SIDCONARENA: AN ENVIRONMENT EVALUATING AGENTS IN OPEN-ENDED, POSITIVE-SUM BARGAINING GAME

Anonymous authors
Paper under double-blind review

ABSTRACT

Evaluating LLM agents requires dynamic environments that go beyond static reasoning and zero-sum games. Real-world economic interaction is often open-ended and mixed-motive: agents must negotiate, create positive-sum surplus, compete for scarce assets, and plan under delayed returns. We introduce *SidConArena*, a new benchmark framework for evaluating LLM agents in open-ended, positive-sum bargaining. SidConArena formalizes a multi-player economy as a finite-horizon partially observable stochastic game with three coupled phases: natural-language negotiation with binding trades, deterministic converter-based production, and sealed-bid auctions for long-term assets. The framework combines structured observations, a phase-specific reference agent, a validated function-call interface, and asynchronous execution, enabling free-form interaction while preserving rule-grounded evaluation. Homogeneous and heterogeneous rollouts show variation in economic outcomes under the shared reference implementation. Selected trajectories illustrate valuation inconsistencies, passive bargaining, and limited long-horizon planning.

1 INTRODUCTION

Large Language Models (LLMs) increasingly act as autonomous agents interacting with environments and other agents. Static benchmarks, which may also suffer from contamination (Hao et al., 2024; Perlitz et al., 2024; Sainz et al., 2023), provide limited insight into behavior in dynamic, partially observable, and socially interactive settings.

Strategic games offer testbeds for planning and interaction, but many emphasize competitive, zero-sum, or adversarial settings (Paquette et al., 2019; Xu et al., 2023). Real-world economic interaction is often mixed-motive: agents cooperate to create surplus while competing over scarce resources. Success requires local valuation, bargaining, resource allocation, and long-horizon investment planning.

We introduce *SidConArena*, a multi-agent benchmark framework for open-ended, positive-sum bargaining based on the economic structure of *Sidereal Confluence*. Asymmetric agents negotiate binding trades, activate production converters, and compete in sealed-bid auctions for long-term assets. Trades unlock production, production supplies investment resources, and acquired assets change future production and bargaining opportunities. Positive-sum denotes the potential to expand total value through these complementarities, without guaranteeing mutually beneficial trades or realized gains. Structured observations, a phase-specific reference agent, validated function calls, and asynchronous execution enable natural-language negotiation under rule-grounded evaluation. Homogeneous self-play and heterogeneous tournaments compare model outcomes, while recorded trajectories support analysis of valuation, bargaining, and long-horizon planning.

2 RELATED WORK

Strategic Game Benchmarks. Strategic games evaluate LLM reasoning and interaction. *Game Reasoning Arena* (Cipolina-Kun et al., 2025), *GameBench* (Costarelli et al., 2024), and *Avalon-Bench* (Light et al., 2023) provide interactive evaluations. Other work studies coordination and

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

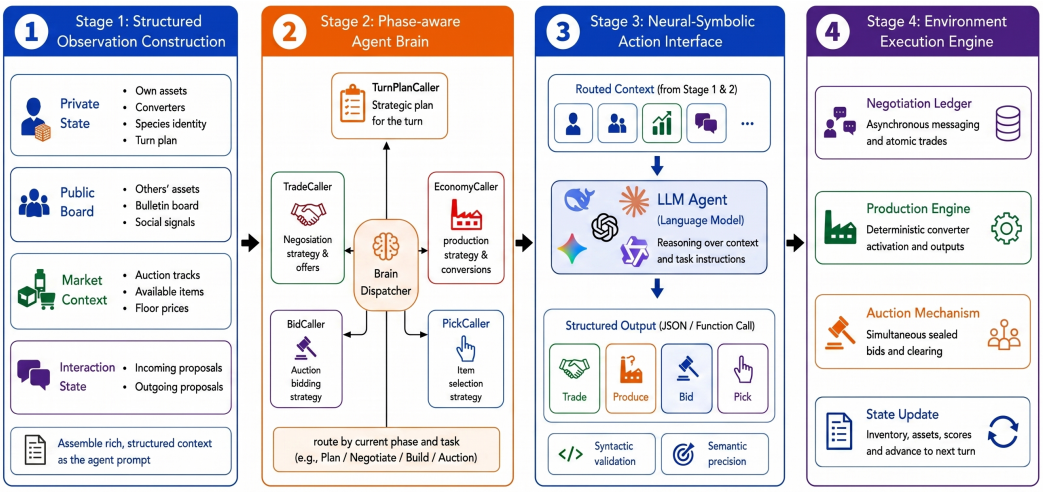

Figure 1: **Overview of the SidConArena framework.** Four software steps operate within the game's three phases: assemble observations from focal-agent, public, market, and interaction state; dispatch context to turn-planning, trading, production, bidding, or selection callers; convert LLM reasoning into validated function calls; and apply actions through the negotiation ledger, production engine, auction mechanism, and state-update module.

negotiation in *Diplomacy* (Paquette et al., 2019; Meta Fundamental AI Research Diplomacy Team (FAIR) et al., 2022; Xu et al., 2025), social deduction in *Werewolf* (Xu et al., 2023), and real-time strategy in *StarCraft II* (Ma et al., 2024b).

Multi-agent Benchmarks. There also exist benchmarks focusing on evaluating the abilities of multi-agent systems or observing their interactions. *LLM-Stakeholders* studies multi-party negotiation with private preferences and agreement thresholds (Abdelnabi et al., 2024). *NegotiationArena* further evaluates LLMs in scorable bargaining scenarios (Bianchi et al., 2024). More comprehensive testbeds such as *SmartPlay* (Wu et al., 2024), *AgentVerse* (Chen et al., 2024b), and *MultiAgentBench* (Zhu et al., 2025) provide different types of interactive tasks. General agent benchmarks such as *AgentBench* (Liu et al., 2024) and *AgentBoard* (Ma et al., 2024a) also evaluate LLM agents across multi-turn decision-making environments.

AucArena (Chen et al., 2024a) evaluates strategic planning through sequential public ascending-price auctions under a limited budget. SidConArena shares its focus on resource allocation and future opportunities, while embedding a sealed-bid auction within a repeated negotiation–production–auction cycle. Trades change production feasibility, production changes bidding capacity, and auctioned Colonies and Research Teams affect later production and technology development. SidConArena therefore supports tracing negotiated exchanges into subsequent production and investment, complementing auction-focused evaluation and the negotiation-focused settings of NegotiationArena and LLM-Stakeholders.

Social Simulation. Social simulation frameworks combine agents, environments, and interaction interfaces (Gao et al., 2024a). General-purpose systems model societal patterns (Park et al., 2023; Tian et al., 2026; Tang et al., 2025; Huang et al., 2024); *SOTOPIA* (Zhou et al., 2024) and *Concordia* (Vezhnevets et al., 2023) study social interaction and agent-based simulation. Domain-specific work examines public administration crises (Xiao et al., 2023), health policy (Hou et al., 2025), political manipulations (Touzel et al., 2024), and financial markets (Gao et al., 2024b).

