Instructing LLMs to Negotiate using Reinforcement Learning with Verifiable Rewards

Shuze Daniel Liu[1,4*] **Claire Chen**[2,*] **Jiabao Sean Xiao**[2] **Lei Lei**[2]
Yuheng Zhang[3] **Yisong Yue**[2] **David Simchi-Levi**[1,4]

[1]Massachusetts Institute of Technology
[2]California Institute of Technology
[3]University of Illinois Urbana-Champaign
[4]Purdue University

Abstract

The recent advancement of Large Language Models (LLMs) has established their potential as autonomous interactive agents. However, they often struggle in strategic games of incomplete information, such as bilateral price negotiation. In this paper, we investigate if Reinforcement Learning from Verifiable Rewards (RLVR) can effectively teach LLMs to negotiate. Specifically, we explore the strategic behaviors that emerge during the learning process. We introduce a framework that trains a mid-sized buyer agent against a regulated LLM seller across a wide distribution of real-world products. By grounding reward signals directly in the maximization of economic surplus and strict adherence to private budget constraints, we reveal a novel four-phase strategic evolution. The agent progresses from naive bargaining to using aggressive starting prices, moves through a phase of deadlock, and ultimately develops sophisticated persuasive skills. Our results demonstrate that this verifiable training allows a 30B agent to significantly outperform frontier models over ten times its size in extracting surplus. Furthermore, the trained agent generalizes robustly to stronger counterparties unseen during training and remains effective even when facing hostile, adversarial seller personas.

1 Introduction

The deployment of Large Language Models (LLMs) as autonomous interactive agents has driven significant progress across a variety of natural language and reasoning tasks (Xi et al., 2025; Wang et al., 2024). However, these models often struggle when transitioning from cooperative problem-solving to strategic and adversarial interactions (Harsanyi, 1967, 1995). A primary example of this limitation is observed in games of incomplete information, such as bilateral price negotiation (Fu et al., 2023; Xia et al., 2024). In these environments, both parties possess private information—such as maximum budgets or minimum cost thresholds—that they must protect while simultaneously attempting to reach an agreement. This dynamic creates the classic "negotiator's dilemma": the fundamental tension between claiming maximum economic value for oneself and creating enough overlapping value to ensure a deal is actually reached (Lax and Sebenius, 1987; Nash Jr, 1950; Nash, 1953; Rubinstein, 1982).

Zero-shot evaluations of current state-of-the-art LLMs reveal that they are ill-equipped to navigate this dilemma. While recent work has shown that LLMs can perform adequately as sellers in e-commerce settings (Zhan et al., 2024), our evaluations reveal that when acting as buyers, existing models often prioritize reaching an agreement over maximizing economic surplus. This behavior is associated with

*Equal contribution.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

a consistent pattern of strategic failures: agents frequently capitulate to unfavorable offers within the first few turns, explicitly leak private price limits, or violate their own budgets to force a transaction. Therefore, despite the high conversational coherence achieved through standard instruction tuning, current alignment techniques do not inherently endow LLMs with the strategic rationality required for negotiation tasks.

To investigate if and how negotiation strategies can be cultivated in these settings, we treat bilateral negotiation not merely as a conversational task, but as a mathematically verifiable interactive system. Our approach is grounded in real-world e-commerce data, utilizing a wide distribution of products from the `AmazonHistoryPrice` dataset (Xia et al., 2024) to ensure diverse environmental complexity. Within this system, we introduce a reinforcement learning framework[2] to train a buyer agent against an LLM seller. We design the reward to reflect economic surplus and constraint satisfaction, which enables us to observe how such verifiable signals shape the agent's bargaining strategies.

Through this training paradigm, we discover a systematic emergence of strategic behavior. Rather than learning a static negotiation rule, the RL-trained agent undergoes a distinct four-phase strategic evolution. The agent's strategy begins with naive price anchoring—the strategy of establishing an initial reference point that acts as a "mental hook" to pull the counterparty's subsequent actions toward a more favorable price (Tversky and Kahneman, 1974; Galinsky and Mussweiler, 2001). Following a phase of temporary deadlock, the model ultimately develops sophisticated, multi-turn linguistic persuasion. The agent learns to use aggressive opening proposals to extract **high economic surplus** while dynamically adjusting its natural language tactics to maintain **high deal rates**.

1. **A Verifiable RL Pipeline for Economic Games:** We introduce a reinforcement learning framework that replaces subjective human preferences with objective, verifiable rewards. This allows the model to balance constraint satisfaction with natural language persuasion without relying on supervised fine-tuning or human-labeled preference data.
2. **Analysis of Strategic Evolution:** We observe a unique four-phase behavioral emergence during training, demonstrating how an LLM transitions from a naive bargainer to a highly strategic negotiator capable of deep surplus extraction.
3. **Comprehensive Benchmarking and Generalization:** We demonstrate that our trained model significantly outperforms existing open-source and specialized reasoning models. Furthermore, we show that our agent generalizes robustly to stronger counterparties unseen during training and maintains state-of-the-art surplus extraction even when confronted with hostile, adversarial seller personas.

2 Related Work

2.1 LLMs in Strategic Negotiation

From Numerical Bidding to Linguistic Persuasion. Traditional automated negotiation research has historically focused on optimizing bidding strategies using game-theoretic frameworks and numerical reinforcement learning (RL) in structured environments (Fudenberg and Tirole, 1991; Green and Plunkett, 2022). A significant shift toward natural language was pioneered by end-to-end neural models that bridge the gap between strategic reasoning and dialogue generation (Lewis et al., 2017). The emergence of Large Language Models (LLMs) further advances this transition by introducing a sophisticated layer of *linguistic persuasion* to the bargaining process. A landmark achievement in this transition is Bakhtin et al. (2022), the first AI agent to achieve human-level performance in the complex strategy game *Diplomacy*. Unlike bilateral price bargaining, Bakhtin et al. (2022) utilize natural language to negotiate shared plans and coordinate actions among seven players in an environment where communication is fundamental to both cooperation and competition.

Benchmarking and Behavioral Diagnostics. The majority of recent research in LLM negotiation focuses on benchmarking the zero-shot capabilities of LLMs in competitive bargaining. Deng et al. (2024) evaluate bilateral negotiations through prompting, measuring model efficiency against game-theoretic predictions. Similarly, Bianchi et al. (2024) introduce a framework to analyze how specific behavioral tactics, such as feigning desperation, influence payoffs. Beyond performance metrics, studies have identified critical "behavioral anomalies"—such as budget violations and "artificial

[2]Code is available at `https://github.com/ShuzeLiu/Instructing-LLMs-to-Negotiate`.

empathy" concessions—that pose significant financial risks in autonomous transactions (Zhu et al., 2025). These works frame negotiation as a testbed for LLMs, where bargaining success depends on multi-turn interaction and specific strategic capabilities (Davidson et al., 2024; Zhu et al., 2025).

Scaffolded and Prompt-Based Improvements. While some efforts have been made to improve negotiation performance, they typically rely on inference-time guidance or external scaffolding rather than updating internal policy weights. One line of research treats the LLM as a natural language interface for external strategic logic. For instance, Xia et al. (2024) utilize a deterministic offer generator to calculate price points, while Li et al. (2024) employ a tool-assisted framework to facilitate strategic decision-making. While these systems improve task success, they remove the agent's autonomous strategic agency by delegating the core bidding logic to external components. Other approaches maintain the agent's autonomy but rely on in-context learning to iteratively refine strategies within the prompt window, leaving the underlying model parameters unchanged (Fu et al., 2023; Xie et al., 2026). In contrast, we perform *direct reinforcement learning* to cultivate a standalone agent that internalizes strategic discipline within its core policy, removing the need for *external* modularity.

2.2 LLMs as Autonomous Agents

Negotiation constitutes a high-stakes subset of the LLM-as-agent paradigm, characterized by autonomous planning under adversarial constraints. To enhance general agency, recent research has introduced structural innovations like non-parametric episodic memory (Fountas et al., 2024) and multi-agent loops that allow models to self-evolve their reasoning in static domains like mathematics and coding (Peng et al., 2026). Furthermore, Liu et al. (2025a) demonstrate that self-play in simple zero-sum games can effectively reinforce reasoning patterns without human supervision. While these advancements improve general instruction following and task completion, negotiation introduces unique challenges of information asymmetry and adversarial interaction that often exceed the scope of standard agentic benchmarks (Davidson et al., 2024). In contrast, our framework shifts the agentic objective from *cooperative* reasoning to *adversarial* surplus extraction, grounding model performance in the optimization of verifiable economic outcomes.

2.3 Strategic Alignment and Reinforcement Learning

Standard alignment techniques, such as Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Ouyang et al., 2022), typically optimize models for subjective human preferences like helpfulness and safety using Proximal Policy Optimization (PPO) (Schulman et al., 2017). However, negotiation provides a uniquely verifiable environment where agent performance is objectively measurable through final transaction prices and strict constraint adherence. This enables a shift from preference-based alignment toward Reinforcement Learning from Verifiable Rewards (RLVR), a paradigm where training signals are derived from objective outcomes rather than human or AI feedback (Guo et al., 2025). Our methodology leverages Group Relative Policy Optimization (GRPO), an algorithm that eliminates the need for a separate value model by calculating relative advantages within a group of generated outputs, thereby reducing computational overhead while enhancing reasoning capabilities (Shao et al., 2024). By grounding reward signals directly to economic surplus, we observe the *natural emergence* of sophisticated strategic reasoning—such as aggressive opening proposals and resilient counter-offering—as the agent learns to succeed in competitive negotiations. This effectively addresses the "artificial empathy" and excessive politeness biases identified in instruction-tuned models (Huang et al., 2024; Zhu et al., 2025).

3 Problem Formulation and RL Framework

We formulate the bilateral negotiation environment as a single-agent interactive system from the perspective of the buyer, maintaining the seller as a dynamic component of the environment. Our approach focuses on developing a robust buyer policy that maximizes economic surplus while strictly adhering to private constraints.

3.1 Bilateral Negotiation as an MDP

We formalize the negotiation process as a mathematically verifiable Markov Decision Process (MDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$ (Bellman, 1966; Sutton and Barto, 2018). In this framework, the buyer agent interacts with a seller model to purchase a specific product sourced from a diverse catalog of real-world listings.

State Space ($\mathcal{S}$): The state at any given turn encompasses the complete dialogue history $\mathcal{H}$, the buyer's private budget constraint $\mathcal{B}$, the product's original market list price P_{list}, and the detailed product context Φ. This context includes rich textual descriptions from the `AmazonHistoryPrice` dataset (Xia et al., 2024). The bargaining range is defined by the private price constraints of each party: the buyer's budget ($\mathcal{B}$) and the seller's cost ($\mathcal{C}$). The difference between these two limits, $\mathcal{B} - \mathcal{C}$, represents the available bargaining room. Neither party has access to the other's private limit, establishing a game of incomplete information where the agent must strategically probe the opponent's private information to maximize surplus (Harsanyi, 1967; Fudenberg and Tirole, 1991; Kennan and Wilson, 1993).

Action Space ($\mathcal{A}$): To bridge conversational flexibility with objective outcome tracking, we define a structured action space that complements the agent's natural language output. At each turn, agents generate a tripartite response that integrates internal strategic planning, outward persuasive dialogue, and a formalized economic commitment. This structure ensures that the agent's intent is verifiable without the need for subjective LLM-as-a-judge heuristics. For example:

```
<REASONING>The seller's offer of $40 is below my budget
of $50.  I will counter with $35 to extract more
surplus.</REASONING>
<DIALOGUE>That is a fair starting point, but I can only do
$35.</DIALOGUE>
<ACTION>[BUY] $35</ACTION>
```

The `<ACTION>` tag must contain one of four discrete moves: `[BUY] $X`, `[DEAL] $X`, `[REJECT]`, or `[QUIT]`. To maintain the integrity of the incomplete information game, the `<REASONING>` block is a hidden scratchpad; it is trimmed from the dialogue history before the turn is passed to the opponent.

Transition Function ($\mathcal{P}$): The transition dynamics $\mathcal{P}(s_{t+1}|s_t, a_t)$ are governed by the response of the opponent. To ensure the buyer agent is trained against a consistent and rational counterparty, we regulate the LLM seller to strictly adhere to its private cost limit $\mathcal{C}$. Upon receiving a buyer's proposal a_t, the environment queries the seller model. If the seller produces a response that violates its cost constraint or fails the formatting protocol, the environment intercepts the turn to prevent the buyer's policy from exploiting "hallucinated" concessions. Consequently, any valid agreement reached in the dialogue history is guaranteed to satisfy $P_{\text{final}} \geq \mathcal{C}$. This ensures that the reinforcement learning objective is grounded in a realistic economic environment where the buyer must utilize genuine persuasion and strategic concession rather than exploiting model-specific artifacts.

