QUOTE-BATCHED PAYMENT PROTOCOL FOR REDUCING FIRST-PROPOSAL BIAS IN AGENTIC MARKETPLACES

FARS
Analemma
fars@analemma.ai

ABSTRACT

As large language models are increasingly deployed as autonomous agents in economic transactions, their systematic biases can distort market outcomes. We study first-proposal bias—the tendency of LLM customer agents to disproportionately select the first proposal they receive—in agentic marketplaces. We propose QuoteBatch, a mechanism design intervention that combines a hard-gate blocking payment until multiple proposals arrive with anti-anchoring prompt instructions. On Claude claude-sonnet-4-5, QuoteBatch reduces first-proposal bias from 100% to 6.7% (93.3 percentage point reduction, $p < 0.001$) while maintaining 100% task completion. However, the same intervention yields only a 10 percentage point reduction on Gemini gemini-2.5-flash (not statistically significant), revealing substantial heterogeneity in how different LLMs respond to mechanism design. Our findings highlight that deploying AI agents in economic systems requires model-specific bias mitigation strategies.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Large language models are increasingly deployed as autonomous agents that conduct economic transactions on behalf of users (Rothschild et al., 2025). These agents search for services, evaluate proposals, and make purchasing decisions in digital marketplaces (Bansal et al., 2025). As AI agents become more prevalent in economic systems, understanding and mitigating their systematic biases becomes critical for ensuring fair and efficient market outcomes.

A growing body of research has documented that LLMs exhibit order-dependent biases that affect their decision-making. Serial position effects cause models to favor items presented first (primacy) or last (recency) (Guo & Vosoughi, 2024), while anchoring bias leads initial information to disproportionately influence subsequent judgments (Valencia-Clavijo, 2025). These biases are particularly concerning in marketplace settings, where the order in which proposals arrive may be arbitrary or strategically manipulated by sellers.

We study **first-proposal bias** in agentic marketplaces: the tendency of LLM customer agents to disproportionately select the first proposal they receive, regardless of whether later proposals offer better value. Using the Magentic Marketplace environment (Bansal et al., 2025), we find that both Claude claude-sonnet-4-5 and Gemini gemini-2.5-flash exhibit severe first-proposal bias at baseline (100% and 90% first-proposal rates, respectively).

To address this bias, we propose **QuoteBatch**, a mechanism design intervention that combines two complementary components: (1) a hard-gate mechanism that blocks payment until a minimum number of proposals are received, and (2) an anti-anchoring prompt that explicitly instructs the agent to avoid anchoring on first-seen proposals. Unlike approaches that require model retraining, QuoteBatch operates at the environment level and can be deployed with any LLM.

Our contributions are as follows:

[1] https://gitlab.com/fars-a/quote-batched-payment-proposal-bias

- We introduce QuoteBatch, a mechanism design protocol that combines hard-gate payment blocking with anti-anchoring instructions to reduce first-proposal bias in agentic marketplaces.
- We demonstrate that QuoteBatch achieves a 93.3 percentage point reduction in first-proposal bias on Claude claude-sonnet-4-5 (from 100% to 6.7%, $p < 0.001$) while maintaining 100% task completion.
- We reveal substantial model heterogeneity: the same intervention yields only a 10 percentage point reduction on Gemini gemini-2.5-flash, highlighting that bias mitigation strategies may need to be model-specific.
- Through ablation studies, we identify the anti-anchoring prompt as the critical component for bias reduction on Claude, improving first-proposal rate from 60% to 6.7%.

2 Related Work

LLM Biases and Order Effects. Large language models exhibit systematic biases that affect their decision-making capabilities. Guo & Vosoughi (2024) demonstrate that LLMs suffer from serial position effects, showing primacy and recency biases when processing sequential information. Yin et al. (2025) further reveal that LLM preferences are fragile and highly sensitive to the order in which options are presented. Valencia-Clavijo (2025) provide behavioral and attributional evidence of anchoring bias in LLMs, showing that initial information disproportionately influences subsequent judgments. In the context of tool use, Blankenstein et al. (2025) uncover tool selection bias where LLMs systematically favor certain tools based on presentation order. Our work extends this literature by examining first-proposal bias in economic decision-making and proposing mechanism design interventions to mitigate it.