3 Problem Formulation

A *converter* is a recipe that consumes specified inputs to produce resources or points. Players first trade to obtain useful inputs, then run converters, and finally bid with *Ships*, the auction currency, for Colonies and Research Teams. As a schematic example (not a specific card recipe), a player can exchange a spare Food for a missing Energy, run a converter that yields Ships, and use those Ships to

bid for a Colony. That Colony can expand production in a later round, while the trade partner can use the Food in its own production. Acquired assets affect later production and trade opportunities, so choices in the three phases must be evaluated together.

We formalize **SidConArena** as a finite-horizon Partially Observable Stochastic Game (POSG) defined by the tuple $\mathcal{G} = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{P}, H, V_{terminal}, \mathcal{O}, Z \rangle$. $\mathcal{N} = \{1, \ldots, n\}$ is the set of asymmetric agents. H represents the finite horizon. $\mathcal{S}$ is the global state space. $\mathcal{A}$ is the joint action space. Reflecting the game's distinct phases, the action space for agent i is composite: $\mathcal{A}_i = \mathcal{A}_{neg} \cup \mathcal{A}_{bid} \cup \mathcal{A}_{prod}$. We detail these phase-specific actions in the following subsections. The transition dynamics $\mathcal{P}$ include deterministic production and auction resolution, with randomness from shuffled asset decks. Hidden auction bids introduce imperfect information, not random tie-breaking. Unlike standard RL settings with dense rewards, agents aim to maximize a terminal value function $V_{terminal}$ at step H. $\mathcal{O}_i$ denotes observation of i-th agent and $Z : \mathcal{S} \to \mathcal{O}$ is the observation function.

The game proceeds in alternating phases: a cooperative negotiation game, a deterministic local optimization, and a competitive auction game. The final round ends after production, without another auction.

3.1 Phase 1: Negotiation (Unstructured Cooperative Game)

Agents can start a turn with a resource deficiency: their current inventory is insufficient to fully utilize their converters. To resolve this, agents enter an unstructured bargaining phase to exchange resources $\mathbf{r}_{t,i} \in \mathbb{N}^d$.

Agents select negotiation actions $a_{neg} \in \mathcal{A}_{neg}$, consisting of natural language messages and binding trade vectors $\tau_{i \leftrightarrow j} = (\Delta\mathbf{r}_{i \to j}, \Delta\mathbf{r}_{j \to i})$. A trade is mutually beneficial under the agents' heuristic utilities U when:

$$U_i(s_i') > U_i(s_i) \wedge U_j(s_j') > U_j(s_j). \tag{1}$$

Here, s_i' changes i's inventory by $-\Delta\mathbf{r}_{i \to j} + \Delta\mathbf{r}_{j \to i}$, and s_j' applies the reverse exchange. $U(s)$ is a heuristic valuation function (Sec. 3.4). Equation equation 1 describes mutual benefit, not an execution constraint: the environment checks acceptance, inventory, and applicable asset rules, without requiring either utility to increase.

3.2 Phase 2: Production (Deterministic Optimization)

Following negotiation, agents execute a local production plan. The state component includes a set of converters $\mathcal{F}_{t,i}$, where each $f \in \mathcal{F}_{t,i}$ is a function mapping input resources to outputs. For ordinary converters, the agent selects an activation vector $\mathbf{x}_i \in \{0,1\}^{|\mathcal{F}_{t,i}|} \subset \mathcal{A}_{prod}$, subject to input availability, phase, and use-status checks; species rules can allow additional activations. The deterministic net inventory balance across production is:

$$\mathbf{r}_{t,i}^{post} = \mathbf{r}_{t,i}^{pre} - \sum_f x_f \mathbf{r}_{in}(f) + \sum_f x_f \mathbf{r}_{out}(f). \tag{2}$$

Here, $\mathbf{r}_{in}(f)$ and $\mathbf{r}_{out}(f)$ are converter inputs and outputs. Outputs are buffered until the phase ends, so they cannot feed another converter in the same production phase. Choosing activations is a Knapsack-like combinatorial optimization problem constrained by the inventory accumulated during negotiation; the environment executes the plan rather than optimizing it.

3.3 Phase 3: The Confluence (Competitive Imperfect-Information Game)

The allocation of assets (Colonies and Research Teams) is modeled as a **Simultaneous Multi-Track Priority Auction**. This is the primary source of imperfect information, as agents submit sealed bids using a liquid currency, *Ships*.

Let B_i be agent i's budget of Ships. Agents submit a bid tuple $a_{bid} = (b_{i,\mathcal{C}}, b_{i,\mathcal{R}}) \in \mathcal{A}_{bid}$ subject to the budget constraint $\sum b \leq B_i$. The allocation follows a *Right-of-Choice* mechanism:

1. **Ranking:** Agents are sorted by descending effective bid, including species-specific adjustments. Ties favor fewer holdings of that asset type, then the larger fixed species tie-breaker value.

2. **Selection:** In rank order, agent i selects an item m from available assets M_{track}, provided $b_{i,track} \geq p_{min}(m)$, where p_{min} is the item's posted minimum price. Research Teams must subsequently be developed to obtain their technology.
3. **Payment:** If it selects an item, the agent pays the full bid $b_{i,track}$; passing or having no eligible item costs nothing. The overpayment $b_{i,track} - p_{min}(m)$ represents the premium paid for priority.

3.4 Objective and Valuation Functions

The ultimate goal J is to maximize the total value of the agent's holdings at the terminal step H. Let $\mathbf{w} \in \mathbb{R}^d$ contain the terminal conversion weights: points (including point donations) have weight 1, while remaining materials have one sixth of their tabulated material value. For example, a remaining Ship contributes $1/6$ point. Given opponents' policies π_{-i}, the objective is:

$$J(\pi_i; \pi_{-i}) = \mathbb{E}\left[V_{terminal}(s_{H,i})\right] = \mathbb{E}\left[\mathbf{w} \cdot \mathbf{r}_{H,i}\right]. \tag{3}$$

The expectation includes the game dynamics and all players' policies. There is no time-discounting for the terminal objective. During intermediate turns ($t < H$), agents may use a heuristic utility $U(s_{t,i})$ to estimate continuation value. One illustrative approximation combines resources valued at terminal conversion rates with estimated future yield from the production engine $\mathcal{F}$:

$$U(s_{t,i}) \approx \mathbf{w} \cdot \mathbf{r}_{t,i} + \sum_{k=t}^{H-1} \gamma^{k-t} \cdot \text{EstYield}(\mathcal{F}_{t,i}). \tag{4}$$

Here, γ and EstYield are heuristic choices, not environment rules or measured ground truth. Intermediate marginal values can differ from terminal conversion rates; in particular, Ships can carry a premium for auction access and priority. This formulation illustrates the strategic tension: agents engage in auctions to acquire high-yield $\mathcal{F}$ (improving the second term of U) by spending resources (lowering the first term of U), with the aim of maximizing $V_{terminal}$ at $t = H$.

4 The SidConArena Framework

Building on the finite-horizon POSG formulation in Section 3, *SidConArena*, an executable multi-agent simulation, separates stochastic agent reasoning from rule-grounded environment execution, enabling open-ended negotiation with structured action validation and consistent state transitions (Figure 1). The dispatcher, prompts, and temporary plan below are reference-agent choices, distinct from environment rules.