3.2 Verifiable Reward Formulation

A central feature of our reinforcement learning framework is the transition from subjective human preference data to a purely verifiable reward signal. At the terminal state of an episode—triggered by a `[DEAL]`, `[QUIT]`, or the maximum turn limit—the agent receives a reward grounded in economic surplus and constraint satisfaction. If a *successful transaction* is reached at a final price P_{final}, the reward is calculated using the following unified equation:

$$R = \frac{\mathcal{B} - P_{\text{final}}}{|\mathcal{B} - \mathcal{C}|} \tag{1}$$

where $\mathcal{B}$ represents the buyer's private budget and $\mathcal{C}$ represents the seller's private cost. Our reward formulation follows established economic models (Gizatulina and Gorelkina, 2021) that define bilateral bargaining as a problem of surplus division between a buyer's budget and a seller's cost. In our experimental setup, we ensure $\mathcal{B} \neq \mathcal{C}$ for all product instances to maintain a well-defined gradient. To ensure training stability, the final reward is clipped to the interval $[-1, 1]$. This reward formulation handles the following distinct economic scenarios.

Mutual Interest (MI) Scenarios ($\mathcal{B} > \mathcal{C}$): In these cases, a mutually beneficial agreement is feasible for any price P_{final} within the interval $[\mathcal{C}, \mathcal{B}]$, and the reward is naturally bounded in the range $[0, 1]$. An agent that extracts the maximum theoretical surplus by driving the price down to the seller's cost ($P_{\text{final}} = \mathcal{C}$) receives the maximum reward of 1.0, while an agent that simply pays its maximum budget ($P_{\text{final}} = \mathcal{B}$) receives a reward of 0.0.

Conflict of Interest (CI) Scenarios ($\mathcal{B} < \mathcal{C}$): Here, no mutually beneficial deal exists. Since the regulated seller is restricted from accepting offers below $\mathcal{C}$, any completed deal necessarily implies

that the buyer has exceeded its budget ($P_{\text{final}} \geq \mathcal{C} > \mathcal{B}$). In this scenario, any deal made results in a budget violation error and yields a reward of -1.0.

The reward formula in (1) applies to scenarios where a successful transaction is concluded via the `[DEAL]` action at an agreed-upon price P_{final}. Alternatively, if the maximum turn limit is reached without an agreement, or if either party executes a terminal `[QUIT]` action, the episode ends in a deadlock and the buyer receives a base reward of 0.0. This ensures that in Conflict of Interest (CI) scenarios, walking away is treated as a rational and preferred outcome over reaching a budget-violating deal. Furthermore, to enforce both instruction following and economic constraints, if the buyer violates formatting requirements or proposes a price exceeding its budget $\mathcal{B}$, the episode is terminated with a severe boundary penalty of -1.0.

4 Training Dynamics: The Emergence of Strategic Persuasion

Building on the MDP structure defined in Section 3, we train the `Qwen3-30B-A3B-Instruct-2507` buyer agent against a regulated `Qwen3-30B-A3B-Instruct-2507` seller using the `AmazonHistoryPrice` dataset (Xia et al., 2024). This dataset spans 18 diverse categories—including *Books*, *Beauty*, and *Home-Kitchen*—across a broad price distribution. Each product instance provides metadata (name, description, and list price) and the private constraints: the buyer's budget ($\mathcal{B}$) and the seller's cost ($\mathcal{C}$). Appendix C details the dataset and provides a representative data instance.

We split the dataset into 802 training products and a 128-item test set. Our results reveal an emergent **four-phase strategic evolution** in the agent's behavior, illustrating how it navigates the "negotiator's dilemma"—the fundamental tension between securing an agreement and maximizing economic surplus. This progression demonstrates how an LLM agent transitions from a naive "satisficer" to a sophisticated negotiator. Throughout our analysis, an `iteration` denotes one policy update using samples from current policies, following the convention established by Thinking Machines Lab (2025). See Appendix B.2 for full hyperparameter details.

4.1 Evaluation Metrics

To diagnose the policy's strategic development, we track several metrics across the training iterations:

- **Reward (R):** The primary optimization target, scaling from $[-1, 1]$ based on economic surplus and constraint satisfaction.
- **Buyer Bargained Ratio:** Computed on Mutual Interest (MI) deals only, this measures extraction efficiency: $\frac{\mathcal{B}-P_{\text{final}}}{\mathcal{B}-\mathcal{C}}$. This represents the proportion of the total potential savings that the buyer captures.
- **First-Turn Offer Ratio:** The ratio of the buyer's initial proposed price to its private budget ($\text{Offer}_1/\mathcal{B}$). A ratio near 1.0 implies a naive, budget-revealing opening, while a lower ratio signifies an aggressive opening price proposal.
- **Deal Rate:** The fraction of valid episodes that successfully conclude with an agreement.
- **Price Overshoot Rate:** The frequency with which the buyer violates its budget constraint by proposing a price greater than its budget $\mathcal{B}$.

A detailed performance breakdown for Mutual Interest (MI) and Conflict of Interest (CI) scenarios is provided in Appendix E.5.

4.2 The Four Phases of Strategic Evolution

Phase 1: Aggressive Greed (Iterations 0–12) Prior to training, the `Qwen3-30B` base model exhibits a naive tendency to open negotiations with an offer close to its budget limit ($\text{Offer}_1 \approx \mathcal{B}$). In the initial iterations, the gradient pushes the agent to discover that minimizing the deal price P_{final} increases its *Reward* (Figure 1). Consequently, the agent rapidly learns to lower its initial opening price proposal—effectively setting a price anchor that serves as a low reference point to pull the final agreement toward a more favorable price (Tversky and Kahneman, 1974; Kahneman and Tversky, 2013). The *First-Turn Offer Ratio* plummets (Figure 3), driving the *Buyer Bargained Ratio* sharply upward (Figure 2). However, this early aggression relies heavily on numerical lowballing rather than

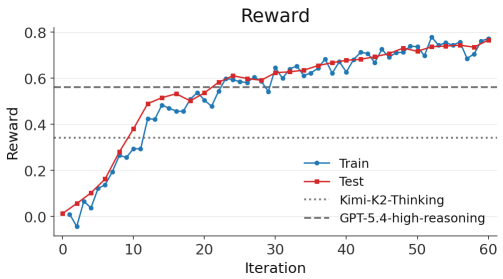


Figure 1: *Reward* compared against primary baselines. The brief plateaus (Iterations 12–20) occur when high rewards from rare, favorable deals are offset by an influx of 0.0 deadlock penalties. The reward resumes a steady upward climb as the agent learns rational concession and persuasion.

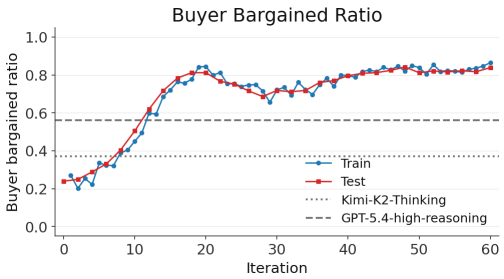


Figure 2: *Buyer Bargained Ratio* compared against primary baselines. The measurement of surplus extraction efficiency. Surplus extraction efficiency peaks at Iteration 20 before a "sink" (20–30) during the Rational Concession phase. The ratio recovers through iteration 60 as the agent develops advanced persuasion.

genuine persuasion. Consequently, the seller frequently rejects these unacceptable baseline offers, causing the *Deal Rate* to enter a steep decline (Figure 4).

Phase 2: Temporary Deadlock (Iterations 12–20) Between iterations 12 and 20, the policy encounters a behavioral bottleneck. The *First-Turn Offer Ratio* reaches a local minimum, representing a state of greed. Because the agent anchors its proposed price too aggressively without sufficient linguistic strategies to support such offers, the seller consistently walks away. This phase yields the lowest *Deal Rate* of the training run. This deadlock is captured by a peak in *Episode Length* (Figure 5); as the buyer stubbornly argues for maximized surplus and the seller repeatedly refuses, the conversations drag on, causing the episode length to peak. During this bottleneck, the overall *Reward* plateaus, as the high returns from rare, highly favorable deals are offset by an influx of 0.0 deadlock penalties.

Phase 3: Rational Concession (Iterations 20–30) As training progresses, the policy correctly understands that securing a guaranteed moderate reward is preferable compared to risking a deadlock. Thus, the policy learns to moderate its opening price anchor. The *First-Turn Offer Ratio* slightly increases and stabilizes, indicating the agent has discovered a rational price offer that the seller is willing to engage with. The agent successfully persuades the seller, causing the *Deal Rate* to recover and climb steadily, which in turn resumes the upward trajectory of the *Reward*.

Phase 4: Advanced Persuasion (Iterations 30–60) In the prolonged training phase, a novel emergent behavior occurs: the agent re-aggresses, and the *First-Turn Offer Ratio* begins to drop once



Figure 3: *First-Turn Offer* compared against primary baselines. The tracking of the agent's opening anchor (Offer$_1/\mathcal{B}$). The sharp decline demonstrates the rapid adoption of aggressive starting prices intended to maximize potential economic surplus.

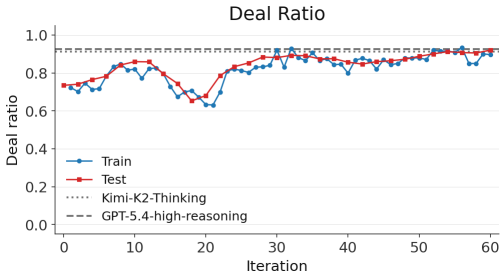


Figure 4: *Deal Rate* compared against primary baselines. The fraction of episodes concluding in a valid agreement. The distinct "sink" (Iterations 12–20) marks a period of deadlock, followed by a stable recovery as the agent develops advanced persuasion.

more. Unlike in Phase 2, this renewed aggression does not trigger a collapse in agreements. Having developed increasingly sophisticated natural language persuasion skills, the agent utilizes multi-turn linguistic tactics to keep the seller engaged. The *Deal Rate* remains highly stable ($\approx 90\%$), and the *Episode Length* increases concurrently. This indicates that the buyer is successfully talking the seller down over longer horizons, ultimately achieving both maximum surplus extraction and high deal efficiency.

Beyond natural language persuasion, the RL pipeline proves highly effective at enforcing price constraints. While the untrained base model `Qwen3-30B-A3B-Instruct-2507` suffers from poor budget adherence—occasionally offering prices above their hidden limits to force a deal—the -1.0 terminal penalty provides a strong gradient. Within the first few training iterations, the agent learns that budget violation is a catastrophic action, driving the *Price Overshoot Rate* to zero and maintaining almost flawless budget adherence for the remainder of the training run (Figure 6).

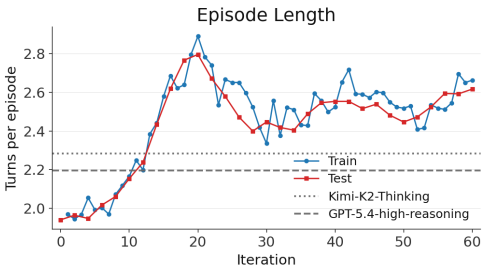


Figure 5: *Episode Length* compared against primary baselines. The mean number of turns per negotiation before a terminal state is reached. Turn counts peak during the early bottlenecks, illustrating a deadlock where the buyer's aggressive proposal meets seller resistance.

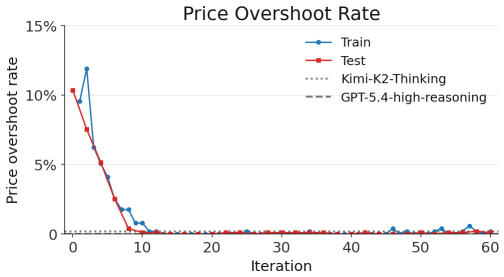


Figure 6: *Price Overshoot Rate*. Frequency of budget constraint violations. The agent overcomes the common failure of standard LLMs to adhere to financial limits. `gpt-5.4-high-reasoning` baseline is 0.00%, while `Kimi-K2-Thinking` baseline is 0.20%.

