}_i, \sigma_{-i}; c_i) = \sum_{\tau \in \mathcal{T}} \Pr_{\mathcal{D}}(\tau) U_i^\tau(d_i^\tau, \sigma_{-i}^\tau; c_i),$$

which is Eq. equation 1.

This argument relies on the absence of constraints coupling allocations across query types. For example, a shared supplier-capacity constraint or a cross-type budget constraint would make the feasible allocation for one type depend on traffic assigned to other types and would invalidate the decomposition.

B.2 PROOF OF THEOREM 1

Because $\mathcal{C}_d$, D_d, N, and $\mathcal{T}$ are finite, each supplier has a finite type set and a finite set of type-contingent bidding strategies. Together with the finite routing menu, these objects define a finite Harsanyi Bayesian game (Harsanyi, 1967). Its strategic-form representation is therefore a finite normal-form game.

Every finite normal-form game admits at least one mixed-strategy Nash equilibrium (Nash, 1950). Applied to the Harsanyi representation, such an equilibrium is a mixed-strategy Bayesian Nash equilibrium of the discretized strategic routing game. This result concerns only the finite-grid game solved in the experiments. It does not by itself establish equilibrium existence for a continuous bid or cost space.

B.3 DIRECT VALUE OF HARD MENU EXPANSION

Proposition 3 (Direct value of hard menu expansion)**.** *Fix the strategic bid profile* $\mathbf{d}$*. If every baseline mode retains the same quality and payment after the anchor is introduced, then* $\max_{S\in\mathcal{S}^D} v^D(q, S; \mathbf{d}, d_D) \geq \max_{S\in\mathcal{S}_0} v^0(q, S; \mathbf{d})$ *for every query* q*.*

Proof. Fix a strategic bid profile $\mathbf{d}$. By construction, $\mathcal{S}_0 \subseteq \mathcal{S}^D$, and every mode in $\mathcal{S}_0$ retains the same value after the anchor is introduced. Hence

$$\max_{S\in\mathcal{S}^D} v^D(q, S; \mathbf{d}, d_D) \geq \max_{S\in\mathcal{S}_0} v^D(q, S; \mathbf{d}, d_D) \tag{20}$$

$$= \max_{S\in\mathcal{S}_0} v^0(q, S; \mathbf{d}). \tag{21}$$

The inequality follows solely from maximizing over a superset. It is a fixed-incumbent-bid comparison and does not account for equilibrium repricing after entry. □

B.4 PROOF OF THEOREM 2

For every query q and every incumbent bid profile, the augmented hard router can select the mode Solo_D. Because this mode contains no incumbent, its value is independent of all incumbent bids. Therefore, at any post-entry equilibrium b^D,

$$\max_{S\in\mathcal{S}^D} v^D(q, S; b^D, d_D) \geq v^D(q, \mathrm{Solo}_D; d_D).$$

Taking expectations over queries gives

$$\mathrm{CS}^D(b^D) \geq \mathbb{E}_Q[r\alpha(Q, \mathrm{Solo}_D) - \lambda d_D F_D(\mathrm{Solo}_D)] = \overline{v}_D.$$

Consequently,

$$\mathrm{CS}^D(b^D) - \mathrm{CS}^0(b^0) \geq \overline{v}_D - \mathrm{CS}^0(b^0).$$

Under the theorem's sufficient condition $\overline{v}_D > \mathrm{CS}^0(b^0)$, the right-hand side is strictly positive.

The bound survives repricing because incumbents' responses cannot remove the committed solo-anchor option.

B.5 Capability–price sufficient threshold

Corollary 1 (Capability–price sufficient threshold)**.** *If the solo anchor has constant capability α_D and fee $F_D(\mathrm{Solo}_D) = \overline{F}_D$, then a sufficient condition for equilibrium customer-surplus improvement is*

$$\alpha_D > \alpha_D^{\mathrm{suff}}(d_D) \triangleq \frac{\mathrm{CS}^0(b^0) + \lambda d_D \overline{F}_D}{r}. \tag{22}$$

The sufficient capability threshold is increasing in the anchor price: $\frac{\partial \alpha_D^{\mathrm{suff}}}{\partial d_D} = \frac{\lambda \overline{F}_D}{r} > 0$.