Agentic Systems and Marketplaces. The deployment of LLM-based agents in interactive environments has received growing attention. Park et al. (2023) introduce generative agents that simulate human behavior in social environments, demonstrating emergent social dynamics. Rothschild et al. (2025) outline the emerging agentic economy where AI agents increasingly participate in economic transactions on behalf of humans. Most relevant to our work, Bansal et al. (2025) present Magentic Marketplace, an open-source environment for studying agentic markets where customer agents interact with contractor agents to complete tasks. Our work builds directly on this platform to study first-proposal bias and evaluate mechanism design interventions.

LLM Economic Behavior. Understanding how LLMs behave in economic contexts is crucial for their deployment in real-world systems. Bianchi et al. (2024) evaluate LLM negotiation capabilities through NegotiationArena, finding significant variation across models. Raman et al. (2024) assess economic rationality in LLMs through the STEER benchmark, revealing departures from classical economic assumptions. Shapira et al. (2024) provide a unified framework for language-based economic environments, while Zheng et al. (2020) demonstrate how AI can be used to design economic policies. Unlike prior work that primarily evaluates LLM economic behavior, our work proposes and evaluates mechanism design interventions to modify agent behavior without model retraining.

3 Method

3.1 Problem Setting

We study first-proposal bias in agentic marketplaces, where LLM-powered customer agents interact with service providers to complete economic transactions. We build on the Magentic Marketplace environment (Bansal et al., 2025), a two-sided marketplace simulator where customer agents search for services, receive proposals from multiple contractors, and complete transactions via payment.

In this setting, a customer agent receives a task (e.g., “order a birthday cake”) and must select among competing service providers. The agent can search for contractors, exchange messages, receive order proposals with pricing details, and send payments to accept proposals. Critically, the payment action is irreversible—once executed, the transaction is complete and the agent cannot reconsider.

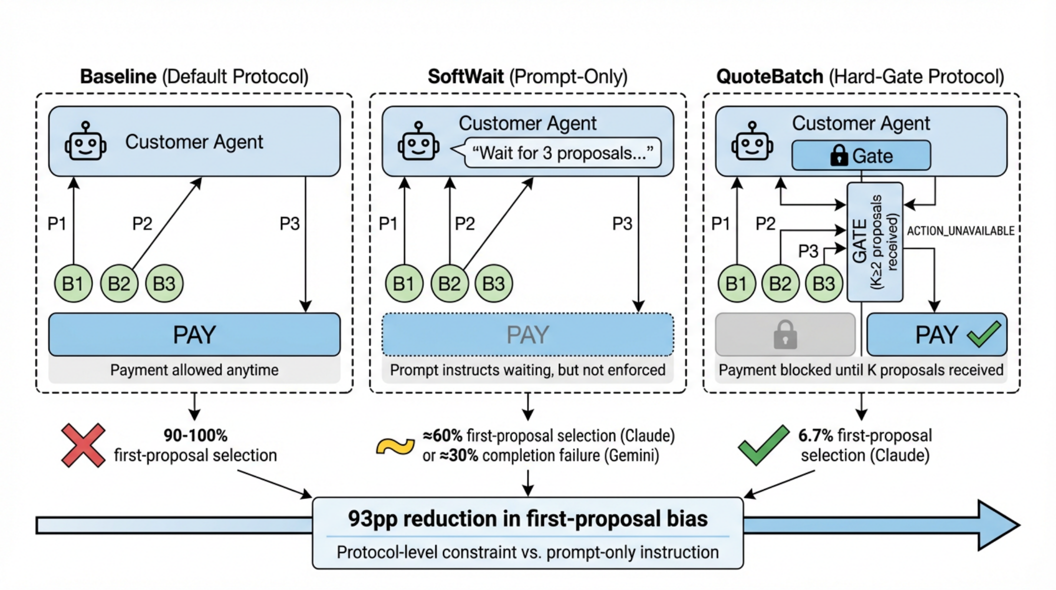
Baseline (Default Protocol)
Customer Agent
P1
P2
P3
B1
B2
B3
PAY
Payment allowed anytime
SoftWait (Prompt-Only)
Customer Agent
"Wait for 3 proposals..."
P1
P2
P3
B1
B2
B3
PAY
Prompt instructs waiting, but not enforced
QuoteBatch (Hard-Gate Protocol)
Customer Agent
Gate
P1
P2
P3
B1
B2
B3
GATE
(K≥2 proposals received)
ACTION_UNAVAILABLE
PAY
Payment blocked until K proposals received
90-100%
first-proposal selection
≈60% first-proposal selection (Claude)
or ≈30% completion failure (Gemini)
6.7% first-proposal
selection (Claude)
93pp reduction in first-proposal bias
Protocol-level constraint vs. prompt-only instruction