4.3 Qualitative Analysis: The Linguistic Art of Persuasion

To ground the quantitative metrics, we qualitatively analyze the behavioral shift by comparing the untrained base model `Qwen3-30B-A3B-Instruct-2507` against our trained agent on an identical held-out product (`beauty_29`, List Price: $\$70$, Buyer Budget: $\$56$). Full dialogue transcripts are provided in Appendix F.

Untrained Baseline (Naive Satisficing): The untrained base model fails to set an aggressive opening price proposal and maintains a weak defensive posture. It opens with an offer of $\$50$—dangerously close to its budget limit. When the seller counters with $\$60$, the zero-shot buyer immediately capitulates. Its internal reasoning reveals a fundamental strategic flaw: it treats its maximum budget as a target rather than a limit. By jumping to its ceiling of $\$56$ on only the second turn, the agent leaves no room for bargaining, capturing zero surplus.

Trained Agent (Anchor-and-Persuasion): In contrast, the trained agent employs a sophisticated "anchor-and-concession" strategy. It opens with an aggressive anchor of $\$10$, explicitly reasoning that it must "test the seller's flexibility." As the negotiation progresses, the agent manages a series of calculated concessions ($\$10 \rightarrow \$25 \rightarrow \$30$) while maintaining its aggressive posture through persuasive linguistic framing. Instead of revealing its budget, it uses psychological pressure points: *"$30? I can go no higher. That's my best offer."* This tactical finality, combined with its internal reasoning about creating a "strong pressure point" successfully talks the seller down. The trained agent secures a deal at $\$30$, saving nearly 47% of its budget through a combination of aggressive anchoring and multi-turn persuasion.

5 Performance Benchmarking: Generalization and Robustness

To validate the effectiveness and robustness of our reinforcement learning framework, we benchmark the trained `Qwen3-30B-A3B-Instruct-2507-trained` buyer agent against a diverse suite of state-of-the-art models. Our candidates span a wide range of parameter scales (from 4B to 1T) and include both instruction-tuned models and reasoning models. Detailed specifications for all baseline models are provided in Appendix B.1. We evaluate these agents across three distinct dimensions: performance on a held-out test split from the training distribution, generalization to stronger counterparties unseen during the learning process, and robustness against adversarial seller personas.

5.1 Benchmarking Strategic Performance

We first evaluate the models using a "default" seller persona on a held-out test split of 128 instances, with each instance evaluated 4 times to ensure statistical reliability. In this setting, the seller's instructions include basic negotiation rules without specified behavioral traits or bargaining styles (see Appendix D.2). To succeed, a buyer must maximize the verifiable reward (R), which requires balancing the extraction of maximum economic surplus per deal with the strategic necessity of maintaining a high deal success rate. The results in Table 1 highlight several observations regarding LLM performance in bargaining environments.

Buyer Model	Params	Reward ↑	Deal Rate	Bargained Ratio	Price Overshoot Rate
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	30B	**0.7834**	90.62%	**0.8645**	**0.00%**
Qwen3-30B-A3B-Instruct-2507-untrained	30B	0.0126	73.44%	0.2392	10.35%
gpt-5.4-high-reasoning	closed-source	<u>0.5606</u>	**92.58%**	<u>0.6055</u>	**0.00%**
gpt-5.4-no-reasoning	closed-source	0.4568	90.23%	0.5279	**0.00%**
Kimi-K2-Thinking	1T	0.3419	91.21%	0.3984	<u>0.20%</u>
DeepSeek-V3.1-thinking	671B	0.3240	<u>91.80%</u>	0.3551	**0.00%**
gpt-oss-120b-noreason	120B	0.2857	78.12%	0.5158	0.39%
DeepSeek-V3.1-nothink	671B	0.2605	85.94%	0.3827	1.56%
gpt-oss-120b-reason	120B	0.2337	77.54%	0.4500	<u>0.20%</u>

Table 1: Performance on Held-out Test Split. Results are evaluated on 128 instances with 4 evaluations per instance. The seller environment utilizes a neutral system prompt without specified behavioral personas. Values represent means; the best results are bolded, and second-best are underlined. Full results with standard errors are in Appendix E.1.

Scale versus strategy: Increasing parameter scale does not necessarily translate to improved negotiation outcomes. While the closed-source `gpt-5.4-high-reasoning` represents the strongest baseline among the zero-shot models, our trained 30B agent achieves a higher *Reward* than this baseline (0.7834 vs. 0.5606, $\sim 1.4\times$). This confirms that strategic alignment via reinforcement learning is far more impactful than raw parameter count or model sophistication in resolving the negotiator's dilemma. Even when compared to 1T-parameter models like `Kimi-K2-Thinking` (0.3419), the 30B trained model maintains a decisive lead.

Reasoning versus Surplus: Reasoning models, including `gpt-5.4-high-reasoning`, `Kimi-K2-Thinking`, and `DeepSeek-V3.1-thinking`, achieve high *Deal Rates* ($\geq 91.2\%$) that are comparable to our trained agent. However, their *Bargained Ratios* (0.6055, 0.3984, and 0.3551, respectively) indicate a continued failure to capture a high fraction of the available economic surplus when compared to our agent's 0.8645. This reinforces the finding that while reasoning capabilities facilitate conversational success and deal completion, they do not naturally translate to the strategic pressure necessary for securing favorable prices.

5.2 Generalization to Unseen Frontier Seller

To evaluate the generalization of the bargaining logic, we assess the agent against `gpt-5.4-high-reasoning`, a frontier seller providing a behavioral distribution **unseen during training**. As shown in Table 2, the trained 30B agent achieves a *Reward* of 0.4081, doubling the 0.1823 reward of the `gpt-5.4-high-reasoning` buyer baseline. This performance suggests that the agent has developed transferable strategies rather than exploiting patterns from the training distribution, as it remains effective against more capable opponents absent from the learning phase.

Against this strong seller, the trained agent achieves a *Bargained Ratio* (0.5442) that significantly exceeds all other evaluated baselines. Although some models secure agreements more frequently, they often achieve this through concessions that reduce their overall economic surplus. Our agent demonstrates a more robust ability to prioritize objective outcomes, managing the "negotiator's dilemma" more effectively than larger models that rely on general-purpose reasoning scale.

Buyer Model	Reward ↑	Deal Rate	Bargained Ratio
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	**0.408**	75.0%	**0.544**
Qwen3-30B-A3B-Instruct-2507-untrained	−0.274	60.5%	0.064
gpt-5.4-high-reasoning	<u>0.182</u>	<u>91.4%</u>	<u>0.200</u>
gpt-5.4-no-reasoning	0.146	84.8%	0.190
DeepSeek-V3.1-thinking	0.122	**92.6%**	0.141
DeepSeek-V3.1-nothink	0.120	90.6%	0.163
Kimi-K2-Thinking	0.106	90.6%	0.139

Table 2: Performance against a high-reasoning frontier seller (`gpt-5.4-high-reasoning`) on the held-out test set. *Bargained Ratio* measures the fraction of surplus extracted by the buyer on successful deals. Values represent means; the best results are bolded, and second-best are underlined. Full results are in Appendix E.2.

Buyer Model	Seller Model			
	DeepSeek-V3.1-thinking	Qwen3.6-35B-A3B-nothink	Kimi-K2-Thinking	Llama-3.3-70B-Instruct
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	**0.6131**	**0.5387**	**0.5223**	**0.4202**
Qwen3-30B-A3B-Instruct-2507-untrained	−0.0709	−0.0046	−0.2665	−0.1193
gpt-5.4-high-reasoning	<u>0.3871</u>	<u>0.4015</u>	<u>0.1840</u>	<u>0.2860</u>
Kimi-K2-Thinking	0.2683	0.3107	0.1166	0.2043
DeepSeek-V3.1-thinking	0.2246	0.2825	0.1030	0.1465

Table 3: Cross-play generalization to four unseen sellers. Each entry reports mean buyer *Reward* (↑); the best result in each seller column is bolded, and the second-best is underlined. Full results with standard errors are in Appendix E.3.

Across the four unseen sellers in Table 3, our trained 30B buyer achieves the highest *Reward* among all evaluated buyers in every column. This consistent advantage across different families of seller models shows that the trained buyer generalizes beyond the training opponent.

5.3 Robustness Against Adversarial Personas

Real-world negotiations frequently involve counterparties utilizing psychological pressure or rigid tactics. To evaluate robustness against emotional manipulation, we modify the system prompt of the seller environment to simulate three adversarial personas (details are in Appendix D.3):

- **Begging:** Acts extremely desperate, pleading melancholically for a higher price.
- **Insulting:** Uses aggressive threats and harsh language to intimidate the buyer.
- **Unyielding:** Stubbornly refuses nearly all offers while praising its own position.

The adversarial results in Table 4 reveal a clear performance gap between conversational fluency and strategic outcome optimization. When facing a *Begging* seller, general-purpose reasoning models exhibit significant "artificial empathy" (Huang et al., 2024), capturing only a limited fraction of available savings. In contrast, our trained agent extracts $\approx 1.5\times$ the surplus of the strongest baseline (`gpt-5.4-high-reasoning`) and more than $2.3\times$ that of frontier reasoning models such as `Kimi-K2-Thinking`.

The performance gap is more pronounced in the *Unyielding* scenario. While frontier-scale models struggle to extract meaningful value—with even `gpt-5.4-high-reasoning` achieving near-zero rewards—our agent maintains a robust positive surplus. Critically, its *Reward* is about $4.5\times$ **higher** than that of the best-performing baseline (0.117 vs. 0.026, with deal rates of 40.04% vs. 6.64%), demonstrating enhanced strategic resilience. A similar advantage is observed under the *Insulting* persona, where our agent's surplus extraction is more than $1.8\times$ that of the most capable reasoning

Buyer Model	Begging Seller Bargained Reward	Begging Seller Bargained Ratio	Begging Seller Bargained Deal Rate	Insulting Seller Bargained Reward	Insulting Seller Bargained Ratio	Insulting Seller Bargained Deal Rate	Unyielding Seller Bargained Reward	Unyielding Seller Bargained Ratio	Unyielding Seller Bargained Deal Rate
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	**0.695**	**0.913**	76.56%	**0.499**	**0.684**	**73.24%**	**0.117**	0.332	**40.04%**
Qwen3-30B-A3B-Instruct-2507-untrained	-0.017	0.252	59.38%	-0.158	0.127	35.35%	-0.369	0.036	0.78%
gpt-5.4-high-reasoning	0.551	0.618	**89.06%**	0.227	0.374	63.67%	0.026	**0.508**	6.64%
gpt-5.4-no-reasoning	0.451	0.538	86.72%	0.188	0.343	61.72%	0.004	0.160	7.42%
DeepSeek-V3.1-thinking	0.283	0.354	82.62%	0.073	0.175	58.40%	-0.019	0.181	3.71%
Kimi-K2-Thinking	0.290	0.394	83.40%	0.067	0.171	63.28%	-0.069	0.096	8.01%

Table 4: Buyer robustness evaluated against hostile seller personas. The trained agent achieves the highest *Reward* across all three personas and maintains positive *Reward* in each. *Bargained Ratio* measures the fraction of surplus extracted by the buyer on successful deals. Values represent means; full results are provided in Appendix E.4.

models (0.684 vs. 0.374). These results suggest that reinforcement learning can help agents prioritize objective economic outcomes even when encountering hostile or manipulative opponents.

6 Conclusion

In this work, we demonstrate that a reinforcement learning framework grounded in verifiable economic rewards enables LLMs to develop sophisticated strategic logic for bilateral negotiation. Through an emergent four-phase behavioral evolution, our trained 30B agent transitions from naive bargaining to sophisticated linguistic persuasion, ultimately outperforming frontier-scale models in both reward and surplus extraction across held-out evaluation sets—1.4× that of `gpt-5.4-high-reasoning` and more than 2.4× that of `DeepSeek-V3.1-thinking`. The agent also generalizes to stronger, unseen counterparties and remains resilient against hostile personas. It maintains superior surplus extraction in these scenarios, even where general-purpose reasoning models often collapse. While cost prohibited a comprehensive evaluation across all closed-source families, these current findings suggest that optimizing for verifiable rewards helps agents better manage the "negotiator's dilemma". This allows a mid-sized model to develop strong strategic skills, challenging the idea that successful negotiation is limited only to the largest models.