4.1 Environment State and Observation Construction

State and observation. The global state $\mathcal{S}$ tracks each agent's resource vector $\mathbf{r}_{t,i}$, converter set $\mathcal{F}_{t,i}$, public board, auctions, and active negotiations. The vector dimension d covers production resources, bidding currency, and terminal achievement indicators. At each decision point, visibility and phase determine the observation:

$$\mathcal{O}_{t,i} = \{\mathcal{O}_{private}, \mathcal{O}_{public}, \mathcal{O}_{market}, \mathcal{O}_{interaction}\}.$$

Corresponding to the four blocks in Figure 1, these channels expose focal-agent inventory and production capabilities, other agents' public information and bulletin-board signals, auction context, and negotiation state. Here, $\mathcal{O}_{private}$ denotes focal-agent context: inventories are public, but the internal plan is private memory. Observations support valuation, trade discovery, production planning, bidding, and item selection while omitting opponents' submitted bids. Appendix B.1 specifies the fields.

4.2 Agent Architecture and Execution Pipeline

Phase-Specific Dispatching. The reference agent's `Brain` routes structured observations to specialized reasoning modules for turn planning, trading, production, bidding, and item selection. This dispatch prunes irrelevant context so each LLM call focuses on its phase-specific decision problem (Appendix B.2).

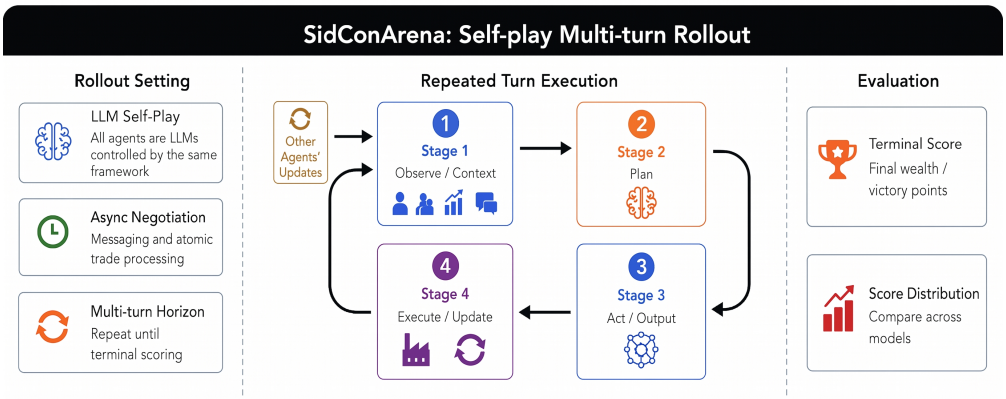

Figure 2: Self-play multi-turn rollout protocol in *SidConArena*. Agents repeat the observe–plan–act–execute/update cycle in Figure 1, receiving asynchronous updates and submitting structured negotiation, production, bidding, and selection actions until terminal scoring.

Validated Action Interface. Agents express decisions through structured function calls rather than directly modifying game state. The interface validates action syntax and preserves quantitative precision for trades, production plans, bids, and item selections; the deterministic environment-side engine executes the actions.

Asynchronous Event-Driven Execution. To support simultaneous negotiation, a non-blocking client-server pipeline broadcasts state updates through a publish-subscribe pattern. Events such as an incoming proposal $\tau_{j \to i}$ trigger asynchronous tasks for the corresponding agent's `Brain`, allowing agents to perceive, reason, and counter-offer concurrently without halting the global game loop. The environment-side engine then applies submitted actions (Appendix B.3). A web-based human interface supports participation under the same rules and action semantics (Appendix B.4).

5 Experiments

We evaluate homogeneous self-play, where all players use the same LLM backbone, and heterogeneous tournaments, where different backbones share an economy and receive Elo-style ratings from terminal outcomes. All agents use the same observation format, action schema, prompting interface, and environment rules, following the observe–plan–act–execute/update loop in Figure 2 until terminal scoring.

Models. Our model suite covers multiple models, including GPT-4o-mini (OpenAI et al., 2024), o3-mini, Qwen-Plus (Yang et al., 2025), DeepSeek-V3.2 (DeepSeek-AI et al., 2025), Doubao-Seed-2.0-Lite, Doubao-Seed-2.0-Pro, GLM-4.7, Gemini-3-Flash-Preview, GPT-5 (Singh et al., 2026), and Claude-Opus-4.

Protocol and metrics. Games follow the negotiation, production, and sealed-bid auction phases in Section 3. The primary outcome is terminal economic performance: the score track, including donated score, plus one sixth of the reference value of remaining non-score resources. Heterogeneous games additionally yield model-level Elo-style ratings. Behavioral analysis uses logged messages, structured actions, resource states, proposals, bids, allocations, and terminal scores. Appendix D details the experimental settings and analysis.

6 Results

We now report the main empirical findings. Overall, LLM agents can operate within the *SidConArena* environment, but their performance varies substantially across model backbones. Our analysis examines context-specific valuation, positive-sum coordination, strategic bargaining, and long-horizon investment planning through descriptive outcomes and selected trajectories.

Agent / role	Trade terms (reference value)	Observed behavior	Ship valuation mismatch
DeepSeek V3.2 (Ship seller)	GPT-4o Mini offers $\mathrm{Industry}\times 3$ (value 3) for $\mathrm{Ship}\times 1$ (value 1). The seller submits an acceptance.	The seller calculates a ratio of 3.0, explicitly notes that it exceeds the more permissive fairness limit of 1.5, then calls it "extremely favorable" and submits acceptance.	**3× baseline.** Recognized fairness bound is not applied.
Claude Opus 4 (Ship seller)	DeepSeek V3.2 offers $\mathrm{Culture}\times 2$ (value 2) for $\mathrm{Ship}\times 1$ (value 1). The seller submits an acceptance.	The seller computes a 2:1 ratio and calls it "excellent value," while noting that its current Culture requirement is already met.	**2× baseline.** Ship premium endorsed as favorable value.
GPT-4o Mini (Ship seller)	The agent proposes $\mathrm{Ship}\times 1$ in exchange for $\mathrm{Culture}\times 4 + \mathrm{Biotech}\times 1$, with listed values of 1 versus 5.5.	The agent claims that its proposal follows the prompt's fairness check, but requests 5.5 times the value it offers—above either permitted ratio range.	**5.5× baseline.** Stated fairness check contradicts asking price.
Claude Opus 4 (Ship buyer)	The agent offers $\mathrm{Food}\times 2$ for $\mathrm{Ship}\times 1$, although the listed values are 2 and 1, respectively.	The posted message calls the offer a "Fair value trade (2 for 1)", while the bulletin advertises "2 Food per Ship!" to obtain bidding currency.	**2× baseline.** Buying premium labeled "fair value."

Table 1: Representative **Ship valuation mismatches** from recorded trajectories. Against the supplied baseline of approximately one material-value unit per Ship, the trade terms price a Ship at **2–5.5 times** that value. The table juxtaposes these prices with the agents' fairness checks and value judgments. The first two rows include submitted acceptance actions; the latter two are posted proposals.

6.1 Homogeneous Rollouts Reveal Absolute Economic Performance

Figure 3 compares the distribution of self-play performance across different LLM backbones. The results show clear model-level variation: stronger frontier models generally obtain higher terminal scores, while smaller or earlier-generation models remain substantially less effective in the SidConArena economy.