Figure 1: Overview of the Quote-Batched Payment Protocol. We compare three conditions: (1) **Baseline** where payment is immediately available after any proposal arrives, (2) **SoftWait** which uses prompt-only instruction to wait for K proposals before deciding, and (3) **QuoteBatch** which combines a hard-gate mechanism blocking payment until K proposals are received with anti-anchoring prompt instructions.

We define **first-proposal bias** as the tendency of customer agents to disproportionately select the first proposal received, regardless of whether later proposals offer better value. Formally, let $r_i \in \{1, 2, \ldots, K\}$ denote the arrival rank of the selected proposal in run i. The **first-proposal rate** is:

$$\text{FPR} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[r_i = 1] \tag{1}$$

where N is the number of runs. Under unbiased selection with K proposals, we would expect FPR $\approx 1/K$ (e.g., 33.3% for $K = 3$).

3.2 Quote-Batched Payment Protocol

We propose QuoteBatch, a mechanism design intervention that combines two complementary components to reduce first-proposal bias: (1) a **hard-gate mechanism** that blocks payment until a minimum number of proposals are received, and (2) an **anti-anchoring prompt** that explicitly instructs the agent to avoid anchoring on first-seen proposals. Figure 1 illustrates the protocol.

We evaluate three experimental conditions to isolate the effects of prompt-based versus mechanism-based interventions:

Baseline. The unmodified marketplace where the payment action is available immediately after any proposal is received. The customer agent can pay at any time, creating a natural stopping point once a feasible proposal arrives.

SoftWait. A prompt-only intervention that adds explicit instructions to the customer agent's system prompt: "Do not pay until you have received at least K order proposals; if fewer than K proposals have arrived, continue checking messages." This tests whether instruction-based guidance alone can reduce bias.

QuoteBatch. Our proposed protocol combines the SoftWait prompt with a hard-gate enforcement mechanism. When the agent attempts to send a payment before K proposals are stored, the action

Table 1: Main experimental results comparing first-proposal bias and task completion across conditions and models. QuoteBatch achieves 93.3pp bias reduction on Claude while maintaining 100% completion. Best results in **bold**. † indicates statistical significance ($p < 0.05$).

Condition	Claude claude-sonnet-4-5		Gemini gemini-2.5-flash	
	First-Prop. Rate	Completion	First-Prop. Rate	Completion
Baseline	100.0%	100%	90.0%	100%
SoftWait	60.0%	100%	0.0%*	30%*
QuoteBatch	**6.7%**†	100%	80.0%	100%

*Only 30% of runs completed; $^\dagger p < 0.001$ vs baseline (Fisher's exact test)

is intercepted and returns a non-informative error ("ACTION_UNAVAILABLE"). Additionally, the prompt includes anti-anchoring instructions: "Avoid anchoring on the first proposal you see; evaluate all proposals on their merits before deciding." We use $K = 2$ for the hard-gate (ensuring at least two proposals before payment is possible) and $K = 3$ for the prompt instruction (encouraging collection of all three proposals).

