Multi-Agent IPL Auction Simulation

***Abstract*—This paper presents a multi-agent IPL 2026 mini auction simulation, where ten LLM-powered franchise agents compete for remaining squad slots under individual purse constraints (2.75 Cr to 64.3 Cr). Agents strategically bid on 370+ uncapped Indian and overseas players across four specializations to complete 25-player squads, using DeepSeek-based reasoning modules to analyze player statistics, team composition gaps, rival budgets, and market dynamics before making raise/pass decisions.**

***Index Terms*—IPL auction, multi-agent systems, LLM agents, auction simulation**

I. Introduction

The Indian Premier League (IPL) [1] auction is one of the most complex and high-stakes decision-making environments in modern sports, where ten franchises compete in a constrained marketplace to assemble competitive squads under strict budget and role requirements. In the 2026 mini auction this became even more interesting, with uneven remaining budgets, a heavy focus on uncapped Indian talent, and a few intense bidding wars.

This project recreates the 2026 mini auction as a full simulation but replaces human decision-makers with a team of LLM-based agents. Each agent knows the status of its team and operates within a LangGraph pipeline that closely follows the real auction flow: introducing sets, collecting bids, deciding the winner, and updating squads after every player. Behind the scenes, the agents rely on DeepSeek based reasoning LLM to parse player stats, assess their current team balance, monitor what other teams can still afford, and then produce structured bidding actions.

Milgrom and Weber's [2] groundbreaking research on English auctions lays the groundwork for understanding how people bid competitively [3] [4]. In our context of IPL auction, our LLM agents attempt to replicate this strategic reasoning within the constraints of budget and team composition requirements, with bidding wars, attempts to drain other teams' pockets, and decisions shaped not only by data but also by gut feel.

II. Related Work

The problem of optimal bidding under budget constraints has been studied in the context of reinforcement learning. Zhang et al. [5] show that model-free deep RL can learn effective bidding policies under strict budget limits, while theoretical work on simultaneous all-pay auctions [6] characterizes how binding budgets distort equilibrium bids. Multi-agent auction simulation has been explored using reinforcement learning agents [7] [8], where multiple agents learn bidding strategies through repeated auction interactions. Our approach used LLM-based agents (DeepSeek) for optimal bidding strategies within budget constraints.

III. Methodology

The auction mechanism we simulate closely follows the ascending auction format analyzed by Cramton, where prices incrementally increase as teams bid against each other. [9] This format has been shown to encourage price discovery, where information about team valuations is gradually revealed through successive bids. In our simulation, the Trade Master implements this ascending format by collecting bids in successive rounds at increasing price points. The system is built on a LangGraph-driven state machine where a single shared AgentState tracks all auction information. This system is of 5 Nodes.

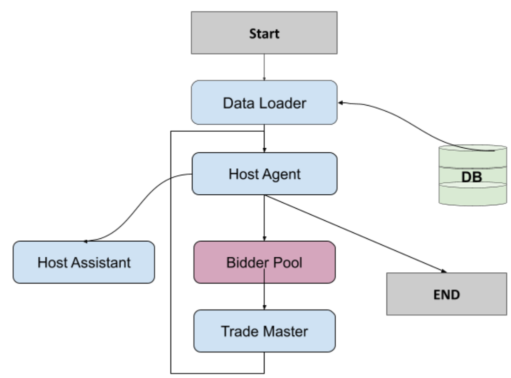

Fig. 1. System architecture: Five interconnected nodes in the LangGraph state machine.

A. Data Loader

The Data Loader node forms the initial setup of the project by building team objects as if they were team after player retentions. It imports data scraped from the official IPL website and stores it in a CSV file, ensuring each team's budget and player list accurately reflect the real post-retention scenario, which is derived from another separate CSV file. [1] [10] [11] It also imports data of players in auction and their stats, Set etc.

B. Host

The Host node controls the simulation by deciding what happens next at each step. It can either send control to the Host Assistant node to select the next player, or to the Bidder Pool to run the bidding process, or terminate the process once the auction is fully completed.

C. Host Assistant

The Host Assistant node selects the next player from the current set of players for auction and route back to Host. When all players in the current set have been processed, it advances the simulation by moving on to the next set.