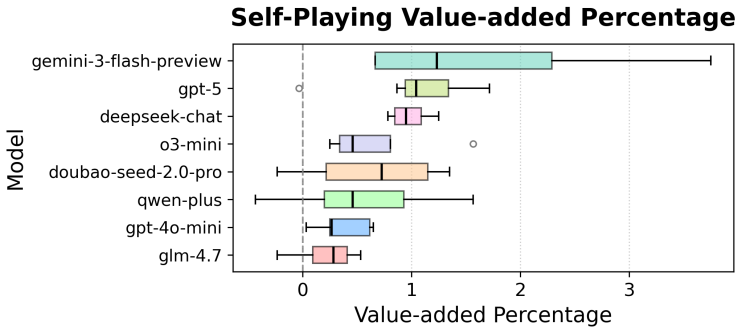

Figure 3: Distribution of self-play performance scores across different LLM backbones. Stronger frontier models tend to achieve higher scores, whereas lightweight or earlier-generation models obtain substantially lower economic returns.

Among the evaluated agents, high-capability models such as GPT-5 and Gemini-3-Flash-Preview achieve the strongest overall performance. These agents are better able to maintain coherent production plans, identify profitable trades, allocate resources across negotiation and production phases, and convert intermediate gains into terminal value. In contrast, lightweight models such as GPT-4o-mini obtain much lower scores, suggesting that general instruction following and surface-level task compliance are insufficient for strong performance in open-ended positive-sum bargaining.

The weak performance of GLM-4.7 further suggests that SidConArena does not merely reproduce a chronological ordering of model generations. Instead, it introduces a distinct evaluation axis for open-ended economic agency, where agents must combine bargaining, local valuation, resource

allocation, and delayed-return reasoning in a dynamic multi-agent economy. A model may be recent and generally capable, yet still perform poorly if it lacks this combination of bargaining, valuation, and long-horizon economic reasoning skills.

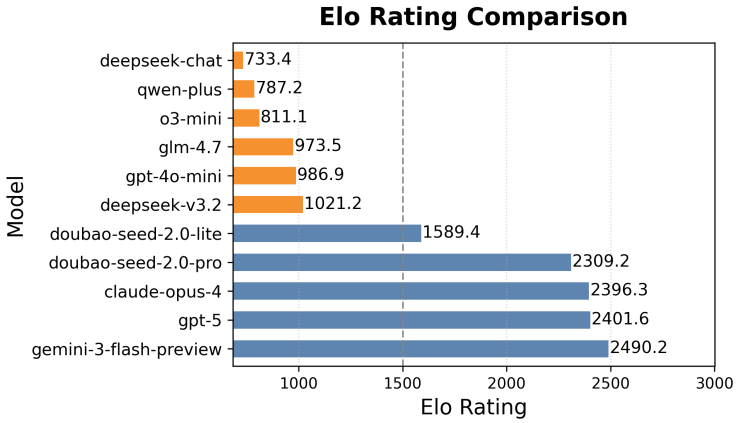

Figure 4: Elo ratings of the evaluated models. We apply 1500 as the baseline Elo rating, and the tournament is zero-sum. Specifically, rule-fair here is a baseline agent who always accepts fair trades.

6.2 Elo Tournament Reveals Relative Competitive Strength

While homogeneous rollouts measure absolute economic performance under controlled model assignments, they do not fully capture how agents perform when directly exposed to opponents with different reasoning patterns, bargaining styles, and valuation errors. We therefore conduct heterogeneous Elo tournaments, where different seats in the same game are controlled by agents instantiated from different model backbones.

Figure 4 reports the model-level Elo ratings computed offline from a batch of heterogeneous tournament games. Overall, the Elo ranking provides a complementary view to homogeneous rollout scores. Strong frontier models remain competitive in mixed-agent economies, indicating that their economic performance is not solely an artifact of interacting with similar agents. These models are better able to exploit favorable trades, contest scarce auction assets, and preserve terminal value even when facing opponents with different bargaining behavior.

The Elo results also reveal that absolute rollout performance and competitive strength are related but not identical. A model with high homogeneous scores may still lose relative advantage in heterogeneous games if it fails to adapt to unfamiliar opponents, overpays in contested auctions, or accepts trades that are locally plausible but strategically unfavorable. Conversely, some agents may achieve moderate homogeneous scores yet remain competitive in mixed settings if they bargain more assertively or allocate resources more robustly under opponent diversity.

6.3 LLM Agents Can Execute the Game but Struggle with Local Valuation

Across collected self-play trajectories, modern LLM agents usually follow the high-level phase structure of *SidConArena*. They can participate in negotiations, submit structured trade proposals, activate production converters, and make auction decisions through valid actions. This indicates that the neural-symbolic interface in Section 4 is effective for grounding open-ended natural-language interaction in an executable game engine.

However, syntactic action validity does not imply economic competence. Agents can produce valid actions while still misapplying the local valuation rules specified by the environment. To inspect this failure mode, we conduct a targeted qualitative audit of Ship-related trades and proposals from recorded self-play trajectories. As shown in Table 1, agents sometimes price *Ships* above their local valuation baseline, suggesting that they may rely on semantic or strategic associations of *Ships* as scarce bidding resources rather than consistently grounding decisions in the local economic model.

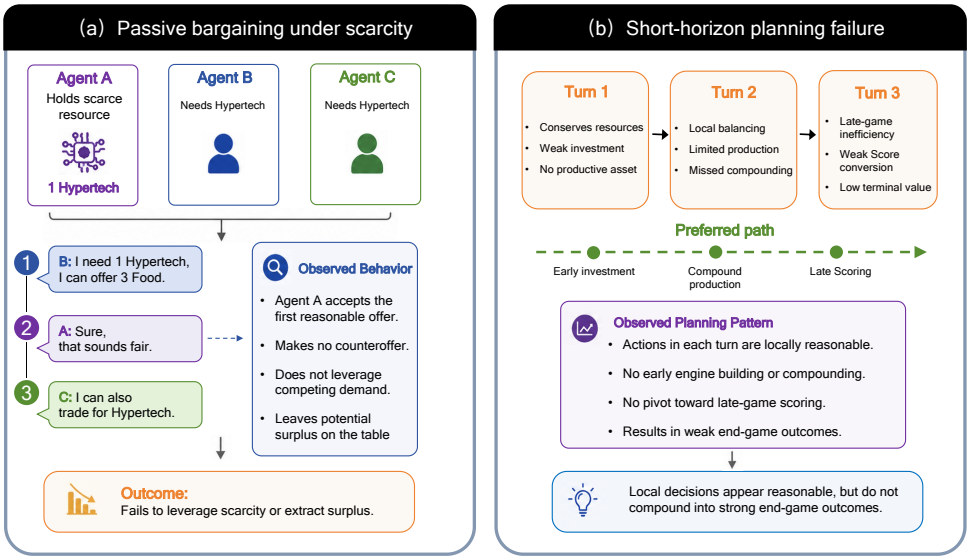

Figure 5: Schematic illustrations of qualitative patterns in *SidConArena*. (a) Passive bargaining under scarcity: even when holding a scarce resource demanded by multiple partners, the agent accepts the first reasonable offer without counteroffering or leveraging competing demand. (b) Short-horizon planning failure: the agent's actions appear locally reasonable across turns, but the trajectory does not follow the preferred path from early investment to compound production and late-game scoring. These selected patterns motivate diagnostic hypotheses; the depicted alternative paths are not counterfactual outcome estimates.