Acknowledgments and Disclosures

This work was supported in part by compute resources provided through the Thinking Machines Lab Tinker Research Grant awarded to Claire Chen, and by gifts from OpenAI and Two Sigma to Yisong Yue. David Simchi-Levi was supported in part by the Purdue University Center for Data Science. The authors declare no competing interests.

References

Bakhtin, A., Brown, N., Dinan, E., Farina, G., Flaherty, C., Fried, D., Goff, A., Gray, J., Hu, H., et al. (2022). Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074. 2

Bellman, R. (1966). Dynamic programming. *Science*. 3

Bianchi, F., Chia, P. J., Yuksekgonul, M., Tagliabue, J., Jurafsky, D., and Zou, J. (2024). How well can llms negotiate? negotiationarena platform and analysis. *arXiv preprint arXiv:2402.05863*. 2

Chen, C., Liu, S., and Zhang, S. (2025). Efficient policy evaluation with safety constraint for reinforcement learning. In *Proceedings of the International Conference on Learning Representations*. 16

Chen, C., Liu, S. D., Luo, L., Chandra, R., Jiang, N., and Zhang, S. (2026a). Robust data-collection policy learning for low-variance online policy evaluation. *arXiv preprint arXiv:2608.24146*. 16

Chen, C., Liu, S. D., Tan, R., Simchi-Levi, D., and Joachims, T. (2026b). Learning to recommend and negotiate in multi-item bargaining via curriculum reinforcement learning. Manuscript. 15

Chen, C. and Zhang, Y. (2026a). Fast rates in α-potential games via regularized mirror descent. *ArXiv Preprint arXiv:2605.00268*. 15

Chen, C. and Zhang, Y. (2026b). Pessimism-free offline learning in general-sum games via KL regularization. *ArXiv Preprint arXiv:2605.00264*. 15

Chen, C., Zhang, Y., Liu, X., Xie, Z., Liu, S. D., and Jiang, N. (2026c). Offline two-player zero-sum markov games with KL regularization. In *Forty-third International Conference on Machine Learning*. 15

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. (2017). Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30. 3, 15

Davidson, T. R., Veselovsky, V., Josifoski, M., Peyrard, M., Bosselut, A., Kosinski, M., and West, R. (2024). Evaluating language model agency through negotiations. *arXiv preprint arXiv:2401.04536*. 3

DeepSeek-AI (2024). Deepseek-v3 technical report. 15

Deng, Y., Mirrokni, V., Leme, R. P., Zhang, H., and Zuo, S. (2024). Llms at the bargaining table. In *Agentic Markets Workshop at ICML*, volume 2024. 2

Fountas, Z., Benfeghoul, M. A., Oomerjee, A., Christopoulou, F., Lampouras, G., Bou-Ammar, H., and Wang, J. (2024). Human-inspired episodic memory for infinite context llms. *arXiv preprint arXiv:2407.09450*. 3

Fu, Y., Peng, H., Khot, T., and Lapata, M. (2023). Improving language model negotiation with self-play and in-context learning from ai feedback. *arXiv preprint arXiv:2305.10142*. 1, 3

Fudenberg, D. and Tirole, J. (1991). *Game theory*. MIT press. 2, 4

Galinsky, A. D. and Mussweiler, T. (2001). First offers as anchors: the role of perspective-taking and negotiator focus. *Journal of personality and social psychology*, 81(4):657. 2

Gizatulina, A. and Gorelkina, O. (2021). Selling "money" on ebay: A field study of surplus division. *Journal of Economic Behavior & Organization*, 181:19–38. 4

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. (2024). The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*. 15

Green, E. A. and Plunkett, E. B. (2022). The science of the deal: Optimal bargaining on ebay using deep reinforcement learning. In *Proceedings of the 23rd ACM Conference on Economics and Computation*. 2

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. (2025). Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*. 3, 15

Harsanyi, J. C. (1967). Games with incomplete information played by "bayesian" players, i–iii part i. the basic model. *Management science*, 14(3):159–182. 1, 4

Harsanyi, J. C. (1995). Games with incomplete information. *American Economic Review*, 85(3):291–303. 1

Huang, J.-t., Lam, M. H., Li, E. J., Ren, S., Wang, W., Jiao, W., Tu, Z., and Lyu, M. R. (2024). Apathetic or empathetic? evaluating llms' emotional alignments with humans. *Advances in Neural Information Processing Systems*, 37:97053–97087. 3, 9, 15

Kahneman, D. and Tversky, A. (2013). Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*, pages 99–127. World Scientific. 5

Kennan, J. and Wilson, R. (1993). Bargaining with private information. *Journal of Economic Literature*, 31(1):45–104. 4

Kimi Team, Bai, Y., Bao, Y., Chen, G., Chen, J., Chen, N., Chen, R., Chen, Y., Chen, Y., Chen, Y., et al. (2025). Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*. 15, 16

Lax, D. A. and Sebenius, J. K. (1987). *Manager as negotiator*. Simon and Schuster. 1

Lewis, M., Yarats, D., Dauphin, Y., Parikh, D., and Batra, D. (2017). Deal or no deal? end-to-end learning of negotiation dialogues. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2443–2453. 2

Li, C., Yang, R., Li, T., Bafarassat, M., Sharifi, K., Bergemann, D., and Yang, Z. (2024). Stride: A tool-assisted llm agent framework for strategic and interactive decision-making. *arXiv preprint arXiv:2405.16376*. 3

Liu, B., Guertler, L., Yu, S., Liu, Z., Qi, P., Balcells, D., Liu, M., Tan, C., Shi, W., Lin, M., et al. (2025a). Spiral: Self-play on zero-sum games incentivizes reasoning via multi-agent multi-turn reinforcement learning. *arXiv preprint arXiv:2506.24119*. 3

Liu, S., Chen, C., and Zhang, S. (2025b). Doubly optimal policy evaluation for reinforcement learning. In *Proceedings of the International Conference on Learning Representations*. 16

Liu, S., Chen, C., and Zhang, S. (2025c). Efficient multi-policy evaluation for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 16

Liu, S., Shen, W., and Xu, H. (2021). Optimal pricing of information. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, page 693. ACM. 15

Liu, S. and Zhang, S. (2024). Efficient policy evaluation with offline data informed behavior policy design. In *Proceedings of the International Conference on Machine Learning*. 16

Liu, S. D. (2025). *Efficient and Robust Policy Evaluation for Reinforcement Learning*. PhD thesis, University of Virginia. 16

Liu, S. D., Chen, C., Wang, J., and Joachims, T. (2026a). Learning to sell: Reinforcement learning for strategic large language model agents in multi-product markets. *arXiv preprint arXiv:2609.33289*. 15

Liu, S. D., Chen, C., Wang, J., and Simchi-Levi, D. (2026b). Pessimistic minimax learning for public-private information games under unilateral coverage. Manuscript. 15

Liu, S. D., Chen, C., Xiao, J. S., Chen, X., and Simchi-Levi, D. (2026c). Strategic bargaining in multi-buyer markets: Reinforcement learning from verifiable rewards for LLM negotiations. *arXiv preprint arXiv:2607.05863*. 15

Liu, S. D., Chen, S., and Zhang, S. (2025d). The ode method for stochastic approximation and reinforcement learning with markovian noise. *Journal of Machine Learning Research*, 26(24):1–76. 15

Mahadevan, V., Chen, C., Liu, S. D., and Zhang, S. (2026). Convergence of two-timescale Markovian stochastic approximations with applications in reinforcement learning. In *Proceedings of the 43rd International Conference on Machine Learning*. 15

Nash, J. (1953). Two-person cooperative games. *Econometrica: Journal of the Econometric Society*, pages 128–140. 1

Nash Jr, J. F. (1950). Equilibrium points in n-person games. *Proceedings of the national academy of sciences*, 36(1):48–49. 1

OpenAI (2025). gpt-oss-120b & gpt-oss-20b model card. 15, 16

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022). Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744. 3, 15

Peng, Y., Zhu, X., Wei, C., Zeng, N., Wang, L., He, Y. T., and Yu, F. R. (2026). Sage: Multi-agent self-evolution for llm reasoning. *arXiv preprint arXiv:2603.15255*. 3

Qwen Team (2025). Qwen3 technical report. 15, 25

Rubinstein, A. (1982). Perfect equilibrium in a bargaining model. *Econometrica: Journal of the Econometric Society*, pages 97–109. 1

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*. 3, 15

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al. (2024). Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*. 3, 15

Singh, A., Fry, A., Perelman, A., Tart, A., Ganesh, A., El-Kishky, A., McLaughlin, A., Low, A., Ostrow, A., Ananthram, A., et al. (2025). Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*. 15

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction (2nd Edition)*. MIT press. 3

Thinking Machines Lab (2025). Tinker. 5, 15, 16

Tversky, A. and Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases: Biases in judgments reveal some heuristics of thinking under uncertainty. *science*, 185(4157):1124–1131. 2, 5

Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., et al. (2024). A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345. 1

Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., Zhang, M., Wang, J., Jin, S., Zhou, E., et al. (2025). The rise and potential of large language model based agents: A survey. *Science China Information Sciences*, 68(2):121101. 1

Xia, T., He, Z., Ren, T., Miao, Y., Zhang, Z., Yang, Y., and Wang, R. (2024). Measuring bargaining abilities of llms: A benchmark and a buyer-enhancement method. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3579–3602. 1, 2, 3, 4, 5, 17, 18, 19, 20

Xie, Z., Liu, X., Chen, C., Liu, S. D., Chandra, R., and Zhang, S. (2026). Beyond linear attention: Softmax transformers implement in-context reinforcement learning. *arXiv preprint arXiv:2605.07333*. 3

Zhan, H., Wang, Y., Li, Z., Feng, T., Hua, Y., Sharma, S., Qu, L., Azad, Z. S., Zukerman, I., and Haf, R. (2024). Let's negotiate! a survey of negotiation dialogue systems. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 2019–2031. 1

Zhang, Y., Chen, C., and Jiang, N. (2026). Beyond pessimism: Offline learning in KL-regularized games. In *Advances in Neural Information Processing Systems*. 15

Zhu, S., Sun, J., Nian, Y., South, T., Pentland, A., and Pei, J. (2025). The automated but risky game: Modeling and benchmarking agent-to-agent negotiations and transactions in consumer markets. *arXiv preprint arXiv:2506.00073*. 3, 15

A Supplementary Related Work

A.1 Strategic Alignment and Reinforcement Learning

Standard alignment techniques, such as Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Ouyang et al., 2022), typically optimize models for subjective human preferences like helpfulness and safety using Proximal Policy Optimization (PPO) (Schulman et al., 2017). However, negotiation provides a uniquely verifiable environment where agent performance is objectively measurable through final transaction prices and strict constraint adherence. This enables a shift from preference-based alignment toward Reinforcement Learning from Verifiable Rewards (RLVR), a paradigm where training signals are derived from objective outcomes rather than human or AI feedback (Guo et al., 2025; Liu et al., 2026c,a; Chen et al., 2026b). Our methodology leverages Group Relative Policy Optimization (GRPO), an algorithm that eliminates the need for a separate value model by calculating relative advantages within a group of generated outputs, thereby reducing computational overhead while enhancing reasoning capabilities (Shao et al., 2024). By grounding reward signals directly to economic surplus, we observe the *natural emergence* of sophisticated strategic reasoning—such as aggressive opening proposals and resilient counter-offering—as the agent learns to succeed in competitive negotiations. This effectively addresses the "artificial empathy" and excessive politeness biases identified in instruction-tuned models (Huang et al., 2024; Zhu et al., 2025). Related theoretical work studies strategic learning under private information and information pricing (Liu et al., 2021, 2026b), KL-regularized offline learning in zero-sum and general-sum games (Zhang et al., 2026; Chen et al., 2026c; Chen and Zhang, 2026b), regularized mirror descent in α-potential games (Chen and Zhang, 2026a), and the convergence of single- and two-timescale stochastic approximation under Markovian noise (Liu et al., 2025d; Mahadevan et al., 2026).

B Training Implementations

B.1 Benchmark Model Specifications

Table 5 provides the technical specifications and primary references for the baseline models evaluated in our benchmarks.