D. Bidder Pool

The Bidder Pool node consists of ten AI-controlled franchise agents that compete for the current player in a Greedy Sequential Order, using past bidding history to arrange teams in a sequence that both reduces the number of LLM/API calls and recreates realistic bidding wars.

For each player, teams are queried one by one in this order; each agent either bids or passes, and the process continues until either all teams have passed or at least one valid bid is placed, at which point the result for that round is returned and control is handed over to the Trade Master for validation and possible completion of the sale.

Each bidding turn uses a chain-of-thought (CoT) [12] [13] reasoning process, which lets agents think through important decision points clearly before deciding whether to bid or pass. This structured reasoning makes sure that bidding behavior is not random or just a reaction, but rather understandable, consistent, and based on strategy.

1) **Player valuation bands:** Establish lower (safe) and upper (fair ceiling) value ranges for each player based on recent form and consistency.
2) **Evaluate player utility:** Assess player fit and performance metrics relative to team needs.
3) **Check squad constraints:** Verify role balance, overseas quota, and current roster composition.
4) **Analyze rivals:** Consider remaining purses and existing bidding patterns of other teams.
5) **Update purse trajectory:** Ensure budget alignment with future auction phases.

E. Trade Master

The Trade Master node processes the bid information received from the Bidder Pool and updates the current price accordingly. If no new bids are placed in a round, it advances to next round. If this has happened for three consecutive, it finalizes the player's trade, invokes an AI reasoner to generate an explanation for why that team bought that player at that price, stores this rationale in the state for that team, and then routes control back to the Bidder Pool so that this accumulated reasoning can inform the team's subsequent trades.

The AI reasoner employs a constraint-based prompt architecture [12] [13] [14] that structures explanations across ten dimensions namely: player role, statistics, squad fit, optimal slot, price valuation, risk management, upside, fan appeal, auction timing, and closing justification in a single paragraph. This reason generated here would be a key to retain information about this player in later bids of other players. This reduces hallucinations about players in team.

IV. Experiments

The simulator is evaluated as a ground-truth replication of the IPL 2026 mini auction, treating the official 2026 mini-auction player list, reserve prices, and final team squads as the reference outcome the system is trying to reproduce. In other words, instead of constructing a synthetic benchmark, the real auction itself defines the target distribution of players across teams and the price levels at which those players change hands.

All bidding decisions are generated by ten franchise agents calling the deepseek-ai/deepseek-v3.1 model through the NVIDIA API, within the LangGraph workflow. Full-auction simulations are run on an AWS EC2 instance to avoid mid-run interruptions, with a typical end-to-end run taking around three hours for the complete 2026 mini auction.

Two execution modes are used: a faster, terminal-based mode for reliable full simulations, and a Streamlit dashboard mode for real-time monitoring and visual comparison of simulated squads and spending patterns against the ground-truth auction.

V. Results

Our method of evaluation, which compares simulated outcomes to real auction outcomes, is similar to how auction agents are evaluated these days, where agents' strategies are judged by how well they can replicate real-world auction results. [8] The mock auction reproduces a substantial portion of the IPL 2026 mini auction while still showing systematic shifts in valuation behaviour, with 68 players bought in the simulation compared to 77 in the real event.

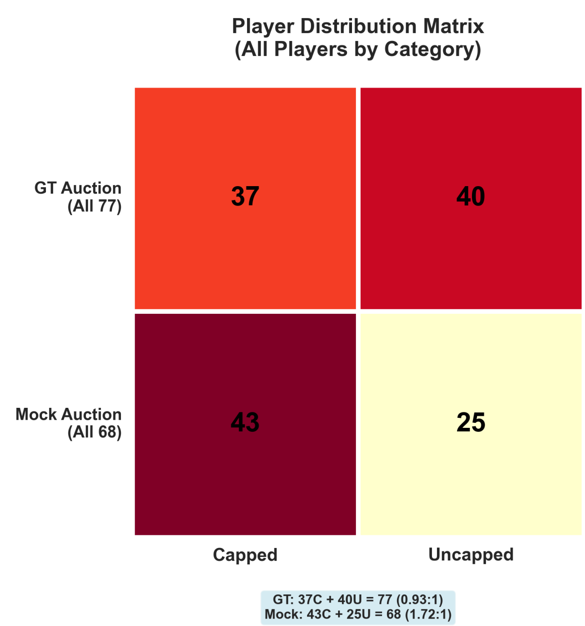

Fig. 2. Player distribution heatmap comparing capped and uncapped representation in the ground-truth and mock auctions

A. Overlap

The simulation matches 28 of 77 ground-truth players (36.4% overlap), with most of this overlap coming from established, capped players: 78.6% of the common picks come from the capped category. In pricing, capped players in the mock retain 80.5% of their ground-truth average value, whereas uncapped players retain only 12.7%.