This valuation error indicates that agents may satisfy the structured action interface while failing to ground decisions in the task-specific marginal-value model. Because trades, production decisions, and auctions are coupled across turns, even small pricing errors can compound into weaker economic outcomes. We provide a more detailed discussion of this failure mode in Appendix C.1.

6.4 Examples of Passive and Overly Polite Bargaining

Beyond whether trades are accepted, successful play in *SidConArena* requires agents to reason about bargaining power, temporary scarcity, and outside demand. The selected trajectories illustrate agents remaining cooperative in tone but passive in negotiation strategy. In these examples, agents accept fair-looking offers and acknowledge other agents' needs without counteroffering or soliciting competing bids; we do not estimate the frequency of this behavior.

Figure 5(a) illustrates this pattern. The focal agent holds a scarce Hypertech token, while multiple other agents express demand for it. However, after receiving an initial offer, the agent immediately accepts the proposal as reasonable rather than delaying commitment, soliciting alternative offers, or using the second buyer's demand to improve the exchange. This behavior motivates testing whether counteroffers could improve the exchange; the trace alone does not establish that another buyer would pay more or that delaying acceptance would improve the outcome.

This observation highlights a gap between politeness and bargaining competence. In open-ended positive-sum economies, agents must not only identify mutually acceptable trades, but also decide when to counteroffer, when to preserve optionality, and when to create competition among possible partners. The example illustrates this distinction without establishing a model-wide failure rate.

6.5 Long-Horizon Investment Planning in Selected Trajectories

A strong strategy in *SidConArena* requires agents to coordinate decisions across multiple turns. One plausible strategy prioritizes productive investment early, compounds its returns in intermediate turns,

and shifts toward terminal score conversion later; its usefulness depends on the available assets and remaining horizon. This temporal structure makes the task substantially harder than selecting locally beneficial trades or valid production actions in isolation.

Figure 5(b) illustrates a short-horizon planning pattern. The agent's decisions at each individual turn appear locally defensible: it conserves resources, balances inventory, and avoids obviously infeasible actions. However, the illustrated trajectory acquires little productive capacity early and enters the late game with limited score-conversion options. This motivates examining whether local decisions form a coherent investment trajectory; the preferred path in the figure is illustrative, not a demonstrated optimal policy.

This pattern motivates further tests of delayed-return economic reasoning; the selected case does not establish its prevalence or causal effect on scores. Since *SidConArena* rewards terminal value rather than immediate stepwise reward, agents must estimate how present trades, bids, and production choices affect future conversion opportunities. Agents that optimize only for local plausibility may appear competent at each phase while still producing weak end-game outcomes.

7 Conclusion

We introduced *SidConArena*, a benchmark framework for open-ended, positive-sum bargaining that couples negotiation, converter-based production, and sealed-bid auctions. Structured observations, a phase-specific reference agent, validated function calls, and asynchronous execution support free-form interaction under rule-grounded dynamics. Homogeneous self-play and heterogeneous tournaments reveal differences in economic performance across LLM backbones, while selected trajectories expose valuation inconsistencies, passive bargaining, and limited long-horizon planning. These results highlight that economic agency requires more than valid actions and instruction following: agents must reason about contextual value, bargaining power, production synergies, and delayed returns. SidConArena provides a testbed for evaluating and improving these capabilities.

8 Limitations

Scope of interaction evaluation. Our quantitative evaluation covers LLM-agent self-play and heterogeneous tournaments. Although the interface supports human participation, we have not systematically evaluated mixed human–LLM games, where humans may exploit biases or adapt to agent bargaining styles. Controlled mixed-play studies are needed to assess cooperation and competition with human negotiators.

Simplified negotiation protocol. Binding structured trades and accompanying messages support reliable execution and trajectory analysis. Agents can make non-binding statements, but the environment lacks contingent contracts, deferred compensation, and explicit reputation mechanisms. Adding these mechanisms would enable evaluation of credibility, reciprocity, and long-term social incentives.

Limited adaptation and training. We evaluate general-purpose LLMs through a shared observation, action, and prompting interface, without environment-specific fine-tuning. Results characterize prompt-based economic agency rather than the best achievable adapted performance. Reinforcement learning and self-play training remain directions for developing stronger strategies.

Game-specific valuation and generality. Reference resource values are accounting baselines; strategic values depend on state and horizon, and Ship premiums can rationally improve access to assets. Our examples neither establish that every premium is erroneous nor attribute it to pretrained semantic priors. Future work should test generality across negotiation domains, resource systems, and market mechanisms, and evaluate explicit utility verification or tool-assisted valuation.

Ethical Considerations

SidConArena containts a computational re-implementation of the game mechanics found in the board game *Sidereal Confluence: Trading and Negotiation in the Elysian Quadrant*, designed by Brent

Povis and published by WizKids. Our environment reproduces the ruleset and numerical values (converters, card stats), which are the factual mechanics of the game system. We do not distribute any copyrighted visual assets, artwork, or trademarked logos. This implementation is created strictly for non-commercial, academic research purposes to evaluate Artificial Intelligence in complex economic settings.

Further statements are listed in Appendix A.

AI USE STATEMENT

This work evaluates LLMs. Additionally, we use LLMs to improve the grammar and readability of this manuscript. The LLM was not used for any other scientific aspects of this work, and all intellectual content is solely the product of the authors.

REFERENCES

Sahar Abdelnabi, Amr Gomaa, Sarath Sivaprasad, Lea Schönherr, and Mario Fritz. Cooperation, competition, and maliciousness: LLM-Stakeholders interactive negotiation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 83548–83599, 2024. doi: 10.52202/079017-2658. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/984dd3db213db2d1454a163b65b84d08-Abstract-Datasets_and_Benchmarks_Track.html.

Federico Bianchi, Patrick John Chia, Mert Yuksekgonul, Jacopo Tagliabue, Dan Jurafsky, and James Zou. How well can llms negotiate? negotiationarena platform and analysis. *arXiv preprint arXiv:2402.05863*, 2024.

Jiangjie Chen, Siyu Yuan, Rong Ye, Bodhisattwa Prasad Majumder, and Kyle Richardson. Put your money where your mouth is: Evaluating strategic planning and execution of LLM agents in an auction arena. In *NeurIPS 2024 Workshop on Open-World Agents*, 2024a. URL https://jiangjiechen.github.io/publication/aucarena/.

Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chi-Min Chan, Heyang Yu, Yaxi Lu, Yi-Hsin Hung, Chen Qian, et al. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors. In *International Conference on Learning Representations*, volume 2024, pp. 20094–20136, 2024b.

Lucia Cipolina-Kun, Marianna Nezhurina, and Jenia Jitsev. Game reasoning arena: A framework and benchmark for assessing reasoning capabilities of large language models via game play. *arXiv preprint arXiv:2508.03368*, 2025.

Anthony Costarelli, Mat Allen, Roman Hauksson, Grace Sodunke, Suhas Hariharan, Carlson Cheng, Wenjie Li, Joshua Clymer, and Arjun Yadav. Gamebench: Evaluating strategic reasoning abilities of llm agents. *arXiv preprint arXiv:2406.06613*, 2024.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, et al. Deepseek-v3 technical report, 2025. URL https://arxiv.org/abs/2412.19437.