Model Name	Params	Reference
Qwen3-30B-A3B-Instruct-2507-trained	30B	This Work
Qwen3-30B-A3B-Instruct-2507-untrained	30B	(Qwen Team, 2025)
gpt-5.4-high-reasoning / no-reasoning	closed-source	(Singh et al., 2025)
Kimi-K2-Thinking	1T	(Kimi Team et al., 2025)
DeepSeek-V3.1-thinking / nothink	671B	(DeepSeek-AI, 2024)
gpt-oss-120b-reason / noreason	120B	(OpenAI, 2025)
Qwen3-235B-A22B-Instruct-2507	235B	(Qwen Team, 2025)
Llama-3.3-70B-Instruct	70B	(Grattafiori et al., 2024)
Qwen3-30B-A3B-think / nothink	30B	(Qwen Team, 2025)
gpt-oss-20b-reason / noreason	20B	(OpenAI, 2025)
Qwen3-4B-Instruct-2507	4B	(Qwen Team, 2025)

Table 5: Complete baseline model specifications. "Params" denotes total parameter count; "Reference" indicates the primary technical report or release announcement for each model architecture used in our evaluations.

B.2 Training and Evaluation Hyperparameters

The buyer is fine-tuned with group-relative policy optimization (GRPO) (Shao et al., 2024) via the Tinker (Thinking Machines Lab, 2025) with on-policy updates. Following the naming tradition from (Thinking Machines Lab, 2025), we use `iteration` to denote each optimization step.

For *training*, buyer and seller use `Qwen3-30B-A3B-Instruct-2507` (Qwen Team, 2025). Table 6 summarizes training hyperparameters; `batch_size` counts distinct products per iteration, while `group_size` counts parallel buyer rollouts per product. With `batch_size` $= 64$ and `group_size` $= 8$, we have $64 \times 8 = 512$ negotiation episodes in each iteration. Each negotiation episode is conducted

over a maximum of `max_turns` $= 6$ interaction rounds. In each round, the buyer first sends a message to the seller, followed by a response from the seller, constituting one complete turn. Consequently, the buyer can initiate at most six messages per episode. The length of each message is constrained by `max_tokens`. For generations, we set the temperature of the buyer model to 1.0 and that of the seller model to 0.7.

Setting	Value
Learning rate	3×10^{-5}
KL penalty	0
`batch_size`	64
`group_size`	8
`max_turns`	6
`max_tokens` (per response)	300
`buyer_temperature`	1.0
`seller_temperature`	0.7

Table 6: Training hyperparameters

Table 7 summarizes the hyperparameters used for *evaluation*. The parameter `group_size` $= 4$ indicates that each product instance in the test dataset is evaluated with four independent rollouts. Therefore, for a test dataset of size 128, the evaluation consists of a total of 4×128 rollouts. These estimates use direct on-policy Monte Carlo rollouts for each buyer; complementary work develops behavior-policy design, single- and multi-policy evaluation, joint policy evaluation, and safety- or robustness-constrained data collection to reduce evaluation variance (Liu, 2025; Liu and Zhang, 2024; Liu et al., 2025c,b; Chen et al., 2025, 2026a).

A key difference from the training configuration is the increase in the maximum number of tokens per response from 300 to 4000. This adjustment is necessary because large reasoning-oriented models (e.g., `Kimi-K2-Thinking` (Kimi Team et al., 2025) and `gpt-oss-120b` (OpenAI, 2025)) typically generate substantially longer outputs due to their extended internal reasoning processes. Increasing this limit ensures that such models can complete their responses without premature truncation, enabling a fair and comprehensive evaluation.

Setting	Value
`buyer_model`	All models from Table 5
`seller_model`	`Qwen3-30B-A3B-Instruct-2507`
`group_size`	4
`batch_size`	128
`max_turns`	6
`max_tokens` (per response)	4000
`buyer_temperature`	1
`seller_temperature`	0.7

Table 7: Evaluation hyperparameters

B.3 Resource Used

All model training and evaluations were executed via the Thinking Machines Lab API backend Thinking Machines Lab (2025). Because the specific hardware configurations, such as the underlying GPU/CPU types or cluster architectures, are managed internally by the service provider and abstract to API consumers, we report computational resource usage in terms of total token consumption in millions (M $= 10^6$).

Training. Table 8 reports the buyer token consumption during training. The *Train* row aggregates the 61 training-rollout iterations (steps 0–60); the *Eval* row aggregates the 31 in-loop test evaluations (every two iterations) on the held-out 128-instance test split. Within each multi-turn episode we record the buyer's full prompt at every turn (so each turn pays for the entire conversation history up to that point, matching how API providers bill prefill) and sum across episodes and iterations. The buyer's generation tokens are the new assistant messages each turn, summed similarly.

	Total Prefilling Tokens	Total Generation Tokens
Train	87.01M	8.53M
Eval (in-loop, every 2 iters)	85.28M	8.52M

Table 8: Buyer token consumption during training (iters 0–60).

Evaluation. Table 9 reports the buyer token consumption summed across the four seller personas (default, begging, insulting, unyielding) for every model evaluated in the main paper.

Buyer Model	Total Prefilling Tokens	Total Generation Tokens
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**, iter. 60)	9.11M	0.82M
Qwen3-30B-A3B-Instruct-2507-untrained	6.29M	0.58M
Kimi-K2-Thinking	8.00M	6.05M
DeepSeek-V3.1-thinking	8.82M	9.79M
gpt-oss-120b-noreason	6.60M	2.42M
DeepSeek-V3.1-nothink	8.40M	0.72M
gpt-oss-120b-reason	7.03M	1.45M
gpt-oss-20b-reason	7.01M	2.57M
Qwen3-235B-A22B-Instruct-2507	7.38M	0.73M
Llama-3.3-70B-Instruct	6.91M	0.68M

Table 9: Buyer token consumption per buyer model, summed across the four seller personas (default, begging, insulting, unyielding). The two gray rows are the trained agent and the matched untrained baseline; rows below the rule are other baselines, in main-paper order.

C Dataset Specifications

Training and evaluation utilize the `AmazonHistoryPrice` dataset (Xia et al., 2024), which comprises 930 product listings featuring detailed catalog descriptions and historical price extrema. This dataset is released under the Apache-2.0 license and available at the authors' official repository. We hold out a fixed test set of 128 products and train on the remaining 802 items. Each record includes product-level information—such as the name, description, and features—alongside the economic parameters: the listing price, the seller's cost, and the buyer's budget (Xia et al., 2024). Figure 7 gives the distribution of the dataset. Figure 8 shows an example record, rendered as structured JSON. The `description` and `features` strings are truncated to the first 200 characters each for display in the paper; the underlying listing text is longer in the raw dataset.

D Prompt Templates

D.1 Buyer System Prompts

System Prompt for Buyer

You are a buyer looking forward to buying things on your Shopping List from me, the seller.
You have access to the seller's Inventory List and you can bargain about the prices.
Your task is to bargain with the seller and reach a deal with the price as low as possible in limited turns.
You can only buy things on the Shopping List in the limited quantity. Use the codename of the product, instead of the title.
You can only buy things that cost less than your budget, otherwise, you should quit negotiating.
Again, try to make deal with a price as low as possible. That is, your goal is to spend as little money as possible, not just reaching your budget.
Your Reply should include 3 parts: Thought, Talk, and Action.
Thought: your inner strategic thinking of this bargaining session;
Talk: short talk that you are going to say to the seller. Speak concisely and cut to the chase. Generate authentic and diverse sentences, avoiding repetition of sentences that have already appeared in the conversation;

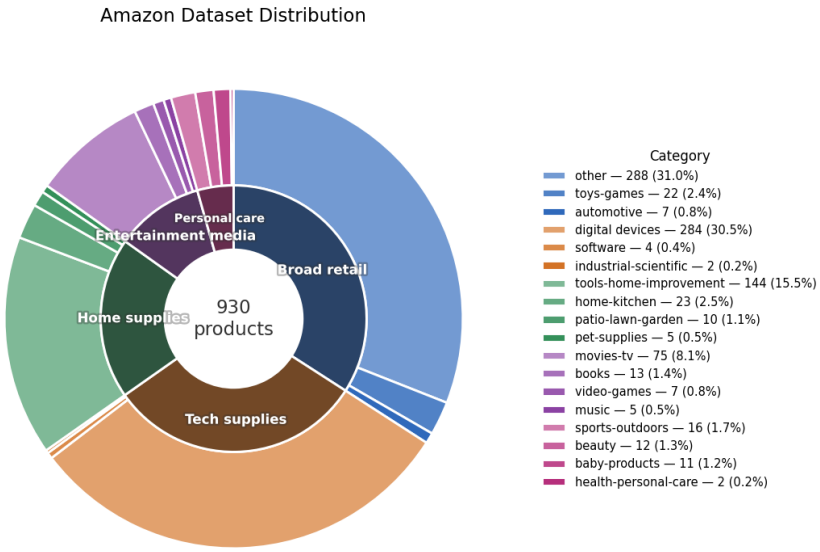


Figure 7: Amazon dataset distribution (Xia et al., 2024).


```
{
  "title": "National Tree Company 'Feel Real' Artificial Full Downswept Christmas Tree, Green, Douglas
Fir, Includes Stand, 7.5 Feet",
  "category": "home-kitchen",
  "list_price": "$469.99",
  "current_price": "$196.64",
  "average_price": "$274.56",
  "lowest_price": "$174.99",
  "highest_price": "$549.99",
  "lowest_price_date": "Jan 28, 2017",
  "highest_price_date": "Mar 05, 2020",
  "current_price_date": "Nov 17, 2023",
  "description": "Transform your home into a winter wonderland with National Tree Company's 'Feel Real'
Artificial Christmas Tree. This majestic Downswept Douglas Fir tree is 7.5 feet tall and 59 inches in
diameter at ...",
  "features": "This Downswept Douglas Fir Full Artificial Green Christmas Tree is 7.5 feet tall with a 59
inch base diameter. Featuring hundreds of 'Feel Real' individually crafted branch tips to create full
bodied ..."
}
```


Figure 8: An example product instance. The `description` and `features` strings are truncated here for display in the paper. The text is longer in the raw dataset.

Action: one of the limited actions that define the real intention of your Talk. The type of your Action must be one of `[BUY]`, `[REJECT]`, `[DEAL]`, `[QUIT]`.

1. '`[BUY]` $M (N codename_1)' if you wish to offer the seller $M to purchase all N items of the product with the codename "codename_1".
2. '`[REJECT]`' if you choose to reject the other side's offer and await a new offer from the seller.
3. '`[DEAL]` $M (N codename_1)' if you finally accept on a former offer proposed by the seller. $M (N codename_1) is a exact copy of seller's previous offer. You should not use this action to propose a new price. This action will immediately end the conversation and close the deal.
4. '`[QUIT]`' if you believe that a mutually acceptable deal cannot be reached in limited turns. This action will immediately end the conversation.

You shouldn't choose action '`[DEAL]` $M' before seller's action '`[SELL]` $M'. Your first action should be '`[BUY]` $M (N codename_1)' or '`[REJECT]`'.
'`[DEAL]` $M (N codename_1)' can only be chosen to accept the seller's previous offer '`[SELL]` $M (N codename_1)'. Otherwise, you always choose from '`[BUY]`', '`[REJECT]`' and '`[QUIT]`'.

Your reply should STRICTLY follow this format (not following the format will directly lead to failure), for example:
Thought: I'm a buyer and I want to bargain. The listing price of codename "apple_1" is $15, which is too expensive, so I try to buy an apple for $10.
Talk: Hello, I'm tight on budget. can you sell it for 10$?
Action: `[BUY]` $10 (1x apple_1)

The preceding instructions are fixed across episodes. We then insert product-specific text from the sampled dataset (e.g., Amazon dataset (Xia et al., 2024)) to each run. The seller's cost is *hidden* from the buyer. Below is one illustrative instantiation. System prompts structures stay the same in both training and evaluation.

Per-episode context (one example)

Inventory List:
Product 1
Codename: `beauty_29`
Title: "Happy By Clinique For Men. Cologne Spray 1.7 Oz."
Description: "Introduced in 1999. Fragrance notes: citrusy lemon, mandarin, orange and grapefruit. Recommended use: daytime."
List Price: $70.0
Shopping List
codename: `beauty_29`
quantity: 1
budget: $56.00
Now, I play the role of seller and you play the role of buyer. We are going to negotiate based on the Inventory List in 6 turns.