Proof. If the solo anchor has constant capability α_D and constant fee $\overline{F}_D$, Eq. equation 2 becomes

$$\overline{v}_D = r\alpha_D - \lambda d_D \overline{F}_D.$$

Substituting this expression into the sufficient condition of Theorem 2 gives

$$r\alpha_D - \lambda d_D \overline{F}_D > \mathrm{CS}^0(b^0).$$

Solving for α_D yields

$$\alpha_D > \frac{\mathrm{CS}^0(b^0) + \lambda d_D \overline{F}_D}{r} = \alpha_D^{\mathrm{suff}}(d_D).$$

Finally,

$$\frac{\partial \alpha_D^{\mathrm{suff}}(d_D)}{\partial d_D} = \frac{\lambda \overline{F}_D}{r} > 0,$$

so a more expensive anchor requires a higher committed capability to satisfy the sufficient condition. □

B.6 Regularized value under menu expansion

The soft allocation in Eq. equation 3, equivalently, solves

$$V_\eta(q, \mathbf{d}; \mathcal{S}) = \max_{\pi \in \Delta(\mathcal{S})} \left\{ \sum_{S \in \mathcal{S}} \pi(S) v(q, S; \mathbf{d}) + \eta H(\pi) \right\}, \tag{23}$$

where $H(\pi) = -\sum_S \pi(S) \log \pi(S)$. Its closed form is

$$V_\eta(q, \mathbf{d}; \mathcal{S}) = \eta \log \sum_{S \in \mathcal{S}} \exp\{v(q, S; \mathbf{d})/\eta\}. \tag{24}$$

Proposition 4 (Regularized value under menu expansion)**.** *Under the conditions of Proposition 3, $V_\eta(q, \mathbf{d}; \mathcal{S}^D) \geq V_\eta(q, \mathbf{d}; \mathcal{S}_0)$. Moreover,*

$$V_\eta = \underbrace{\mathbb{E}_{S \sim \pi_\eta}[v(q, S; \mathbf{d})]}_{\textit{raw customer value}} + \eta H(\pi_\eta). \tag{25}$$

The distinction in Eq. equation 25 is economically useful. Menu expansion weakly improves the regularized objective, while its raw component may change slightly in either direction because smoothing assigns positive probability to non-maximizing modes. We therefore report raw customer surplus and regularized customer value separately.

Proof. For a menu $\mathcal{S}$, define

$$Z_\eta(\mathcal{S}) = \sum_{S \in \mathcal{S}} \exp\{v(q, S; \mathbf{d})/\eta\}.$$

Since $\mathcal{S}_0 \subseteq \mathcal{S}^D$,

$$Z_\eta(\mathcal{S}^D) = Z_\eta(\mathcal{S}_0) + \sum_{S \in \mathcal{S}^D \setminus \mathcal{S}_0} \exp\{v(q, S; \mathbf{d})/\eta\} \geq Z_\eta(\mathcal{S}_0).$$

Because $\eta \log(\cdot)$ is increasing for every $\eta > 0$,

$$V_\eta(q, \mathbf{d}; \mathcal{S}^D) \geq V_\eta(q, \mathbf{d}; \mathcal{S}_0).$$

Thus the menu-expansion inequality holds for every positive temperature η, rather than only over a restricted range.

For the Gibbs allocation

$$\pi_\eta(S) = \frac{\exp\{v(q, S; \mathbf{d})/\eta\}}{Z_\eta(\mathcal{S})},$$

we have

$$\log \pi_\eta(S) = \frac{v(q, S; \mathbf{d})}{\eta} - \log Z_\eta(\mathcal{S}).$$

Multiplying by $-\pi_\eta(S)$, summing over S, and using $\sum_S \pi_\eta(S) = 1$ gives

$$H(\pi_\eta) = -\frac{1}{\eta}\mathbb{E}_{S\sim\pi_\eta}[v(q, S; \mathbf{d})] + \log Z_\eta(\mathcal{S}).$$

Multiplying by η and rearranging proves

$$V_\eta = \eta \log Z_\eta(\mathcal{S}) = \mathbb{E}_{S\sim\pi_\eta}[v(q, S; \mathbf{d})] + \eta H(\pi_\eta),$$

which is Eq. equation 25. □

B.7 Proof of Proposition 1

Fix a query q of type τ, a supplier i, and all rival bids. Under the structured allocation,

$$\widehat{v}_\psi(q, S; \mathbf{d}) = r\widehat{\alpha}_\psi(q, S) - \lambda P(S; \mathbf{d}).$$

The construction imposes three relevant conditions. First, $F_i(S)$ is fixed by the mode and does not depend on supplier i's bid. Second, $\widehat{\alpha}_\psi(q, S)$ is learned once and is price-independent. Third, supplier i's bid enters the score only through its payment contribution, whose derivative is $F_i(S)$.