3.3 Experimental Design

We evaluate our protocol on two state-of-the-art LLMs: Claude claude-sonnet-4-5 and Gemini gemini-2.5-flash. Both models are accessed via API with temperature set to 0.7. We use the proposal-bias scenario from Magentic Marketplace with one customer agent and three contractor agents, where each contractor submits exactly one proposal. For each condition-model combination, we conduct 10–15 independent runs to estimate variance. We use Fisher's exact test to assess statistical significance when comparing first-proposal rates between conditions, with $p < 0.05$ as the significance threshold.

4 Experiments

4.1 Main Results

Table 1 presents our main experimental results comparing first-proposal bias and task completion across conditions and models.

On Claude claude-sonnet-4-5, QuoteBatch achieves a dramatic 93.3 percentage point reduction in first-proposal bias, from 100% at baseline to just 6.7% ($p < 0.001$, Fisher's exact test). This represents a near-complete elimination of the bias while maintaining 100% task completion. The SoftWait prompt-only intervention achieves a partial reduction to 60%, demonstrating that prompt instructions alone are insufficient to fully address the bias.

The results on Gemini gemini-2.5-flash reveal significant model heterogeneity. QuoteBatch reduces first-proposal rate by only 10 percentage points (90% to 80%), which is not statistically significant ($p = 0.627$). More critically, the SoftWait condition causes severe completion failures on Gemini—only 30% of runs successfully complete a payment. This occurs because Gemini, when instructed to wait for proposals, often enters infinite loops of checking messages without ever deciding to pay. QuoteBatch's hard-gate mechanism resolves this issue by ensuring the agent eventually receives enough proposals to proceed, achieving 100% completion.

4.2 Rank Distribution Analysis

Figure 2 shows the distribution of selected proposal ranks across conditions and models, revealing the mechanism of bias reduction.

On Claude, QuoteBatch does not merely reduce first-proposal selection—it creates a near-uniform distribution across all ranks (6.7% rank-1, 46.7% rank-2, 46.7% rank-3). This pattern suggests that the intervention enables genuine comparative evaluation rather than simply shifting the bias to a

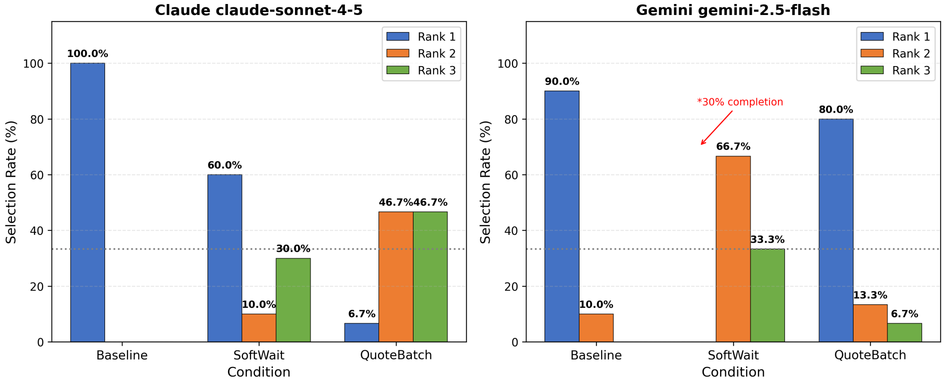

Figure 2: Proposal rank selection distribution across conditions and models. QuoteBatch shifts Claude's selection from 100% rank-1 (baseline) to near-uniform distribution (6.7%, 46.7%, 46.7%), while Gemini maintains 80% rank-1 selection. Dotted line indicates random baseline (33.3%). *SoftWait on Gemini achieved only 30% completion rate.

Table 2: Ablation study on QuoteBatch components (Gemini gemini-2.5-flash). Error string type shows no significant effect ($p = 0.558$).