B. Capped vs Uncapped

TABLE I
AVERAGE AND MAXIMUM PRICES FOR CAPPED VS UNCAPPED PLAYERS

Category	Average Price (Cr)		Maximum Price (Cr)	
	Ground Truth	Mock	Ground Truth	Mock
Capped	5.22	4.20	25.20	12.00
Uncapped	3.30	0.42	14.20	0.75

The mock auction is far more conservative and flattened in its pricing, particularly for uncapped players. For capped players, it reduces the average from 5.22 Cr to 4.20 Cr and the maximum from 25.20 Cr to 12 Cr, clearly undervaluing them but still preserving a noticeable spread and a recognizable top tier of expensive picks. For uncapped players, both the mean and the ceiling effectively collapse; the average falls from 3.30 Cr to 0.42 Cr, and the maximum falls from 14.20 Cr to 0.75 Cr, showing that the mock almost never places big, differentiated bets on youngsters and instead compresses them into a narrow, low-value band. This is unlike the real auction, where a few uncapped players attract very aggressive bids.

TABLE II
PLAYER COUNTS IN MOCK AUCTION VS COMMON WITH GROUND TRUTH

Category	Total in Mock	Common with GT	Common %
Capped	43	22	51.2%
Uncapped	25	6	24.0%

The difference in ratios shows that the mock auction is far more in sync with the real auction for capped players than for uncapped ones. For capped players, a little over half of the mock picks overlap with the actual auction, while for uncapped players not even a quarter match.

VI. DISCUSSION

A. Data Inconsistency

Limited statistics are available for many uncapped players, which makes it difficult to evaluate them with the same confidence as capped players. This data gap naturally leads to inconsistency between how capped and uncapped players are treated in the mock auction versus the ground truth auction.

B. Greedy Sequential Order

While greedy sequential bidding captures the real-world auction wars [2] [3] [4] , early aggressive jumps often anchor inflated prices that discourage broader team participation. This dynamic prevents many franchises from getting a chance to bid at all, suppressing the competitive pressure needed to keep valuations realistic. Ultimately, this lack of engagement allows a few dominant bidders to pull the final price away from a player's underlying true market value.

Akeal Hosein (Left-arm Spinner).

Only SRH and CSK engaged in a back-and-forth contest across 5 bidding rounds. In all 5 rounds, either SRH or CSK appeared early in the processing order, placed the first bid, and immediately stopped further evaluation. Teams such as KKR, DC, LSG, and GT were only checked at the final price of Rs.2.80 Cr and never had a chance to participate at the intermediate price points (Rs.2.00, 2.20, 2.40, 2.60).

David Miller (Batter).

Here, only RR and KKR traded bids over 3 rounds. In every round, one of these two teams was processed first and bid early, which short-circuited evaluation for 6–8 other interested teams that were never allowed to respond at lower intermediate prices.

C. Bid History Causing Model Context Confusion

As IBM's analysis notes, managing context effectively is crucial for decision-making systems. [15] The system currently feeds a very long bid-history of a particular player into the model before each new bidding decision. This caused it to

- lose track of the actual current auction state,
- hallucinate justifications by overfitting to earlier, similar bidding patterns, and
- produce almost identical bid responses across multiple teams.

Devon Conway at INR 8.25 Cr

Teams showing suspiciously similar responses:

KKR's Response: Price exceeds our value band by 20%+ ; better to target younger openers like Fraser-McGurk or Shaw later who offer similar upside at potentially lower cost.

DC's Response: Price exceeds our value band by 20%+; already have KL Rahul as anchor opener and multiple overseas batters. Better to target younger openers like Fraser-McGurk or Shaw later at potentially lower cost.