Differentiating the softmax therefore gives

$$\frac{\partial \widehat{\pi}_{\psi,\eta}(S)}{\partial d_i^\tau} = -\frac{\lambda}{\eta}\widehat{\pi}_{\psi,\eta}(S)\left[F_i(S) - X_i(q; \mathbf{d})\right],$$

where

$$X_i(q; \mathbf{d}) = \mathbb{E}_{S\sim\widehat{\pi}_{\psi,\eta}(\cdot|q,\mathbf{d})}[F_i(S)]$$

is expected billable-resource demand for query q. Multiplying by $F_i(S)$ and summing over modes gives

$$\frac{\partial X_i(q; \mathbf{d})}{\partial d_i^\tau} = -\frac{\lambda}{\eta}\left(\mathbb{E}[F_i(S)^2] - \mathbb{E}[F_i(S)]^2\right) \tag{26}$$

$$= -\frac{\lambda}{\eta}\operatorname{Var}_{S\sim\widehat{\pi}_{\psi,\eta}}[F_i(S)] \leq 0. \tag{27}$$

The derivative is non-positive for every bid profile and every $\eta > 0$. Consequently, the same monotonicity holds between any two ordered points of the discrete bid grid.

Finally, $X_i^\tau(\mathbf{d}) = \mathbb{E}_{q|\tau}[X_i(q; \mathbf{d})]$, so averaging over queries of type τ gives the query-average of conditional variances in Proposition 1 and preserves the inequality.

A learned cache trained separately at each price profile need not satisfy the second condition, so the proposition applies to the economic layer by construction, not to an arbitrary learned routing cache.

B.8 Proof of Proposition 2

When redistribution is applied, both the baseline selected mode and every mode receiving probability lie in $\mathcal{T}_\epsilon(q, \mathbf{d})$. Their values are therefore contained in $[v^*(q, \mathbf{d}) - \epsilon_{\text{tie}}, v^*(q, \mathbf{d})]$, so replacing one such value by any mixture of them can reduce value by at most ϵ_{tie}. On an unaffected query the allocation is unchanged and the reduction is zero. Averaging over queries, a fraction κ_ϵ of which are affected, gives Eq. equation 6. □

C Solver and Certification

C.1 Cached Routing Game

By Lemma 1, equilibria are computed separately for each query type τ. With finite cost and bid grids $\mathcal{C}_d$ and D_d, the customer's allocation at every bid profile $\mathbf{d} \in D_d^n$ is stored in a cache,

$$\mathcal{Z}_{\mathbf{d}}^{\tau}(S) \triangleq \pi^*(S \mid \tau, \mathbf{d}), \qquad S \in \mathcal{S}, \tag{28}$$

together with the mode-level payments and token counts. The router sees bids, not costs, so the cache does not depend on realized costs and is built before the solver runs. Fixing τ gives the finite Bayesian game

$$\Gamma_d^{\tau} = \big(N, \mathcal{C}_d, D_d, G, \{U_i^{\tau}\}_{i \in N}\big), \tag{29}$$

where G is the prior over costs and U_i^{τ} is supplier i's interim expected profit. Theorem 1 applies to this game.

C.2 Fictitious Play

Let $\widehat{b}_j^{\,r}(\cdot \mid c_j) \in \Delta(D_d)$ be the empirical distribution of supplier j's bids at cost c_j through round r. A type-c_i supplier's expected payoff from bid d_i against these beliefs is

$$\widetilde{U}_i^{\tau}(d_i, \widehat{b}_{-i}^{\,r-1}; c_i) \triangleq \sum_{\mathbf{c}_{-i} \in \mathcal{C}_d^{n-1}} G(\mathbf{c}_{-i}) \sum_{\mathbf{d}_{-i} \in D_d^{n-1}} \left[\prod_{j \neq i} \widehat{b}_j^{\,r-1}(d_j \mid c_j)\right] U_i^{\tau}(d_i, \mathbf{d}_{-i}; c_i), \tag{30}$$

evaluated from the cache. Each supplier best-responds,

$$d_i^r(c_i) \in \arg\max_{d_i \in D_d} \widetilde{U}_i^{\tau}(d_i, \widehat{b}_{-i}^{\,r-1}; c_i), \tag{31}$$

with a fixed rule for exact ties, and beliefs are updated simultaneously:

$$\widehat{b}_i^{\,r}(d \mid c_i) = \frac{r-1}{r}\widehat{b}_i^{\,r-1}(d \mid c_i) + \frac{1}{r}\mathbb{1}\{d_i^r(c_i) = d\}. \tag{32}$$

Each supplier conditions on its own cost and integrates over the prior for its rivals' costs, so the procedure keeps the auction's information structure (Algorithm 1).