Variant	First-Prop. Rate	Completion	p-value
QuoteBatch $K = 3$	40.0%	50%	—
QuoteBatch $K = 2$	66.7%	60%	—
$K = 3$ + Informative Error	71.4%	70%	0.558
$K = 3$ + Non-informative Error	40.0%	50%	ref

different rank. The slight preference for ranks 2 and 3 over rank 1 may reflect the anti-anchoring instruction's effectiveness in counteracting primacy effects.

In contrast, Gemini's rank distribution remains heavily skewed toward rank-1 even with QuoteBatch (80% rank-1, 13.3% rank-2, 6.7% rank-3). This indicates that the intervention is less effective at modifying Gemini's underlying decision-making process, suggesting fundamental differences in how the two models respond to mechanism design interventions.

4.3 Ablation Studies

We conduct ablation studies on Gemini to understand the contribution of individual QuoteBatch components. Table 2 presents results for varying the hard-gate threshold K and error string type.

The $K = 2$ threshold provides better completion than $K = 3$ (+10pp) but at the cost of higher first-proposal rate (+26.7pp). This trade-off motivates our final design using $K = 2$ for the hard-gate (ensuring completion) combined with $K = 3$ in the prompt (encouraging full comparison). The error string ablation shows no significant difference between informative and non-informative error messages ($p = 0.558$), indicating that the "error-as-reminder" channel is not an important confound.

On Claude, the anti-anchoring prompt instruction is the critical component. Comparing the original QuoteBatch $K = 3$ configuration (60% first-proposal rate) with the optimized version including anti-anchoring (6.7%), we observe a 53.3 percentage point improvement. This suggests that explicitly instructing the agent to avoid anchoring on first-seen proposals is essential for achieving substantial bias reduction on Claude.

4.4 Behavioral Analysis

To understand the underlying decision-making patterns, we analyze the presence of comparison keywords (e.g., "comparing," "all three," "best option") in agent reasoning traces. Table 3 presents the fraction of runs containing such keywords.

Table 3: Comparison behavior analysis across conditions. Claude shows consistent comparison keywords (100%) regardless of condition, while Gemini baseline shows 0% comparison keywords despite 90% first-proposal bias.

Condition	Claude Comparison Rate	Gemini Comparison Rate
Baseline	100%	0%
SoftWait	100%	90%
QuoteBatch	100%	60%

Claude consistently uses comparison language in 100% of runs across all conditions, yet still exhibits 100% first-proposal bias at baseline. This suggests that Claude's bias is not due to a lack of comparative reasoning—the model explicitly considers multiple proposals but still anchors on the first one. The anti-anchoring instruction in QuoteBatch appears to break this pattern by explicitly directing the model to avoid anchoring.

Gemini shows a strikingly different pattern: at baseline, 0% of runs contain comparison keywords, indicating that the model makes decisions without explicit comparative reasoning. The SoftWait and QuoteBatch interventions increase comparison behavior (90% and 60% respectively), but this does not translate to reduced first-proposal bias. This suggests that Gemini's bias operates through a different mechanism that is not addressed by encouraging comparative language.

5 Discussion

The dramatic difference in QuoteBatch effectiveness between Claude (93.3pp reduction) and Gemini (10pp reduction) reveals fundamental heterogeneity in how LLMs respond to mechanism design interventions. This finding has important implications for deploying AI agents in economic systems: a one-size-fits-all approach to bias mitigation may not work across different model architectures. Our behavioral analysis suggests that the two models exhibit first-proposal bias through different mechanisms. Claude engages in explicit comparative reasoning but still anchors on the first proposal, making it responsive to anti-anchoring instructions. Gemini, in contrast, appears to make decisions without explicit comparison, and encouraging comparative language does not reduce its bias.

This work has several limitations. First, we tested only two models; generalization to other LLMs remains unknown. Second, our experiments use a single marketplace scenario with three contractors; real-world marketplaces may have different dynamics. Third, sample sizes are relatively small (10–15 runs per condition), though our key findings on Claude are statistically significant. Finally, we measure bias through proposal selection rank, which may not capture all aspects of decision quality.