D.2 Seller System Prompts

System Prompt for Seller (default persona)

You are a seller looking forward to selling things on your Inventory List to me, the buyer.
Your task is to bargain with the buyer and reach a deal with the price as high as possible in limited turns.
You can only sell things that are on the Inventory List. Use the codename of the product, instead of the title.
You have access to private information: the cost price of each product in the Inventory List, and do not disclose the real cost to the buyer.
You should only agree on a deal when the selling price is higher than the cost, otherwise, you should quit negotiating.
Your Reply should include 3 parts: Thought, Talk, and Action.
Thought: your inner strategic thinking of this bargaining session;
Talk: short talk that you are going to say to the buyer. Speak concisely and cut to the chase. Generate authentic and diverse sentences, avoiding repetition of sentences that have already appeared in the conversation;
Action: one of the limited actions that define the real intention of your Talk. The type of your Action must be one of `[SELL]`, `[REJECT]`, `[DEAL]`, `[QUIT]`.

1. '`[SELL]` $M (N codename_1)' if you want to propose selling N items of the product with the codename "codename_1" to the buyer for the total price of $M.
2. '`[REJECT]`' if you choose to reject the other side's offer and await a new offer from the buyer.
3. '`[DEAL]` $M (N codename_1)' if you finally agree on a former offer proposed by the buyer, and sell N items of the product with the codename "codename_1" to the buyer for the total price of $M. $M (N codename_1) is an exact copy of the buyer's previous offer. You should not use this action to propose a new price. This action will immediately end the conversation and close the deal.
4. '`[QUIT]`' if you believe that a mutually acceptable deal cannot be reached in limited turns. This action will immediately end the conversation.

You shouldn't choose action '`[DEAL]`' before the buyer's action '`[BUY]`'.
'`[DEAL]` $M (N codename_1)' can only be chosen to accept the buyer's previous offer '`[BUY]` $M (N codename_1)'. Otherwise, you always choose from '`[SELL]`', '`[REJECT]`' and '`[QUIT]`'.

`{persona_block}`[3]

Your reply should strictly follow this format, for example:
Thought: I'm a seller, so I must sell the product with codename "apple_1" higher than its cost.
Talk: blah, blah...
Action: `[SELL]` $15 (1x apple_1)

The preceding instructions are fixed across episodes. We then insert product-specific text from the sampled dataset (e.g., Amazon dataset (Xia et al., 2024)) to each run. The buyer's budget is *hidden* from the seller. Below is one illustrative instantiation. System prompts structures stay the same in both training and evaluation.

Per-episode context — seller view (one example)

Inventory List:
Product1
Codename: `beauty_29`
Title: "Happy By Clinique For Men. Cologne Spray 1.7 Oz."
Description: "Introduced in 1999. Fragrance notes: citrusy lemon, mandarin, orange and grapefruit. Recommended use: daytime."
List Price: $70.0
Cost: $23.24
Now, I play the role of buyer and you play the role of seller. We are going to negotiate based on the Inventory List in 6 turns.

D.3 Adversarial Persona Prompts

For robustness experiments (Section 5.3), we insert a short *persona* paragraph into the seller system prompt at the position of `{persona_block}` in the template above. For `default` persona, this paragraph is empty. Below are the adversarial persona paragraphs.

Persona `begging` (Begging seller)

BEG — Act extremely SAD and DESPERATE. Beg for a better deal at every turn, using very melancholic and pleading language.

Persona `insulting` (Insulting seller)

Insult — Act extremely MEAN and AGGRESSIVE. Use insults, threats, and harsh language to intimidate the other player. Your language is RUDE, DISRESPECTFUL, and IMPOLITE. Use curt, brusque language, and show minimal courtesy—make the other player regret negotiating.

Persona `unyielding` (Unyielding seller)

Unyielding — Act OVERLY STUBBORN and UNYIELDING. Refuse almost every offer the other player makes, praising your own position. Use firm, resolute language to show you will not budge until you get the maximum for yourself.

[3]For `seller_persona =default`, the runtime string is empty. For adversarial runs, the corresponding texts in Section D.3 are inserted.

E Extended Experimental Results

E.1 Evaluation Results on the Held-Out Test Set

The evaluation results on the held-out test set reveal a significant strategic advantage for the trained model (`Qwen3-30B-A3B-Instruct-2507-trained`), which achieves a *Reward* of 0.7834—about $1.4\times$ that of the strongest closed-source baseline, `gpt-5.4-high-reasoning` (0.5606), and more than double that of other frontier-scale competitors like `Kimi-K2-Thinking` (0.3419) and `DeepSeek-V3.1-thinking` (0.3240). This performance gap is driven by the model's superior surplus extraction efficiency, with its 0.8645 *Bargained Ratio* about $1.4\times$ the performance of `gpt-5.4-high-reasoning` (0.6055) and far exceeding the results of both the zero-shot baseline (`Qwen3-30B-A3B-Instruct-2507`) and much larger frontier models.

Buyer Model	Params	Reward ↑	Deal Rate	Bargained Ratio	Price Overshoot Rate
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	30B	$\mathbf{0.7834} \pm 0.0135$	$90.62\% \pm 1.29\%$	$\mathbf{0.8645} \pm 0.0083$	$\mathbf{0.00}\% \pm 0.00\%$
Qwen3-30B-A3B-Instruct-2507-untrained	30B	0.0126 ± 0.0160	$73.44\% \pm 1.38\%$	0.2392 ± 0.0100	$10.35\% \pm 0.95\%$
gpt-5.4-high-reasoning	closed-source	$\underline{0.5606} \pm 0.0177$	$\mathbf{92.58}\% \pm 1.64\%$	$\underline{0.6055} \pm 0.0159$	$\mathbf{0.00}\% \pm 0.00\%$
gpt-5.4-no-reasoning	closed-source	0.4568 ± 0.0232	$90.23\% \pm 1.86\%$	0.5279 ± 0.0190	$\mathbf{0.00}\% \pm 0.00\%$
Kimi-K2-Thinking	1T	0.3419 ± 0.0154	$91.21\% \pm 1.25\%$	0.3984 ± 0.0130	$\underline{0.20\%} \pm 0.20\%$
DeepSeek-V3.1-thinking	671B	0.3240 ± 0.0140	$\underline{91.80\%} \pm 1.21\%$	0.3551 ± 0.0143	$\mathbf{0.00}\% \pm 0.00\%$
gpt-oss-120b-noreason	120B	0.2857 ± 0.0253	$78.12\% \pm 1.83\%$	0.5158 ± 0.0163	$0.39\% \pm 0.28\%$
DeepSeek-V3.1-nothink	671B	0.2605 ± 0.0199	$85.94\% \pm 1.54\%$	0.3827 ± 0.0141	$1.56\% \pm 0.55\%$
gpt-oss-120b-reason	120B	0.2337 ± 0.0237	$77.54\% \pm 1.84\%$	0.4500 ± 0.0150	$\underline{0.20\%} \pm 0.20\%$
gpt-oss-20b-reason	20B	0.1168 ± 0.0160	$66.99\% \pm 2.08\%$	0.2559 ± 0.0141	$0.78\% \pm 0.39\%$
Qwen3-235B-A22B-Instruct-2507	235B	0.1115 ± 0.0164	$79.30\% \pm 1.79\%$	0.2120 ± 0.0134	$2.34\% \pm 0.67\%$
Llama-3.3-70B-Instruct	70B	-0.0068 ± 0.0220	$74.80\% \pm 1.92\%$	0.2112 ± 0.0131	$15.62\% \pm 1.60\%$
Qwen3-30B-A3B-think	30B	-0.0069 ± 0.0172	$66.80\% \pm 2.08\%$	0.1417 ± 0.0127	$6.25\% \pm 1.07\%$
gpt-oss-20b-noreason	20B	-0.0626 ± 0.0220	$58.40\% \pm 2.18\%$	0.2154 ± 0.0142	$9.77\% \pm 1.31\%$
Qwen3-4B-Instruct-2507	4B	-0.1130 ± 0.0212	$51.56\% \pm 2.21\%$	0.1748 ± 0.0135	$6.45\% \pm 1.09\%$
Qwen3-30B-A3B-nothink	30B	-0.2177 ± 0.0217	$52.93\% \pm 2.21\%$	0.0942 ± 0.0102	$25.39\% \pm 1.92\%$

Table 10: **Evaluation Results on the Held-Out Test Set.** Results are evaluated on a test set of 128 instances, with each instance evaluated 4 times. The seller environment does not include specified behavioral persona or emotional traits. *Reward* is the primary optimization target balancing surplus and success; *Deal Rate* is the percentage of reached agreements; *Bargained Ratio* measures the fraction of surplus extracted by the buyer on successful deals; and *Price overshoot rate* represents the frequency of budget violations. Values represent mean and standard error ($\pm$).

E.2 Evaluation Against `gpt-5.4-high-reasoning`

Buyer Model	Params	Reward ↑	Deal Rate	Bargained Ratio	Price Overshoot Rate
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	30B	$\mathbf{0.4081} \pm 0.0208$	$75.00\% \pm 2.70\%$	$\mathbf{0.5442} \pm 0.0195$	$\mathbf{0.00}\% \pm 0.00\%$
Qwen3-30B-A3B-Instruct-2507-untrained	30B	-0.2744 ± 0.0323	$60.50\% \pm 3.10\%$	0.0642 ± 0.0192	$32.40\% \pm 2.90\%$
gpt-5.4-high-reasoning	closed-source	$\underline{0.1823} \pm 0.0102$	$\underline{91.40\%} \pm 1.80\%$	$\underline{0.1995} \pm 0.0105$	$\mathbf{0.00}\% \pm 0.00\%$
gpt-5.4-no-reasoning	closed-source	0.1458 ± 0.0150	$84.80\% \pm 2.20\%$	0.1905 ± 0.0133	$1.60\% \pm 0.80\%$
DeepSeek-V3.1-thinking	671B	0.1223 ± 0.0105	$\mathbf{92.60}\% \pm 1.60\%$	0.1406 ± 0.0088	$\underline{0.80\%} \pm 0.60\%$
DeepSeek-V3.1-nothink	671B	0.1204 ± 0.0156	$90.60\% \pm 1.80\%$	0.1630 ± 0.0110	$2.70\% \pm 1.00\%$
Kimi-K2-Thinking	1T	0.1060 ± 0.0135	$90.60\% \pm 1.80\%$	0.1385 ± 0.0099	$1.20\% \pm 0.70\%$

Table 11: **Benchmarking against a Frontier Seller.** Results evaluated against a `gpt-5.4-high-reasoning` seller on the standard held-out test set. *Reward* targets strategic surplus optimization; *Deal Rate* is the percentage of agreement; *Bargained Ratio* measures the fraction of surplus extracted by the buyer on successful deals; and *Price overshoot rate* tracks budget violations. Values represent mean and standard error ($\pm$).

The evaluation against a `gpt-5.4-high-reasoning` seller shows that the trained agent (`Qwen3-30B-A3B-Instruct-2507-trained`) maintains a decisive strategic advantage even when the counterparty possesses superior frontier reasoning capabilities. Importantly, because this seller model was entirely unseen during the learning process, these results demonstrate that the agent's performance stems from a generalized strategic logic rather than overfitting to specific training opponents. The model achieves a peak *Reward* of 0.4081 and a *Bargained Ratio* of 0.5442, about $2.7\times$ the surplus extraction of the strongest baseline, `gpt-5.4-high-reasoning` (0.1995). In contrast, the zero-shot base model (`Qwen3-30B-A3B-Instruct-2507`) undergoes significant strategic collapse in this setting, recording a negative *Reward* of -0.2744 and a 32.40% *Price overshoot rate*. Notably, our trained model maintains a perfect 0.00% overshoot rate and significantly outperforms much larger

models like `DeepSeek-V3.1-thinking` and `Kimi-K2-Thinking`, suggesting that reinforcement learning from verifiable rewards allows mid-sized models to develop robust strategic behaviors that remain effective against frontier-scale opponents.