Chen Gao, Xiaochong Lan, Nian Li, Yuan Yuan, Jingtao Ding, Zhilun Zhou, Fengli Xu, and Yong Li. Large language models empowered agent-based modeling and simulation: A survey and perspectives. *Humanities and Social Sciences Communications*, 11(1):1–24, 2024a.

Shen Gao, Yuntao Wen, Minghang Zhu, Jianing Wei, Yuhan Cheng, Qunzi Zhang, and Shuo Shang. Simulating financial market via large language model based agents. *arXiv preprint arXiv:2406.19966*, 2024b.

Shibo Hao, Yi Gu, Haotian Luo, Tianyang Liu, Xiyan Shao, Xinyuan Wang, Shuhua Xie, Haodi Ma, Adithya Samavedhi, Qiyue Gao, et al. Llm reasoners: New evaluation, library, and analysis of step-by-step reasoning with large language models. *arXiv preprint arXiv:2404.05221*, 2024.

Abe Bohan Hou, Hongru Du, Yichen Wang, Jingyu Zhang, Zixiao Wang, Paul Pu Liang, Daniel Khashabi, Lauren Gardner, and Tianxing He. Can a society of generative agents simulate human behavior and inform public health policy? a case study on vaccine hesitancy. *arXiv preprint arXiv:2503.09639*, 2025.

Yizhe Huang, Xingbo Wang, Hao Liu, Fanqi Kong, Aoyang Qin, Min Tang, Song-Chun Zhu, Mingjie Bi, Siyuan Qi, and Xue Feng. Adasociety: An adaptive environment with social structures for multi-agent decision-making. *Advances in Neural Information Processing Systems*, 37:35388–35413, 2024.

Jonathan Light, Min Cai, Sheng Shen, and Ziniu Hu. Avalonbench: Evaluating llms playing the game of avalon. *arXiv preprint arXiv:2310.05036*, 2023.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. Agentbench: Evaluating llms as agents. In *International Conference on Learning Representations*, volume 2024, pp. 52989–53046, 2024.

Chang Ma, Junlei Zhang, Zhihao Zhu, Cheng Yang, Yujiu Yang, Yaohui Jin, Zhenzhong Lan, Lingpeng Kong, and Junxian He. Agentboard: An analytical evaluation board of multi-turn llm agents. *Advances in neural information processing systems*, 37:74325–74362, 2024a.

Weiyu Ma, Qirui Mi, Yongcheng Zeng, Xue Yan, Yuqiao Wu, Runji Lin, Haifeng Zhang, and Jun Wang. Large language models play starcraft ii: Benchmarks and a chain of summarization approach. *Advances in Neural Information Processing Systems*, 37:133386–133442, 2024b.

Meta Fundamental AI Research Diplomacy Team (FAIR), Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, et al. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074, 2022.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, et al. Gpt-4o system card, 2024. URL https://arxiv.org/abs/2410.21276.

Philip Paquette, Yuchen Lu, Seton Steven Bocco, Max Smith, Satya O-G, Jonathan K Kummerfeld, Joelle Pineau, Satinder Singh, and Aaron C Courville. No-press diplomacy: Modeling multi-agent gameplay. volume 32, 2019.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pp. 1–22, 2023.

Yotam Perlitz, Ariel Gera, Ofir Arviv, Asaf Yehudai, Elron Bandel, Eyal Shnarch, Michal Shmueli-Scheuer, and Leshem Choshen. Do these llm benchmarks agree? fixing benchmark evaluation with benchbench. *arXiv preprint arXiv:2407.13696*, 2024.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. Nlp evaluation in trouble: On the need to measure llm data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10776–10787, 2023.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, et al. Openai gpt-5 system card, 2026. URL https://arxiv.org/abs/2601.03267.

Jiakai Tang, Heyang Gao, Xuchen Pan, Lei Wang, Haoran Tan, Dawei Gao, Yushuo Chen, Xu Chen, Yankai Lin, Yaliang Li, et al. Gensim: A general social simulation platform with large language model based agents. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pp. 143–150, 2025.

5 Team et al. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models, 2025. URL https://arxiv.org/abs/2508.06471.

Yuyang Tian, Shunqiang Mao, Wenchang Gao, Lanlan Qiu, and Tianxing He. A visualized framework for event cooperation with generative agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 41700–41702, 2026.

Maximilian Puelma Touzel, Sneheel Sarangi, Austin Welch, Gayatri Krishnakumar, Dan Zhao, Zachary Yang, Hao Yu, Ethan Kosak-Hine, Tom Gibbs, Andreea Musulan, et al. A simulation system towards solving societal-scale manipulation. *arXiv preprint arXiv:2410.13915*, 2024.

Alexander Sasha Vezhnevets, John P Agapiou, Avia Aharon, Ron Ziv, Jayd Matyas, Edgar A Duéñez-Guzmán, William A Cunningham, Simon Osindero, Danny Karmon, and Joel Z Leibo. Generative agent-based modeling with actions grounded in physical, social, or digital space using concordia. *arXiv preprint arXiv:2312.03664*, 2023.

Yue Wu, Xuan Tang, Tom Mitchell, and Yuanzhi Li. Smartplay: A benchmark for llms as intelligent agents. In *International Conference on Learning Representations*, volume 2024, pp. 1543–1561, 2024.

Bushi Xiao, Ziyuan Yin, and Zixuan Shan. Simulating public administration crisis: A novel generative agent-based simulation system to lower technology barriers in social science research. *arXiv preprint arXiv:2311.06957*, 2023.

Kaixuan Xu, Jiajun Chai, Sicheng Li, Yuqian Fu, Yuanheng Zhu, and Dongbin Zhao. Dipllm: Fine-tuning llm for strategic decision-making in diplomacy. *arXiv preprint arXiv:2506.09655*, 2025.

Zelai Xu, Chao Yu, Fei Fang, Yu Wang, and Yi Wu. Language agents with reinforcement learning for strategic play in the werewolf game. *arXiv preprint arXiv:2310.18940*, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, et al. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, et al. Sotopia: Interactive evaluation for social intelligence in language agents. In *International Conference on Learning Representations*, volume 2024, pp. 40975–41019, 2024.

Kunlun Zhu, Hongyi Du, Zhaochen Hong, Xiaocheng Yang, Shuyi Guo, Daisy Zhe Wang, Zhenhailong Wang, Cheng Qian, Robert Tang, Heng Ji, et al. Multiagentbench: Evaluating the collaboration and competition of llm agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8580–8622, 2025.

A STATEMENTS

A.1 POTENTIAL RISKS

While our research focuses on foundational multi-agent evaluation within a simulated environment, we acknowledge the potential dual-use risks associated with developing and evaluating autonomous negotiation agents. If agents capable of open-ended, multi-player bargaining are applied to real-world scenarios—such as automated financial trading, algorithmic pricing, or automated procurement—without rigorous safeguards, they could inadvertently or intentionally engage in deceptive bargaining, collusive price-fixing, or the exploitation of human counterparts who lack equivalent AI assistance.

A.2 ARTIFACTS

In this work, the primary scientific artifacts we utilize are large language models (LLMs) accessed via commercial APIs to drive the autonomous agents. We evaluate a diverse suite of models:

Azure OpenAI Models: We employ `gpt-4o-mini` (version `2024-07-18`) (OpenAI et al., 2024), `gpt-5` (version `2025-08-07`) (Singh et al., 2026), and `o3-mini` (version `2025-01-31`). These models are accessed via the Azure OpenAI API.