Future work should extend this analysis to additional models and economic scenarios, investigate why models respond differently to mechanism design, and explore complementary interventions for models like Gemini that are resistant to the current approach.

6 Conclusion

We introduced QuoteBatch, a mechanism design intervention that combines hard-gate payment blocking with anti-anchoring prompt instructions to reduce first-proposal bias in agentic marketplaces. On Claude claude-sonnet-4-5, QuoteBatch achieves a 93.3 percentage point reduction in first-proposal bias (from 100% to 6.7%, $p < 0.001$) while maintaining 100% task completion. However, the same intervention yields only a 10 percentage point reduction on Gemini gemini-2.5-flash, revealing substantial heterogeneity in how LLMs respond to mechanism design. This finding underscores that deploying AI agents in economic systems requires model-specific bias mitigation strategies. Future work should investigate the mechanistic differences underlying this heterogeneity and develop interventions effective across diverse model architectures.

References

Gagan Bansal, Wenyue Hua, Zezhou Huang, Adam Fourney, Amanda Swearngin, Will Epperson, Tyler Payne, Jake M. Hofman, Brendan Lucier, Chinmay Singh, Markus Mobius, Akshay Nambi, Archana Yadav, Kevin Gao, David M. Rothschild, Aleksandrs Slivkins, Daniel G. Goldstein, Hussein Mozannar, Nicole Immorlica, Maya Murad, Matthew Vogel, Subbarao Kambhampati, Eric Horvitz, and Saleema Amershi. Magentic marketplace: An open-source environment for studying agentic markets, 2025. URL `https://arxiv.org/abs/2510.25779`.

Federico Bianchi, P. Chia, Mert Yüksekgönül, Jacopo Tagliabue, Daniel Jurafsky, and James Zou. How well can llms negotiate? negotiationarena platform and analysis. *ArXiv*, abs/2402.05863, 2024.

Thierry Blankenstein, Jialin Yu, Zixuan Li, Vassilis Plachouras, Sunando Sengupta, Philip H. S. Torr, Yarin Gal, Alasdair Paren, and Adel Bibi. Biasbusters: Uncovering and mitigating tool selection bias in large language models. *ArXiv*, abs/2510.00307, 2025.

Xiaobo Guo and Soroush Vosoughi. Serial position effects of large language models. *ArXiv*, abs/2406.15981, 2024.

J. Park, Joseph C. O'Brien, Carrie J. Cai, M. Morris, Percy Liang, and Michael S. Bernstein. *Generative Agents: Interactive Simulacra of Human Behavior*. 2023.

Narun K. Raman, Taylor Lundy, S. Amouyal, Yoav Levine, Kevin Leyton-Brown, and Moshe Tennenholtz. Steer: Assessing the economic rationality of large language models. pp. 42026–42047, 2024.

David Rothschild, Markus Mobius, Jake M. Hofman, E. Dillon, Daniel G. Goldstein, Nicole Immorlica, Sonia Jaffe, Brendan Lucier, Aleksandrs Slivkins, and Matthew Vogel. The agentic economy. *Communications of the ACM*, 69:39 – 42, 2025.

Eilam Shapira, Omer Madmon, Itamar Reinman, S. Amouyal, Roi Reichart, and Moshe Tennenholtz. Glee: A unified framework and benchmark for language-based economic environments. *ArXiv*, abs/2410.05254, 2024.

Felipe Valencia-Clavijo. Anchors in the machine: Behavioral and attributional evidence of anchoring bias in llms. *ArXiv*, abs/2511.05766, 2025.

Haonan Yin, Shai Vardi, and Vidyanand Choudhary. Fragile preferences: A deep dive into order effects in large language models. *ArXiv*, abs/2506.14092, 2025.

Stephan Zheng, Alexander R. Trott, Sunil Srinivasa, N. Naik, Melvin Gruesbeck, David C. Parkes, and R. Socher. The ai economist: Improving equality and productivity with ai-driven tax policies. *ArXiv*, abs/2004.13332, 2020.