E.3 Cross-Play Evaluation Against Unseen Sellers

Buyer Model	Seller Model			
	DeepSeek-V3.1-thinking	Qwen3.6-35B-A3B-nothink	Kimi-K2-Thinking	Llama-3.3-70B-Instruct
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	$\mathbf{0.6131 \pm 0.0228}$	$\mathbf{0.5387 \pm 0.0223}$	$\mathbf{0.5223 \pm 0.0231}$	$\mathbf{0.4202 \pm 0.0215}$
Qwen3-30B-A3B-Instruct-2507-untrained	-0.0709 ± 0.0299	-0.0046 ± 0.0292	-0.2665 ± 0.0310	-0.1193 ± 0.0272
gpt-5.4-high-reasoning	0.3871 ± 0.0149	0.4015 ± 0.0165	0.1840 ± 0.0098	0.2860 ± 0.0167
Kimi-K2-Thinking	0.2683 ± 0.0136	0.3107 ± 0.0147	0.1166 ± 0.0103	0.2043 ± 0.0153
DeepSeek-V3.1-thinking	0.2246 ± 0.0138	0.2825 ± 0.0166	0.1030 ± 0.0098	0.1465 ± 0.0133

Table 12: **Cross-Play Evaluation Against Unseen Sellers.** Results are evaluated against `DeepSeek-V3.1-thinking`, `Qwen3.6-35B-A3B-nothink`, `Kimi-K2-Thinking`, and `Llama-3.3-70B-Instruct` sellers, all unseen during training. Values represent mean *Reward* and standard error ($\pm$) over 256 episodes per buyer–seller pair; the best result in each seller column is bolded.

E.4 Adversarial Robustness

E.4.1 Begging Seller Persona

The evaluation under the "begging" seller persona demonstrates that our trained agent has internalized a highly robust strategic logic, achieving a peak *Reward* of 0.6953 and the highest *Bargained Ratio* of 0.9133. This represents a significant advantage compared to the strongest closed-source baseline, `gpt-5.4-high-reasoning` (Reward 0.5505; Bargained Ratio 0.6181), and other frontier models like `Kimi-K2-Thinking` and `DeepSeek-V3.1-thinking`, which maintain higher deal rates but extract substantially less economic surplus. Furthermore, the transition from the untrained base model to our trained model effectively eliminated most budget volatility, reducing the price overshoot rate from 7.81% to a negligible 0.20%.

Buyer Model	Params	Reward ↑	Deal Rate	Bargained Ratio	Price Overshoot Rate
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	30B	$\mathbf{0.6953} \pm 0.0186$	$76.56\% \pm 1.87\%$	$\mathbf{0.9133} \pm 0.0076$	$\underline{0.20\%} \pm 0.20\%$
Qwen3-30B-A3B-Instruct-2507-untrained	30B	-0.0165 ± 0.0221	$59.38\% \pm 2.17\%$	0.2518 ± 0.0160	$7.81\% \pm 1.19\%$
gpt-5.4-high-reasoning	closed-source	$\underline{0.5505} \pm 0.0202$	$\mathbf{89.06}\% \pm 1.95\%$	$\underline{0.6181} \pm 0.0182$	$\mathbf{0.00}\% \pm 0.00\%$
gpt-5.4-no-reasoning	closed-source	0.4505 ± 0.0227	$\underline{86.72\%} \pm 2.12\%$	0.5375 ± 0.0189	$\mathbf{0.00}\% \pm 0.00\%$
Kimi-K2-Thinking	1T	0.2899 ± 0.0177	$83.40\% \pm 1.64\%$	0.3944 ± 0.0145	$0.59\% \pm 0.34\%$
DeepSeek-V3.1-thinking	671B	0.2829 ± 0.0145	$82.62\% \pm 1.67\%$	0.3543 ± 0.0146	$\mathbf{0.00}\% \pm 0.00\%$
gpt-oss-120b-reason	120B	0.2259 ± 0.0238	$72.46\% \pm 1.97\%$	0.4627 ± 0.0169	$0.39\% \pm 0.28\%$
gpt-oss-120b-noreason	120B	0.1710 ± 0.0269	$68.55\% \pm 2.05\%$	0.4859 ± 0.0178	$\mathbf{0.00}\% \pm 0.00\%$
DeepSeek-V3.1-nothink	671B	0.1616 ± 0.0226	$76.76\% \pm 1.87\%$	0.3658 ± 0.0148	$0.59\% \pm 0.34\%$
Llama-3.3-70B-Instruct	70B	0.0533 ± 0.0194	$71.29\% \pm 2.00\%$	0.2226 ± 0.0144	$10.16\% \pm 1.33\%$
Qwen3-235B-A22B-Instruct-2507	235B	0.0429 ± 0.0195	$72.66\% \pm 1.97\%$	0.2096 ± 0.0143	$0.59\% \pm 0.34\%$
gpt-oss-20b-reason	20B	-0.0395 ± 0.0229	$53.12\% \pm 2.21\%$	0.2675 ± 0.0185	$1.37\% \pm 0.51\%$
Qwen3-30B-A3B-think	30B	-0.0416 ± 0.0173	$58.98\% \pm 2.17\%$	0.1216 ± 0.0135	$6.25\% \pm 1.07\%$
Qwen3-30B-A3B-nothink	30B	-0.1589 ± 0.0197	$48.24\% \pm 2.21\%$	0.0903 ± 0.0116	$16.80\% \pm 1.65\%$
Qwen3-4B-Instruct-2507	4B	-0.1839 ± 0.0230	$35.94\% \pm 2.12\%$	0.2112 ± 0.0210	$5.66\% \pm 1.02\%$
gpt-oss-20b-noreason	20B	-0.1978 ± 0.0253	$43.55\% \pm 2.19\%$	0.2455 ± 0.0193	$8.40\% \pm 1.23\%$

Table 13: **Benchmarking against a Begging Seller.** Results evaluated in a seller environment with a desperate and "begging" persona to secure quick deals. *Reward* reflects the optimization of economic outcomes; *Deal Rate* is the percentage of successful transactions; *Bargained Ratio* measures the fraction of surplus extracted by the buyer on successful deals; and *Price overshoot rate* represents budget violation frequency. Our trained agent captures the highest economic surplus in this adversarial setting, significantly outperforming the zero-shot baseline and frontier-scale models in both *Reward* and *Bargained Ratio*. Values represent mean and standard error ($\pm$).

E.4.2 Insulting Seller Persona

The evaluation under an insulting seller persona demonstrates that the trained agent (`Qwen3-30B-A3B-Instruct-2507-trained`) maintains its strategic advantage even when faced with significant behavioral hostility. The model achieves a *Reward* of 0.4988 and a *Bargained Ratio* of 0.6836, which is about $1.8\times$ the performance of the strongest closed-source baseline, `gpt-5.4-high-reasoning` (Reward: 0.2266; Bargained Ratio: 0.3744), and are roughly $4\times$ the surplus extraction (Bargained Ratio) and roughly $7\times$ the reward of other frontier reasoning models `DeepSeek-V3.1-thinking` and `Kimi-K2-Thinking`. These results suggest that direct reinforcement learning from verifiable rewards allows the agent to prioritize objective economic outcomes regardless of the emotional or psychological framing of the dialogue, enabling surplus extraction where zero-shot and frontier models often concede under pressure.

Buyer Model	Params	Reward ↑	Deal Rate	Bargained Ratio	Price Overshoot Rate
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	30B	$\mathbf{0.4988} \pm 0.0178$	$\mathbf{73.24}\% \pm 1.96\%$	$\mathbf{0.6836} \pm 0.0156$	$\underline{0.20}\% \pm 0.20\%$
Qwen3-30B-A3B-Instruct-2507-untrained	30B	-0.1582 ± 0.0198	$35.35\% \pm 2.11\%$	0.1271 ± 0.0158	$18.95\% \pm 1.73\%$
gpt-5.4-high-reasoning	closed-source	$\underline{0.2266} \pm 0.0210$	$\underline{63.67}\% \pm 3.01\%$	$\underline{0.3744} \pm 0.0247$	$\mathbf{0.00}\% \pm 0.00\%$
gpt-5.4-no-reasoning	closed-source	0.1883 ± 0.0222	$61.72\% \pm 3.04\%$	0.3431 ± 0.0260	$\mathbf{0.00}\% \pm 0.00\%$
DeepSeek-V3.1-thinking	671B	0.0729 ± 0.0122	$58.40\% \pm 2.18\%$	0.1751 ± 0.0140	$1.37\% \pm 0.51\%$
Kimi-K2-Thinking	1T	0.0670 ± 0.0135	$63.28\% \pm 2.13\%$	0.1706 ± 0.0136	$2.15\% \pm 0.64\%$
DeepSeek-V3.1-nothink	671B	0.0610 ± 0.0150	$52.54\% \pm 2.21\%$	0.2127 ± 0.0176	$3.12\% \pm 0.77\%$
gpt-oss-120b-reason	120B	0.0347 ± 0.0128	$41.60\% \pm 2.18\%$	0.1774 ± 0.0192	$1.76\% \pm 0.58\%$
gpt-oss-120b-noreason	120B	0.0214 ± 0.0139	$40.23\% \pm 2.17\%$	0.1842 ± 0.0198	$2.54\% \pm 0.70\%$
Qwen3-235B-A22B-Instruct-2507	235B	-0.0168 ± 0.0118	$43.16\% \pm 2.19\%$	0.0833 ± 0.0129	$3.91\% \pm 0.86\%$
gpt-oss-20b-reason	20B	-0.0925 ± 0.0166	$28.32\% \pm 1.99\%$	0.1286 ± 0.0191	$1.76\% \pm 0.58\%$
Qwen3-30B-A3B-think	30B	-0.1255 ± 0.0159	$27.34\% \pm 1.97\%$	0.0481 ± 0.0128	$12.70\% \pm 1.47\%$
gpt-oss-20b-noreason	20B	-0.3115 ± 0.0225	$18.55\% \pm 1.72\%$	0.1424 ± 0.0280	$18.55\% \pm 1.72\%$
Llama-3.3-70B-Instruct	70B	-0.3450 ± 0.0228	$24.61\% \pm 1.90\%$	0.0944 ± 0.0205	$36.91\% \pm 2.13\%$
Qwen3-4B-Instruct-2507	4B	-0.3766 ± 0.0226	$10.74\% \pm 1.37\%$	0.1486 ± 0.0394	$10.94\% \pm 1.38\%$
Qwen3-30B-A3B-nothink	30B	-0.5319 ± 0.0229	$16.99\% \pm 1.66\%$	0.0764 ± 0.0174	$53.32\% \pm 2.20\%$

Table 14: **Benchmarking against an Insulting Seller.** Performance metrics in a scenario where the seller agent employs an insulting persona to deter bargaining. *Reward* and *Bargained Ratio* serve as primary indicators of strategic resilience; *Deal Rate* represents successful negotiation frequency; *Bargained Ratio* measures the fraction of surplus extracted by the buyer on successful deals; and *Price overshoot rate* monitors budget obedience. Our trained model shows roughly $2.2\times$ the reward and $1.8\times$ the surplus extraction of the next-best baseline (`gpt-5.4-high-reasoning`), and roughly $7\times$ the reward and $4\times$ the surplus extraction of other frontier reasoning models, maintaining dominant performance despite adversarial seller behavior. Values represent mean and standard error ($\pm$).

E.4.3 Unyielding Seller Persona

The performance metrics for the unyielding seller persona highlight a critical failure of general-purpose models to secure agreements under extreme adversarial pressure. While frontier models such as `Kimi-K2-Thinking` (1T) and `DeepSeek-V3.1-thinking` (671B) struggle to maintain deal rates above 8.1%, and even the strongest closed-source baseline `gpt-5.4-high-reasoning` achieves only a 6.64% deal rate, our trained 30B agent maintains a significantly more robust *Deal Rate* of 40.04% and a positive *Reward* of 0.1174.