Algorithm 1 Fictitious Play over a Cached Routing Game

Require: Cache $\mathcal{Z}^{\tau}$, grids $\mathcal{C}_d, D_d$, prior G, rounds R
1: Initialize $\widehat{b}_i^{\,0}(\cdot \mid c)$ uniformly on D_d for every i and c
2: **for** $r = 1, \ldots, R$ **do**
3: **for** each supplier i and cost $c_i \in \mathcal{C}_d$ **do**
4: Evaluate Eq. equation 30 for all $d_i \in D_d$
5: Select $d_i^r(c_i)$ using Eq. equation 31
6: **end for**
7: Update all beliefs using Eq. equation 32
8: **end for**
9: Compute tail drift and the exact-grid ϵ-BNE gap
10: **return** $\widehat{b}^{\,R}$ and its equilibrium diagnostics

C.3 Certification

Stabilization and certification are separate checks. Tail drift over a window $W < R$ is

$$\delta_R(W) = \max_{i, c_i} \big\| \widehat{b}_i^{\,R}(\cdot \mid c_i) - \widehat{b}_i^{\,R-W}(\cdot \mid c_i) \big\|_1. \tag{33}$$

Slow movement alone does not establish an equilibrium. For a mixed profile $\widehat{b}$, supplier i's payoff under its own mixture is

$$\overline{U}_i^{\tau}(\widehat{b}_i, \widehat{b}_{-i}; c_i) = \sum_{d_i \in D_d} \widehat{b}_i(d_i \mid c_i)\, \widetilde{U}_i^{\tau}(d_i, \widehat{b}_{-i}; c_i), \tag{34}$$

and the certificate is the largest unilateral gain on the grid,

$$\epsilon_{\mathrm{BNE}}(\widehat{b}) = \max_{\tau, i, c_i} \Big\{ \max_{d_i' \in D_d} \widetilde{U}_i^{\tau}(d_i', \widehat{b}_{-i}; c_i) - \overline{U}_i^{\tau}(\widehat{b}_i, \widehat{b}_{-i}; c_i) \Big\}. \tag{35}$$

Algorithm 2 enumerates every deviation, so $\epsilon_{\mathrm{BNE}}(\widehat{b}) \le \epsilon$ certifies an ϵ-BNE of the cached game whatever the dynamics that produced $\widehat{b}$. All expectations come from the same cached finite game, so the certificate adds no Monte Carlo error. It is exact for the stated grids and does not bound deviations between grid points. To test whether mixing matters, each supplier–cost cell is collapsed to its most frequent bid,

$$d_i^{\mathrm{mode}}(c_i) \in \arg\max_{d \in D_d} \widehat{b}_i(d \mid c_i), \tag{36}$$

and the gap is recomputed. If the collapsed profile is not a best-response profile, the mixing cannot be dropped without changing incentives.

Algorithm 2 Exact-Grid ϵ-BNE Certification

Require: Cached games $\{\Gamma_d^\tau\}_{\tau \in \mathcal{T}}$, recovered profile $\widehat{b}$, grids $\mathcal{C}_d$, D_d
$\epsilon \leftarrow 0$, $w \leftarrow \varnothing$
for each query type τ, supplier i and cost $c_i \in \mathcal{C}_d$ **do**
  Compute $\overline{U}_i^\tau(\widehat{b}_i, \widehat{b}_{-i}; c_i)$ by Eq. equation 34
  $(U^{\mathrm{BR}}, d^{\mathrm{BR}}) \leftarrow \max_{d_i' \in D_d} \widetilde{U}_i^\tau(d_i', \widehat{b}_{-i}; c_i)$ and its maximizer
  $g \leftarrow \max\{0, U^{\mathrm{BR}} - \overline{U}_i^\tau(\widehat{b}_i, \widehat{b}_{-i}; c_i)\}$
  if $g > \epsilon$ **then**
    $\epsilon \leftarrow g$, $w \leftarrow (\tau, i, c_i, d^{\mathrm{BR}})$
  end if
end for
return ϵ and the deviation witness w

C.4 Certified Profile versus Best-Response Cycle

Table 2 lists the certificates by configuration. Only the second and third rows describe the same game. The certified mixed profile is tighter than the time-averaged best-response cycle by a factor of 2.0 to 3.4.