PBKS's Response: Price exceeds value band for overseas opener; Prabhsimran provides domestic opening option. Better to target younger openers like Fraser-McGurk or Shaw later.

All three teams mention "Fraser-McGurk or Shaw" as alternatives. This suggests the model is picking up on patterns from the bid history rather than evaluating each team's unique situation. The phrase "Fraser-McGurk or Shaw later" appears almost across multiple teams.

Jake Fraser-McGurk at INR 6.00 Cr

Nearly Identical Responses from 8 Different Teams:

Bid 1 (INR 2.50): Proven CSK opener at steal price; custom raise secures ideal partner for Gaikwad while maintaining budget for critical bowling needs.

Bid 2 (INR 3.50): Proven CSK opener at fair price; custom raise secures ideal partner for Gaikwad while maintaining budget for critical bowling needs.

Bid 3 (INR 4.50): Proven CSK opener at fair price; custom raise secures ideal partner for Gaikwad while maintaining budget for critical bowling needs.

The reasoning barely changes ("steal price" → "fair price") despite the bid increasing by INR 2 Cr. This suggests the model is stuck in a loop, reusing the same justification from bid history rather than re-evaluating the value proposition at each price point.

VII. Conclusion

The multi-agent mock auction recreates many structural aspects of the IPL 2026 mini auction, achieving realistic overlap in player selections and spending patterns. At the same time it exposes main limitations of LLMs in long context reasoning and hallucinations.

References

[1] "IPL 2025 auction — Players bought, top bids & team spending — IPLT20," *Iplt20.com*, 2025. [Online]. Available: https://www.iplt20.com/auction/2025 .

[2] P. R. Milgrom and R. J. Weber, "A theory of auctions and competitive bidding," *Econometrica*, vol. 50, no. 5, pp. 1089–1122, 1982.

[3] D. Easley and J. Kleinberg, "Auctions," in *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*, Cambridge University Press, 2010, ch. 9, pp. 249–266.

[4] "English auction," Mechanism.institute, 2017. Accessed: Jan. 14, 2026. [Online]. Available: https://www.mechanism.institute/library/english-auction .

[5] D. Wu *et al.*, "Budget constrained bidding by model-free reinforcement learning in display advertising," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manag. (CIKM)*, Oct. 2018, doi: https://doi.org/10.1145/3269206.3271748.

[6] Y. Liu, Y. Qin, and Z. Wang, "Simultaneous all-pay auctions with budget constraints," arXiv:2505.03291, 2025.

[7] M. Kanmaz and E. Surer, "Using multi-agent reinforcement learning in auction simulations," arXiv:2004.02764, 2020.

[8] G. d'Eon, N. Newman, and K. Leyton-Brown, "Understanding iterative combinatorial auction designs via multi-agent reinforcement learning," in *Proc. 25th ACM Conf. Econ. Comput. (EC)*, pp. 1102–1130, Jul. 2024, doi: https://doi.org/10.1145/3670865.3673644.

[9] P. Cramton, "Ascending auctions," *European Economic Review*, vol. 42, no. 3–5, pp. 745–756, May 1998.

[10] Ptl_aman_, "IPL mini auction 2026 player," *Kaggle.com*, 2026. [Online]. Available: https://www.kaggle.com/datasets/amanpatel111/ipl-auction-player-dataset . [Accessed: Jan. 15, 2026].

[11] "TATA IPL 2026 playing squad – 15.11.2025," [Online]. Available: https://documents.iplt20.com/bcci/documents/1763209725967_TATA%20IPL%202026%20-%20Playing%20Squad%20-%2015.11.2025%20(1).pdf . [Accessed: Jan. 15, 2026].

[12] J. Wei *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," arXiv:2201.11903, 2022.

[13] I. Zarecki, "Chain-of-thought reasoning supercharges enterprise LLMs," K2view.com, Aug. 26, 2024. [Online]. Available: https://www.k2view.com/blog/chain-of-thought-reasoning/. [Accessed: Jan. 14, 2026].

[14] Z. Lin, C. Chan, Y. Song, and X. Liu, "Constrained reasoning chains for enhancing theory-of-mind in large language models," arXiv:2409.13490, 2024.

[15] IBM, "Context window," *Ibm.com*, Nov. 22, 2024. [Online]. Available: https://www.ibm.com/think/topics/context-window .