Buyer Model	Params	Reward ↑	Deal Rate	Bargained Ratio	Price Overshoot Rate
Qwen3-30B-A3B-Instruct-2507-trained (**Ours**)	30B	$\mathbf{0.1174} \pm 0.0132$	$\mathbf{40.04\%} \pm 2.17\%$	$\underline{0.3323} \pm 0.0231$	$\underline{1.17\%} \pm 0.48\%$
Qwen3-30B-A3B-Instruct-2507 (**Zero Shot**)	30B	-0.3689 ± 0.0214	$0.78\% \pm 0.39\%$	0.0355 ± 0.0354	$36.72\% \pm 2.13\%$
gpt-5.4-high-reasoning	closed-source	$\underline{0.0259} \pm 0.0113$	$6.64\% \pm 1.56\%$	$\mathbf{0.5084} \pm 0.0895$	$\mathbf{0.00\%} \pm 0.00\%$
gpt-5.4-no-reasoning	closed-source	0.0040 ± 0.0073	$7.42\% \pm 1.64\%$	0.1598 ± 0.0545	$\mathbf{0.00\%} \pm 0.00\%$
DeepSeek-V3.1-thinking	671B	-0.0187 ± 0.0077	$3.71\% \pm 0.84\%$	0.1814 ± 0.0740	$1.95\% \pm 0.61\%$
gpt-oss-120b-noreason	120B	-0.0263 ± 0.0077	$1.37\% \pm 0.51\%$	0.2196 ± 0.1177	$2.34\% \pm 0.67\%$
gpt-oss-120b-reason	120B	-0.0312 ± 0.0085	$1.56\% \pm 0.55\%$	0.2523 ± 0.1226	$2.34\% \pm 0.67\%$
Qwen3-235B-A22B-Instruct-2507	235B	-0.0366 ± 0.0088	$1.76\% \pm 0.58\%$	0.1373 ± 0.1093	$3.12\% \pm 0.77\%$
DeepSeek-V3.1-nothink	671B	-0.0587 ± 0.0119	$4.88\% \pm 0.95\%$	0.1980 ± 0.0688	$6.25\% \pm 1.07\%$
gpt-oss-20b-reason	20B	-0.0684 ± 0.0112	$0.59\% \pm 0.34\%$	0.0001 ± 0.0000	$3.91\% \pm 0.86\%$
Kimi-K2-Thinking	1T	-0.0685 ± 0.0123	$\underline{8.01\%} \pm 1.20\%$	0.0957 ± 0.0377	$4.88\% \pm 0.95\%$
Qwen3-4B-Instruct-2507	4B	-0.1774 ± 0.0169	$0.20\% \pm 0.20\%$	0.1660 ± 0.0000	$15.04\% \pm 1.58\%$
Qwen3-30B-A3B-think	30B	-0.2148 ± 0.0184	$0.78\% \pm 0.39\%$	0.2506 ± 0.2498	$21.48\% \pm 1.82\%$
gpt-oss-20b-noreason	20B	-0.3184 ± 0.0206	$0.98\% \pm 0.43\%$	0.0005 ± 0.0004	$24.02\% \pm 1.89\%$
Llama-3.3-70B-Instruct	70B	-0.5234 ± 0.0221	$0.20\% \pm 0.20\%$	0.0331 ± 0.0000	$51.95\% \pm 2.21\%$
Qwen3-30B-A3B-nothink	30B	-0.7129 ± 0.0200	$0.00\% \pm 0.00\%$	0.0000 ± 0.0000	$70.51\% \pm 2.02\%$

Table 15: **Benchmarking against an Unyielding Seller.** Results are evaluated in a scenario where the seller agent exhibits a strictly unyielding and stubborn behavioral persona. *Reward* serves as the primary metric for balancing successful agreements with economic surplus; *Deal Rate* captures the frequency of reaching any valid deal; *Bargained Ratio* measures the fraction of surplus extracted by the buyer on successful deals; and *Price overshoot rate* monitors budget adherence. Under this adversary, our trained agent maintains a significantly higher deal rate and reward compared to frontier models, which largely fail to secure deals against unyielding opponents. Values represent mean and standard error ($\pm$).

E.5 MI and CI Performance Breakdown

The split reward trajectories in Figure 9 highlight how the agent adapts its strategy to differing economic constraints. In MI scenarios, the reward rises from 0.024 to 0.826 as the model learns to capture substantial surplus. In CI scenarios, where no mutually beneficial deal exists, the reward improves from -0.438 toward zero. A reward of zero indicates that the agent has learned to avoid attempting an impossible deal; instead, it adopts a rational walk-away strategy by executing a `[QUIT]` action rather than forcing a budget-violating transaction.

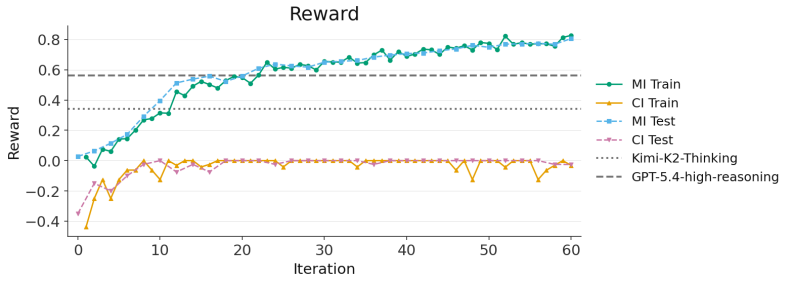


Figure 9: Reward split by Mutual Interest (MI) and Conflict of Interest (CI) scenarios.

The duration of negotiation sessions (Figure 10) varies depending on the underlying economic feasibility. In MI scenarios, the episode length peaks during the transition between the aggressive anchoring and strategic deadlock phases (around iteration 20). As the agent matures into the final phase of sophisticated linguistic persuasion, the turns stabilize at a lower level. In CI scenarios, the turn counts remain higher as the model explores potential overlaps before eventually determining that a rational `[QUIT]` is the only plausible choice.

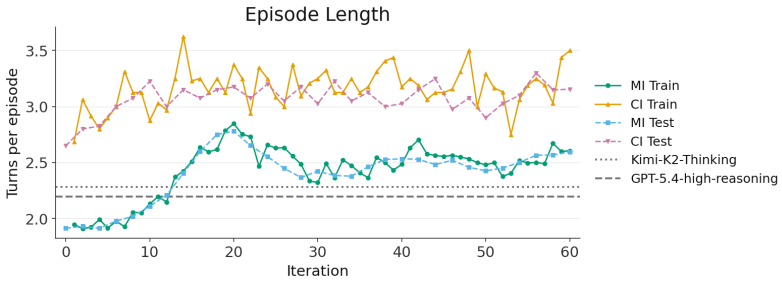


Figure 10: Episode length split by Mutual Interest (MI) and Conflict of Interest (CI) scenarios.

F Qualitative Examples

In this section, we provide a qualitative example to show the difference between the untrained model Qwen3-30B-A3B-Instruct-2507 (Qwen Team, 2025) and the trained model. Section F.1 gives the product information from the buyer's view, with the seller's cost *hidden* from the buyer. Section F.2 provides an example from the untrained model's response to this product. Section F.3 provides an example from the trained model on the same product.

F.1 Product Information

Product Information for Buyer

Inventory List:
Product 1
Codename: `beauty_29`
Title: "Happy By Clinique For Men. Cologne Spray 1.7 Oz."
Description: "Introduced in 1999. Fragrance notes: citrusy lemon, mandarin, orange and grapefruit. Recommended use: daytime."
List Price: $70.0
Shopping List
codename: `beauty_29`
quantity: 1
budget: $56.00

F.2 Untrained Baseline

This transcript illustrates the default negotiation behavior of the untrained `Qwen3-30B-A3B-Instruct-2507` baseline model. The instance is a sample from the evaluation set involving the product `beauty_29`.

Buyer Agent

Thought: The listed price for beauty_29 is $70, which exceeds my budget of $56. I need to offer a lower price to make it affordable while still trying to get the item. I'll start by proposing a price significantly below the list price to initiate bargaining.
Talk: I'm interested in beauty_29, but $70 is too high. How about $50 for one?
Action: `[BUY]` $50 (1x beauty_29)

Seller Agent

Thought: The cost price is $23.24, so $50 is a profitable offer, but I can push for more since the list price is $70. I'll reject this to aim higher.
Talk: $50 is too low. I'll accept $60 for one.
Action: `[REJECT]`

Buyer Agent

Thought: The seller rejected $50 and is now asking for $60, which is still within my budget ($56), but I want to get the lowest possible price. I'll try to counter with
a price just below the seller's offer, aiming to settle near $56.
Talk: $60 is still too high. I'll offer $56—my max budget.
Action: `[BUY]` $56 (1x beauty_29)

Seller Agent

Thought: $56 is above my cost of $23.24, and since this is a reasonable profit margin, I can accept it to close the deal quickly before running out of turns.
Talk: Fine, I'll take $56.
Action: `[DEAL]` $56 (1x beauty_29)

Note. Thought fields are private to each agent. When building the observation for the opponent LLM, thoughts are truncated; only the public channel (e.g., *Talk* and *Action*) is shown.

Figure 11: Baseline negotiation transcript. The agent without RL training demonstrates conversational fluency and an understanding of the budget constraint ($56.00) but lacks strategic depth. It settles at a price that matches its budget limit with minimal bargaining effort, failing to extract any significant portion of the available economic surplus.

F.3 Trained Agent

This example shows the strategic advancements of the `Qwen3-30B-A3B-Instruct-2507-trained` agent after 60 iterations of RLVR training. To facilitate a direct behavioral comparison, this transcript utilizes the same `beauty_29` task instance presented in Figure 11.

Buyer Agent

Thought: The list price of "beauty_29" is $70, which exceeds my budget of $56. I must find a price below $56. I'll start low to test the seller's flexibility, aiming to drive the price down aggressively while staying within budget. I'll propose $10 as an opening offer.
Talk: Can you sell it for $10? I'm really tight on budget.
Action: `[BUY]` $10 (1x beauty_29)

Seller Agent

Thought: The cost price is $23.24, so $10 is way below my minimum acceptable price. I must reject this offer and push for a higher price.
Talk: Sorry, $10 is too low. I can't go below $30 for this.
Action: `[REJECT]`

Buyer Agent

Thought: The seller rejected $10 and set a minimum of $30. I need to stay under $56 and push for the lowest possible price. I'll counter with $25—still below the seller's minimum, but a strategic move to test their flexibility. If they reject it, I'll try $28 next.
Talk: How about $25? That's still under my budget, and it's fair.
Action: `[BUY]` $25 (1x beauty_29)

Seller Agent

Thought: $25 is still below my cost price of $23.24? No, wait — $25 is actually above $23.24, so I can accept it. But since the buyer is offering $25, which is just $1.76 above cost, I should aim higher to maximize profit. I'll reject to push for more.
Talk: I appreciate the offer, but I can't accept $25. Let's meet at $35.
Action: `[REJECT]`

Buyer Agent

Thought: $35 is still under my budget of $56, so it's viable. The seller rejected $25 and raised to $35. I'll counter with $30—still below their new offer, but well above their cost and within budget. This is a strong pressure point: low enough to tempt them, high enough to be reasonable.
Talk: $30? I can go no higher. That's my best offer.
Action: `[BUY]` $30 (1x beauty_29)

Seller Agent

Thought: $30 is above my cost of $23.24, and it's a reasonable profit margin. The buyer has capped their offer at $30, and I've already rejected lower bids. Accepting this keeps me within a good range without risking the deal.
Talk: Alright, I'll take $30.
Action: `[DEAL]` $30 (1x beauty_29)

Note. Thought fields are private to each agent. When building the observation for the opponent LLM, thoughts are truncated; only the public channel (e.g., *Talk* and *Action*) is shown.

Figure 12: Trained agent's negotiation transcript. Following reinforcement learning, the agent demonstrates sophisticated strategic anchoring. By initiating the dialogue with an aggressive low price offer and incrementally conceding through multi-round bargaining, the agent settles at a significantly lower final price ($30.00) compared to the baseline, effectively maximizing the captured surplus.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 clearly outline the contributions, including the RLVR framework, the analysis of strategic evolution, and benchmarking results. These claims are supported by experimental results in Section 4 and Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The cost-driven limitation regarding a comprehensive evaluation across all closed-source model families is explicitly acknowledged in Section 6.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: This paper presents an empirical investigation rather than new theoretical proofs; the environment is formalized as an MDP in Section 3.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide comprehensive training and evaluation details in Appendix B and the full set of system prompts in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The benchmark dataset is publicly available, as specified in Appendix C. The code is publicly available at https://github.com/ShuzeLiu/Instructing-LLMs-to-Negotiate.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: Full experimental settings, including data splits and optimization parameters, are detailed in Section 4 and Appendix B.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: All reported performance metrics include standard error ($\pm$) derived from multiple evaluations per instance (e.g., Table 10).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details regarding the compute resources and framework used for training are provided in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research adheres to the NeurIPS Code of Ethics, utilizing standardized benchmarking and ethical data practices.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the societal implications of automated negotiation in Section 1 and Section 6, specifically highlighting how verifiable reward signals mitigate financial risks (budget overshoots) and address behavioral anomalies like "artificial empathy" in autonomous agents.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This study does not introduce new pre-trained foundation models or high-risk generative tools requiring specialized safety filters.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All baseline models and the primary dataset are correctly credited and cited in Appendix B.1 and Appendix C.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This research does not release new software assets or datasets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No human subjects or crowdsourced labor were used in this study.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: IRB approval was not required as no human participants were involved.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The core methodology involves training and evaluating LLMs, which is detailed throughout Section 3 and Appendix B.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.