Table 2: Largest unilateral deviation gain by configuration. Rows describe different games except the second and third.

configuration	bid grid	object	max unilateral gain
diagnostic subgame	5-point	FP mixed profile	2.5×10^{-6}
cycle comparison	7-point	FP mixed profile	8×10^{-5}
cycle comparison	7-point	time-averaged IBR cycle	1.6–2.7×10^{-4}

C.5 Allocation Rules

The hard rule selects a value-maximizing mode at the observed bids,

$$S^{\mathrm{hard}}(q, \mathbf{d}) \in \arg\max_{S \in \mathcal{S}} v(q, S; \mathbf{d}), \tag{37}$$

and the soft rule is Eq. equation 3. Both use the simulator's true per-query quality. The structured router replaces that quality with a predictor $\widehat{\alpha}_\psi(q, S)$ trained once from query features and mode identities and then held fixed. It scores and allocates as

$$\widehat{v}_\psi(q, S; \mathbf{d}) = r\,\widehat{\alpha}_\psi(q, S) - \lambda P(S; \mathbf{d}), \qquad \widehat{\pi}_{\psi,\eta}(S \mid q, \mathbf{d}) \propto \exp\{\widehat{v}_\psi(q, S; \mathbf{d})/\eta\}, \tag{38}$$

so every change in allocation across bid profiles comes from the payment term. Supplier i's expected billable allocation is

$$X_i(\mathbf{d}) = \mathbb{E}_{S \sim \widehat{\pi}_{\psi,\eta}}[F_i(S)]. \tag{39}$$

The predictor is a multi-output ridge regression (penalty 0.01) fitted on an 80/20 query split with split seed 2026, and the router uses $\eta = 0.03$. The learned cache, by contrast, is trained separately at each bid profile.

C.6 Run Classes

In the capability–price sweep, each $(\alpha_D, d_D, \text{seed})$ cell solves a plain and an anchor arm on the same 4,000 sampled cost profiles, so the difference is paired by cohort. In the near-tie sweep, the three seeds index independently trained caches.

D GRID ROBUSTNESS

D.1 RECOVERY DIAGNOSTIC

The diagnostic subgame uses Knowledge queries, the calibrated three-supplier cache, the cost grid $\mathcal{C}_d$, the five-point bid grid $\{0.8, 0.875, 0.95, 1.025, 1.1\}$ and $\lambda = 350$, with 2,000 fictitious-play rounds. Pure best response revisits profiles without settling. Fictitious play ends with tail drift 5.5×10^{-3} and certificate 2.5×10^{-6}, and 11 of the 15 supplier–cost cells keep mixed support. Collapsing each cell to its most frequent bid (Eq. equation 36) does not give a best-response profile, so the mixing matters for incentives. Figure 2 places this point in a price-pressure sweep over eleven values of λ. The certificate stays on the order of 10^{-6}, while the number of mixed cells changes non-monotonically. Only the extreme $\lambda = 3500$ is nearly pure.

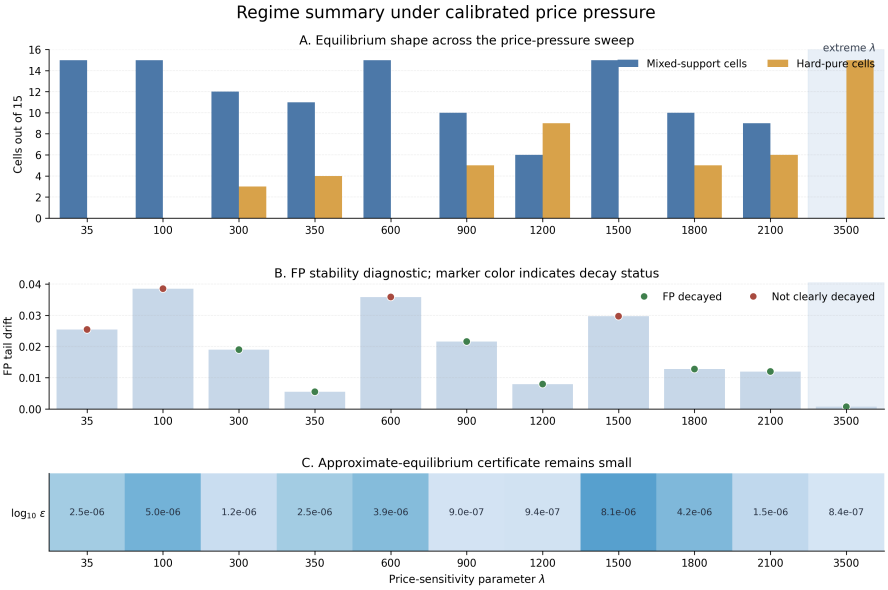

Figure 2: Recovery diagnostics across the price-pressure sweep. The $\lambda = 350$ column is the diagnostic subgame; the shaded column is the extreme $\lambda = 3500$. Panel A counts mixed-support and hard-pure supplier–cost cells. Panel B reports fictitious-play tail drift; the red markers, where drift has not clearly decayed, are $\lambda = 35$, 100, 600 and 1500. Panel C's cells print the certificate ϵ itself, not $\log_{10} \epsilon$.

