,,biblatex-chicago refs.bib

Problem Set 1

Peilin Wu

October 9, 2025

1 Introduction

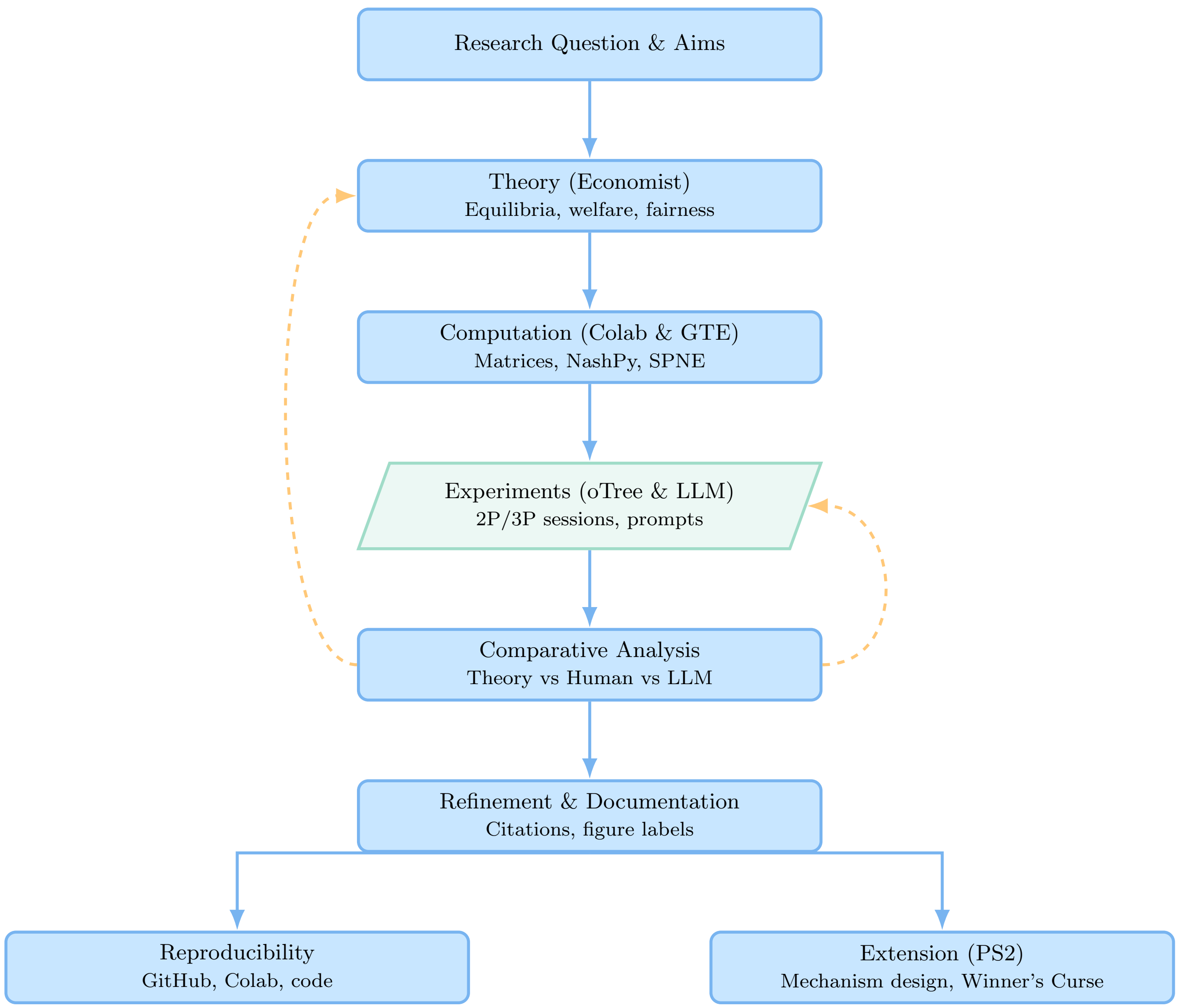

Figure 1: Vertical project flow: from research question to PS2 extension.

Bargaining games are a central topic in game theory. It models how rational agents divide a fixed surplus through strategic interaction. In their simplest form, two players simultaneously demand portions of a common resource (for example, 100 points). If the total of their demands does not exceed the available resource, each player receives exactly what they demanded. If the total exceeds the resource, both players receive nothing. [OR94, p. 117] This setup highlights the tension between self-interest and coordination, as multiple equilibria exist, ranging from equal and fair divisions to extreme and highly unequal outcomes.