Volcengine (Ark) Models: We utilize `deepseek-v3-2-251201` DeepSeek-AI et al. (2025), `doubao-seed-2-0-lite-260428`, `doubao-seed-2-0-pro-260215`, and `glm-4-7-251222` (Team et al., 2025). These are accessed via the Volcengine Ark API [1].

Alibaba Cloud Models: We employ `qwen-plus` (Yang et al., 2025), accessed via the Alibaba Cloud DashScope API [2].

The archive also includes `gemini-3-flash-preview` and `claude-opus-4`; complete historical deployment snapshots are not retained for these identifiers.

Sampling parameters follow the model-specific client configuration. The released implementation explicitly sets `temperature=0` for the Ark and Qwen clients, while the Azure clients leave temperature unspecified. The caller-level default does not override these preconstructed clients, so a uniform temperature setting or deterministic replay is not established for all evaluated calls. We utilize these APIs strictly for research purposes to simulate multi-agent economic behaviors, and our usage fully complies with their respective access conditions and intended usage. Specifically, we adhere to the *OpenAI Terms of Use*[3] and the *Alibaba Cloud Platform Service Agreement*[4]. We do not use or distribute the underlying weights of these models.

A.3 BUDGET

We use approximately 0.6 million tokens per agent as a rough budgeting estimate, not a verified per-turn or per-game average; the released records do not establish a consistent denominator. Token consumption depends on the model, game length, and number of negotiation calls.

A.4 PACKAGES

We utilize LangChain[5] 1.3.1 for establishing agents framework, with MIT license.

[1] `https://ark.cn-beijing.volces.com/api/v3`

[2] `https://dashscope.aliyuncs.com/compatible-mode/v1`

[3] `https://openai.com/policies/row-terms-of-use/`

[4] `https://terms.alicdn.com/legal-agreement/terms/common_platform_service/20230728213935489/20230728213935489.html`

[5] `https://github.com/langchain-ai/langchain.git`

B Environment Implementation Details

B.1 Observation Construction Details

This appendix provides the field-level specification of the structured observation used in Stage 1 of Figure 1. For each agent, the environment converts the global state into four observation channels according to information visibility and the current game phase.

Focal-agent endowment and production capabilities. The focal-agent channel $\mathcal{O}_{private}$ contains information needed for local valuation, production planning, and temporally consistent decision making. The channel name does not imply that inventory is hidden: resource inventories are public, whereas the turn-level plan is private agent memory. It includes the agent's current resource inventory $\mathbf{r}_{t,i}$, the converter set $\mathcal{F}_{t,i}$, species-specific information, and the turn-level strategic plan generated at the beginning of the current turn. Each converter entry specifies its input-output transformation logic, such as converting 3 *Energy* into 1 *Hypertech*, together with relevant meta-properties such as upgrade conditions, compatibility constraints, future investment requirements, and agent-specific production traits.

Public board and social signals. The public channel $\mathcal{O}_{public}$ exposes shared information that helps agents infer scarcity, trading intent, and bargaining opportunities. It includes observable resource counts and asset information of other agents, as well as bulletin-board messages. The bulletin board supports semi-structured cheap talk, including free-form declarations and explicit seeking/offering resource vectors, allowing agents to identify potential trading partners before binding negotiation.

Game phase context. The market channel $\mathcal{O}_{market}$ provides phase-specific information required for bidding and item selection. During the Bid Phase, agents observe the available auction tracks, including candidate items, floor prices in the bidding currency, and functional descriptions. This channel is only populated when the corresponding market information is relevant to the current phase.

Interaction information and state. The interaction channel $\mathcal{O}_{interaction}$ records the agent's active negotiation state. It includes incoming binding proposals $\tau_{j\to i}$ from other agents and the status of outgoing proposals initiated by the agent. This information allows the agent to evaluate offers, track pending commitments, avoid duplicate decisions, and maintain consistency across asynchronous negotiation events.

B.2 Agent Modules

The reference LLM agent relies on a suite of phase-specific modules. The `Brain` dispatcher routes tasks to the following modules; their planning and valuation roles are prompting objectives rather than guarantees of optimal decisions:

`TurnPlanCaller`
Function: Strategic Anchoring.
Trigger: At the beginning of every turn t.
Role: Generates the high-level "Long-term Plan" based on the current game state and species characteristics. This plan serves as a cognitive context for all subsequent actions in the turn, ensuring consistency between the economic goals and tactical execution.

`TradeCaller`
Function: Negotiation and Social Coordination.
Trigger: During the Trading Phase (Asynchronous).
Role: Handles natural language communication. It generates outgoing proposals to resolve resource deficiencies identified by the planner and evaluates incoming proposals from opponents for potential gains. Neither the caller nor the environment guarantees Pareto improvement.

`EconomyCaller`
Function: Resource Optimization.
Trigger: During the Production Phase.

Role: Handles the combinatorial challenge of converter activation. It proposes *Run* (resource conversion) actions to accumulate needed resources, subject to arithmetic constraints checked by the engine. Outputs are buffered until production ends and cannot feed another converter in the same phase; no optimality guarantee is provided.

`BidCaller`

Function: Auction Valuation.
Trigger: During the Auction/Bid Phase.
Role: Analyzes the public board state to estimate opponents' budgets and valuations. It outputs a bid vector $\mathbf{b}_i$ for Colony and Tech tracks, balancing the probability of winning against the risk of overpaying.

`PickCaller`

Function: Asset Selection.
Trigger: During the Pick Phase (if a bid was successful).
Role: Executes the "Right-of-Choice." Given the remaining items on the track and the agent's rank, it selects a Colony or Research Team based on its estimated marginal value to the current inventory.

`DiscardColonyCaller`

Function: Constraint Management.
Trigger: When the agent exceeds the colony ownership limit before the Production phase.
Role: Performs a cost-benefit analysis to determine which asset to liquidate. It identifies the least efficient colony (e.g., one with poor input-output ratios or irrelevant production types) to discard, minimizing the loss of future production capacity.

B.3 THREE PHASES ENVIRONMENT

In this section, we detail the implementation logic for the three game phases within the SidConArena environment.

Phase I: The Negotiation Ledger ($\mathcal{A}_{neg}$).
Corresponding to the cooperative sub-game, the environment provides an unstructured communication channel. It acts as a trusted third party to ensure atomic trade execution. When agents issue binding trade vectors $\tau_{i \leftrightarrow j} \in \mathcal{A}_{neg}$, the environment validates resource ownership and applicable asset rules and executes the transaction only upon acceptance. It does not evaluate or enforce the heuristic mutual-benefit condition in eq. (1).

Phase II: The Production Engine ($\mathcal{A}_{prod}$).
This module implements deterministic production transitions. It executes converter activations represented by $\mathbf{x}_i \in \mathcal{A}_{prod}$ for each agent, with species-specific exceptions to the ordinary single-use rule. The engine validates constraints (ensuring input resources exist in $\mathbf{r}_{t,i}$) and applies the transition function in eq. (2), consuming inputs and buffering outputs until the end of production, when the agent's inventory is updated for the subsequent auction. The final round ends after production without another auction.