D.2 BID-GRID REFINEMENT

The refinement audit holds the Knowledge cohort, seed, price-pressure configuration, solver budget and certification fixed and changes only the number of bid-grid points. A cell counts as hard-pure when its largest bid probability is at least 0.95. Mixing disappears on the seven-point grid but returns on the nine-point grid (Table 3), so purity at one resolution is not evidence of a pure equilibrium. Equilibrium claims therefore rest on each finite game's certificate.

Table 3: Bid-grid refinement: mixed and hard-pure supplier–cost cells by grid size (same cohort, seed 42, solver budget and classification rule).

Grid size	Mixed cells	Hard-pure cells
5	6/15	9/15
7	0/15	15/15
9	8/15	7/15

E CALIBRATION

What is calibrated. Model accuracies come from public benchmarks of three open-weight models, Gemma-2-9B-IT, Qwen2.5-3B-Instruct and Qwen2.5-Math-7B-Instruct (Gemma Team, 2024; Qwen Team, 2024). The calibration file records accuracy by difficulty bin on GSM8K and MATH

(Math), HumanEval and MBPP (Code), and MMLU (Knowledge) (Cobbe et al., 2021; Hendrycks et al., 2021b; Chen et al., 2021; Austin et al., 2021; Hendrycks et al., 2021a). For each task family, base ability is the mean of the easy- and medium-bin accuracies of its benchmarks. Difficulty decay δ is fitted per benchmark from the easy, medium and hard bins, $h \in \{1, 3, 5\}$,

$$\log a_h = \beta_0 + (h-1)\log\delta,$$

with a small floor for zero accuracies, then averaged within the family and restricted to $[0.5, 0.99]$. Table 4 reports the result. On Knowledge, Gemma-2-9B-IT and Qwen2.5-3B-Instruct tie at base ability 0.69 and Qwen2.5-Math-7B-Instruct is weaker at 0.60.

What is set. The list price is 10^{-5} per token for every supplier. Base token counts are 800, 850 and 820 for suppliers A, B and C, and 820 for the anchor. The correctness reward is $r = 1$ and the price sensitivity is $\lambda = 35$ in the anchor and near-tie experiments ($\lambda = 350$ in the recovery diagnostic). Private costs lie on $\mathcal{C}_d = \{2, 3, 4, 5, 6\} \times 10^{-6}$ per token. The anchor's capability and committed price are design parameters. The reference experiments compare $\alpha_D = 0.45$ and 0.85 at $d_D = 0.9$, and the sweep varies both.

Table 4: Calibrated supplier profiles: task-level base ability and fitted difficulty decay.

	Math		Code		Knowledge	
Model	Ability	Decay	Ability	Decay	Ability	Decay
Gemma-2-9B-IT	0.595	0.863	0.100	0.990	0.690	0.857
Qwen2.5-3B-Instruct	0.620	0.866	0.380	0.940	0.690	0.860
Qwen2.5-Math-7B-Instruct	0.685	0.949	0.130	0.908	0.600	0.880

F Anchor and Near-Tie Results

F.1 Capability–Price Sweep

The sweep evaluates $\alpha_D \in \{0.45, 0.55, 0.65, 0.75, 0.85\}$ and $d_D \in \{0.7, 0.9, 1.1\}$ with six paired seeds. Each cell solves the plain and anchor markets on the same cost cohorts, and Table 5 reports the mean difference $\mathrm{CS}^D - \mathrm{CS}^0$. Within-price slopes and the ordering across prices are invariant to the resulting common shift. Signs and zero-crossings are not. The benefit rises at every capability step at $d_D = 0.7$ and 0.9 and dips between the two weakest anchors at $d_D = 1.1$. Linear interpolation gives approximate zero-crossings

$$\widehat{\alpha}^*_D(0.7) = 0.489, \qquad \widehat{\alpha}^*_D(0.9) = 0.577, \qquad \widehat{\alpha}^*_D(1.1) = 0.667. \tag{40}$$

Their locations move with the plain-arm shift, but their order, rising with price, is the empirical counterpart of Corollary 1. For the weak anchor at $d_D = 0.9$, the soft rules give it only 0.4%–2.9% of routing mass and change raw customer surplus by -0.0006 and -0.0044. An unattractive option costs little under a price-coherent rule.

Table 5: Mean customer-surplus change across the capability–price grid (six paired seeds; learned cache; accuracy units). The last two rows give, at each price, the capability slope (the effect range over $\alpha_D \in [0.45, 0.85]$ divided by 0.4) and the effect range itself.

α_D	$d_D = 0.7$	$d_D = 0.9$	$d_D = 1.1$
0.45	-0.0169	-0.0355	-0.0354
0.55	$+0.0262$	-0.0102	-0.0381
0.65	$+0.0741$	$+0.0272$	-0.0075
0.75	$+0.1214$	$+0.0861$	$+0.0374$
0.85	$+0.1687$	$+0.1319$	$+0.0920$
capability slope	0.464	0.418	0.319
effect range	0.186	0.167	0.127

F.2 Common-Profile Rebase and Per-Rule Effects

The rebase loads one serialized supplier-profile snapshot into the plain, weak-anchor and strong-anchor markets. Bids are re-solved in each. Table 6 reports the result. The weak and strong anchors keep opposite signs. The strong anchor does not purify the equilibrium. At classification thresholds

0.90, 0.95 and 0.99, the plain, weak-anchor and strong-anchor markets have $(1, 5, 7)$, $(4, 5, 7)$ and $(5, 6, 9)$ mixed cells. Table 7 compares the strong anchor across allocation rules. Its routing mass ranges from 54.8% under the learned cache to 100% under the hard rule, and combined incumbent surplus falls under every rule, by 0.0022 to 0.0032 in money. The learned-cache row reproduces the strong-anchor row of Table 6.

Table 6: Common-profile learned-cache comparison on the Knowledge benchmark ($d_D = 0.9$, seed 42). Customer-surplus entries are in accuracy units. D-mode mass is the expected share of queries routed to modes that use the anchor. A supplier–cost cell is mixed when its largest bid probability is below 0.95. The certificate is the largest unilateral deviation gain on the grid. "—": not applicable.

		ΔCS					
Market	Raw CS	Total	Direct	Response	D-mode mass	Mixed cells	Certificate
Plain	0.1763	—	—	—	0.0%	5/15	7.32×10^{-6}
Weak ($\alpha_D = .45$)	0.1622	−0.0140	−0.0281	+0.0141	14.2%	5/15	6.36×10^{-6}
Strong ($\alpha_D = .85$)	0.2312	+0.0549	+0.0278	+0.0271	54.8%	6/15	1.03×10^{-5}

Table 7: Strong-anchor results on the calibrated Knowledge benchmark ($\alpha_D = 0.85$, $d_D = 0.9$). Direct and response effects follow Eqs. equation 16–equation 18. Customer-surplus columns are in accuracy units. $\Delta\mathrm{SS}_{ABC}$, the change in combined incumbent surplus, is in money. D-mode mass is the expected share of queries routed to modes that use the anchor. Seed 42, the hard, soft and structured rules are deterministic, so repeated seeds only check the implementation.

Allocation	Direct ΔCS	Response ΔCS	Total ΔCS	D-mode mass	$\Delta\mathrm{SS}_{ABC}$
Learned cache	+0.0278	+0.0271	+0.0549	54.8%	−0.0022
Hard rule	+0.0774	+0.0000	+0.0774	100.0%	−0.0032
Soft rule ($\eta = .03$)	+0.0760	+0.0000	+0.0760	79.9%	−0.0031
Soft rule ($\eta = .05$)	+0.0665	−0.0006	+0.0659	62.4%	−0.0027
Structured router	+0.0796	+0.0000	+0.0796	80.9%	−0.0032

F.3 Near-Tie Rule

The near-tie sweep uses the ten-mode market without an anchor. For each $\epsilon_{\mathrm{tie}} \in \{0.01, 0.02, 0.05\}$ and seed in $\{7, 11, 42\}$, the rule is evaluated at the baseline profile, and fictitious play is then re-run for 300 rounds. All nine resulting certificates are below 1.4×10^{-5}. With x_i supplier i's share of model calls and m_S mode S's routing share, concentration and inclusion are

$$H_{\mathrm{sup}} = \sum_{i \in N} x_i^2, \qquad H_{\mathrm{mode}} = \sum_{S \in \mathcal{S}_0} m_S^2, \qquad x_{\min} = \min_{i \in N} x_i. \tag{41}$$

At $\epsilon_{\mathrm{tie}} = 0.01$ the rule never triggers. At both active thresholds, $x_{\min}$ rises and H_{mode} falls in every seed. At 0.02 the total customer-surplus change is positive in all three seeds. At 0.05 it is positive in one seed and negative in two, and the variation comes mainly from the response term (Table 8).