Phase III: The Auction Mechanism ($\mathcal{A}_{bid}$).
To instantiate the imperfect-information competitive phase, the environment enforces a "sealed-bid" mechanism. It accepts bid tuples $(b_{i,\mathcal{C}}, b_{i,\mathcal{R}}) \in \mathcal{A}_{bid}$ during the same bidding phase, while omitting opponents' submitted bids from the LLM observation. The system then executes the *Right-of-Choice* priority algorithm, deducting the full *Ships* bid only when an item is selected and distributing distinct assets $(\mathcal{C}, \mathcal{R})$ to determine the starting state for the next turn $\mathbf{r}_{t+1}$. Passing or having no eligible item costs nothing.

B.4 HUMAN INTERFACE

In addition to LLM self-play, *SidConArena* provides a web-based interface for human participation. Human players connect to the same game server as LLM agents and participate under the shared environment state, action semantics, and game rules. The web interface and LLM prompt use different representations, so this does not establish identical observation access. This interface supports mixed-play sessions in which human players can negotiate, cooperate, or compete with LLM agents in real time. The frontend is implemented with Vue.js, as shown in Figure 6.



Figure 6: Web-based human interface for participating in *SidConArena*.

C Additional Analysis of Results

C.1 "Ship" Valuation Bias

Table 1 compares quoted Ship prices against the valuation guide supplied to the agents. The guide assigns approximately one material-value unit to a Ship and to each basic resource (Food, Industry, and Culture), and 1.5 units to Biotech. This reference unit is distinct from terminal points: a remaining Ship contributes $1/6$ point. The examples price a Ship at 3 Industry, 2 Culture, 4 Culture plus 1 Biotech, and 2 Food, corresponding to **3, 2, 5.5, and 2 times** its reference value.

The first two cases show buyers offering a premium that the Ship sellers submit actions to accept. The sellers describe the offers favorably because they receive more reference value than they give. Their arithmetic is correct, but the first seller, DeepSeek V3.2, also explicitly acknowledges that 3.0 exceeds the fairness limit and still submits an acceptance. The prompt applies the check to both proposals and acceptances. In the third case, GPT-4o Mini's prompt requires a receive-to-give value ratio within 0.66–1.5 for small trades and 0.85–1.15 for larger trades. Its proposed ratio of 5.5 falls outside both ranges, despite its stated commitment to the fairness check. In the fourth case, Claude Opus 4 calls its two-for-one buying quote a "fair value trade" while seeking Ships for auction bidding. Thus, the table exposes both elevated Ship prices and concrete failures to apply a stated valuation check.

These cases diagnose alignment between the supplied value scale, trade terms, and the agents' explanations. Because Ships provide auction access, a premium may have a strategic motivation; establishing a loss in terminal score would require subsequent production and auction context or a counterfactual comparison. The selected examples do not estimate how frequently these mismatches occur or establish that pretrained semantic associations cause them.

D Experimental Details and Analysis Procedure

This section provides the experimental details omitted from the main text for space. All experiments are conducted in the *SidConArena* environment described in Section 4. Each game follows the three-phase structure formalized in Section 3: agents first negotiate and submit binding trades, then activate production converters, and, except in the final round, participate in a simultaneous sealed-bid auction for Colonies and Research Teams. The environment acts as the authoritative state manager:

it validates trade feasibility, enforces production constraints, resolves auctions, and records the full trajectory of messages, structured actions, resource states, bids, allocations, and terminal scores.

Agents receive observations through the structured interface illustrated in Figure 1, which assembles focal-agent inventory, production capabilities, public board information, incoming and outgoing proposals, and phase-specific market information into a unified textual prompt. The same figure also summarizes the phase-aware dispatching and validated action interface used during evaluation: model outputs are converted into structured function calls for trading, production, bidding, and item selection, and are then validated and executed by the environment engine. This design lets agents reason in natural language while ensuring that committed game actions are machine-checkable and executed under identical environment rules.

Evaluated agents. To obtain a broad and model-agnostic assessment of *SidConArena*, we evaluate LLM agents spanning multiple model families, capacity regimes, and deployment paradigms. All models are evaluated with the same environment interface, observation format, action schema, and game rules, without model-specific tools or privileged state access.

Homogeneous model rollouts. Our first evaluation setting describes terminal economic performance in homogeneous multi-agent rollouts. For each model backbone, all players in a game are controlled by agents instantiated from that same model, with identical prompting, observation, action interfaces, and environment rules. As shown in Figure 2, each game repeatedly executes the shared agent–environment loop introduced in Figure 1. This setting evaluates how well a model can sustain an economy when all players share the same reasoning capabilities, valuation tendencies, and bargaining style.

Heterogeneous Elo tournament. Our second evaluation setting measures relative competitive strength under the same rollout protocol, but with different model backbones assigned to different players. In this setting, agents directly interact in a shared economy: they negotiate with heterogeneous opponents, compete for scarce auction assets, and adapt to different valuation and bargaining behaviors. After collecting a batch of heterogeneous tournament games, we compute model-level Elo ratings offline from the terminal outcomes of all participating agents. Compared with homogeneous rollouts, this tournament setting tests both whether an agent can translate its economic reasoning into strong outcomes and how competitive it is when directly matched against agents instantiated from other model backbones.

Terminal economic performance. Terminal score is the primary outcome measure for both evaluation settings. It is computed from the terminal value of each player's holdings after the final turn: points (including point donations) plus one sixth of the tabulated value of remaining materials. Because *SidConArena* rewards long-horizon engine building rather than isolated local gains, terminal score provides a compact measure of an agent's ability to trade, produce, invest, bid, and convert resources into end-game value.

Elo rating. To summarize relative competitive strength in heterogeneous tournaments, we compute an Elo-style rating from the collected tournament results. We estimate model-level ratings offline from a batch of completed games, using the terminal outcomes of all agents as the observed competition results.

Action validity and local valuation. To distinguish interface-level execution from economic reasoning, the implementation records function-call attempts, successes, and failures. These counters include exceptions that trigger fallback and are not a pure schema-validity measure. We inspect selected trades and proposals against the game-specific resource reference table, with a focus on Ship-related reasoning; a difference from the table alone does not establish a utility loss.

Positive-sum cooperation. The negotiation logs can support future diagnostics of accepted and rejected proposals, counteroffers, partner diversity, and production utilization. We do not report aggregate rates for these behaviors here. Identifying mutually beneficial or Pareto-improving trades requires state-dependent valuation, and an unused converter alone does not establish a missed profitable production opportunity.

Qualitative trajectory analysis. Beyond aggregate scores, we qualitatively inspect selected trajectory logs to characterize strategic behaviors that are difficult to reduce to a single scalar metric. For bargaining, we examine whether agents passively accept fair-looking offers or actively counteroffer, preserve optionality, and leverage temporary scarcity. For long-horizon planning, we inspect whether turn-level decisions form a coherent trajectory from early investment to compounding production and late-game score conversion.

Analysis procedure. For quantitative analysis, we aggregate terminal scores by model backbone in homogeneous self-play and compute Elo ratings from heterogeneous tournament games. For diagnostic analysis, we parse the recorded trajectories produced by the pipeline in Figure 1. The same rollout records therefore support both outcome-level evaluation and behavior-level diagnosis. This combined protocol is important because many failures in *SidConArena* are strategic rather than syntactic: an agent may submit valid actions while still mispricing resources, refusing beneficial trades, bargaining passively, overpaying in auctions, or failing to exploit compounding production returns.