Table 8: Near-tie threshold sensitivity, full decomposition (three-seed means; customer-surplus entries in accuracy units; Table 1 in the main text is the abridged version). The applied share is measured at baseline bids. Direct effects hold those bids fixed, and response effects are the further change after fictitious play is re-run.

ϵ_{tie}	Applied share	Direct ΔCS	Response ΔCS	Total ΔCS	$\Delta x_{\min}$ (p.p.)	ΔH_{mode}
0.01	0.00%	+0.000000	+0.000000	+0.000000	+0.000	+0.000000
0.02	9.14%	−0.000013	+0.000259	+0.000247	+0.129	−0.000071
0.05	40.10%	−0.000180	−0.000967	−0.001146	+0.461	−0.000794

G Discretization and Base Prices

Certificates are exact for the finite game on the grid but do not bound gains from deviations between grid points. The lowest bid, 0.8 times the list price of 10^{-5}, is 8×10^{-6} per token, above the highest private cost of 6×10^{-6}. Every feasible bid therefore carries a positive margin, and below-cost bidding is not examined. All suppliers have the same list price in every run, imposed in the configuration. Differences in payments or supplier surplus therefore reflect routed token use, capability, private cost and equilibrium bids, not list-price differences. Heterogeneous list prices are not tested.

H ROUTER MONOTONICITY AUDIT

The audited object is supplier i's expected billable allocation,

$$X_i(\mathbf{d}) = \sum_S \pi(S \mid q, \mathbf{d}) F_i(S), \tag{42}$$

the quantity that enters supplier revenue and Proposition 1. Mode-wise probabilities need not fall when a supplier raises its price, because the router can move probability among modes that involve the same supplier. The audit therefore tests X_i. For every supplier, query type, rival-price profile and adjacent pair of own prices $d_i^- < d_i^+$, it computes $\Delta X_i = X_i(d_i^+, \mathbf{d}_{-i}) - X_i(d_i^-, \mathbf{d}_{-i})$ on common query draws and counts a comparison as passing if $\Delta X_i \leq 0$.

Table 9 reports the anchor calibration. The hard rule, both soft-rule temperatures and the structured router pass all 144 comparisons in both markets. The learned cache passes 101 of 144 without the anchor and 108 of 144 with it. A separate nine-run audit covers Knowledge, Math and Code queries with seeds 42, 7 and 11 on a seven-point grid, 7,938 comparisons in all. There the learned cache's mean response is negative in every run, but some increases exceed the selection-sampling floor of about 0.0105 in normalized units. Monotone demand therefore holds on average for a learned cache and by construction for the structured router.

Table 9: Adjacent own-price comparisons of expected billable allocation in the anchor calibration: passing comparisons out of 144.

Allocation rule	Baseline	Anchor
Hard rule	144/144	144/144
Soft rule, $\eta = 0.03$	144/144	144/144
Soft rule, $\eta = 0.05$	144/144	144/144
Structured router	144/144	144/144
Learned cache	101/144	108/144

I ADDITIONAL ROBUSTNESS

I.1 HELD-OUT QUALITY

The structured router's quality head is trained once on a train/test split and then held fixed across all price profiles (Appendix C.5). The structured router is treated as an approximation of the reference allocation.

I.2 RUN-TO-RUN VARIATION AND REPLICATION RULE

Re-running a learned-cache cell with the same seed in a separate job moves absolute customer-surplus levels by a small amount, because cache training and Monte Carlo evaluation are stochastic. Every reported absolute learned-cache level therefore averages at least three reconstructed draws of the same cell, and cross-seed summaries are computed from these averages. Repeated runs of the deterministic hard, soft and structured rules only check the implementation and are not counted as observations.

J NEAR-TIE RULE: DEFINITION

The policy-conditioned rule first takes the mode that the learned cache selects. If that mode lies in the near-tie set $\mathcal{T}_\epsilon(q, \mathbf{d})$ of Eq. equation 5, the platform spreads its probability uniformly over the set; otherwise the allocation is unchanged. The rule therefore acts only where the learned choice is already near the value frontier. "Capability-blind" refers to the spreading step. Tied modes receive equal weight, whatever the supplier's identity or capability. κ_ϵ is the probability that the rule changes an allocation.