In this project, I extend the classic two-player simultaneous demand bargaining game by increasing the number of players to three. This modification allows me to investigate how equilibrium concepts, efficiency, and fairness change when coordination becomes more difficult and when the possibility of inefficient "zero-payoff" equilibria emerges. By adapting the player count, I aim to test both theoretical predictions and behavioral responses in more complex group settings.

Following the assignment requirements, my work will integrate three perspectives shown as follows:

Economist's perspective (theory and welfare): I will formally define the strategic game for different player counts, characterize Nash equilibria, and evaluate their efficiency and fairness properties.

Computational scientist's perspective (coding and tools): I will construct payoff representations, compute equilibria using software such as NashPy (for two-player cases) and custom scripts (for three or more players), and visualize outcomes. I will also create extensive-form versions in Game Theory Explorer.

Behavioral scientist's perspective (experiment and AI comparison): I will deploy an oTree implementation of the bargaining game with different player numbers, conduct short human sessions, and run parallel sessions with a large language model. I will then compare equilibrium predictions with observed human and AI behavior to explore discrepancies and potential refinements to existing solution concepts.

To sum up, through this interdisciplinary approach, the project will shed light on how bargaining outcomes evolve when the number of participants grows, combining theoretical analysis, computational methods, and behavioral evidence.

Variant	Players	Payoff rule	Equilibria (summary)
Simultaneous (baseline)	2	$u_i(d) = d_i$ if $\sum_j d_j \le 100$, else 0	All (d_1, d_2) with $d_1+d_2 = 100$; efficient but fairness varies
Simultaneous (extension)	3	Same rule	Efficient if $\sum_i d_i = 100$; also *over-ask* NE with all 0
Alternating offers (O&R)	2	Discounted, perfect information	Unique SPNE; immediate efficient agreement; first–mover advantage
Computational tools	–	–	NashPy (2P), enumeration ($n{\ge}3$), GTE (SPNE), oTree (deploy)

Table 1: Compact overview of models, rules, and equilibria.

2 Economist (theory & welfare)

1. Equilibrium concept

Normal-form simultaneous-demand bargaining. Consider a bargaining game in which two (or n) players simultaneously demand shares of a fixed surplus (e.g., 100 points). Let $d_i \in [0, 100]$ be player i's demand and $S = \sum_j d_j$. Payoffs are

$$u_i(d) = \begin{cases} d_i, & \text{if } S \le 100, \\ 0, & \text{if } S > 100. \end{cases}$$

A *Nash equilibrium (NE)* is a profile $d^\star$ such that no player can profitably deviate: $u_i(d_i^\star, d_{-i}^\star) \ge u_i(d_i, d_{-i}^\star)$ for all d_i in her strategy set [SLB08]. Existence of (mixed) NE in finite normal-form games follows from Nash's fixed-point argument; in continuous games with compact strategy sets and continuous, suitably quasi-concave payoffs, Glicksberg's generalization applies [SLB08].

Extensive-form alternating offers and SPNE. To study timing, commitment, and credibility, I also analyze the alternating-offers model as an extensive game with perfect information, following Osborne and Rubinstein (O&R) [OR94]. A *subgame perfect Nash equilibrium (SPNE)* induces a Nash equilibrium in every subgame and rules out non-credible threats [OR94, §7.3]. Under O&R's assumptions A1–A4, there is an (essentially) unique SPNE with immediate agreement on an efficient division [OR94, Prop. 122.1, pp. 122–126]. In the split-the-pie case with geometric discounting $\delta_i \in (0, 1)$, the proposer gets

$$a^{=\frac{1-\delta_2}{1-\delta_1\delta_2}}, \qquad 1-a^{\text{ to the responder,}}$$

and agreement is immediate [OR94, Example 125.1].

Proof idea (sketch). Stationarity and the one-deviation property pin down the responders' acceptance rules: each accepts any offer weakly better than her continuation value (the next period's efficient agreement, discounted). The two acceptance thresholds intersect at a unique efficient pair $(x^{,y^{)}}$, implying immediate agreement [OR94, §7.3; Prop. 122.1].

2. Analytical solution

(a) Two-player simultaneous-demand game. Best responses imply that every profile with $d_1 + d_2 = 100$ is a (pure) NE: any unilateral increase triggers $S > 100$ and payoff 0, while any unilateral decrease strictly lowers one's own payoff. Hence the NE set is a continuum from $(50, 50)$ to extreme splits like $(99, 1)$. All NE with $S = 100$ are *Pareto efficient*; outcomes with $S < 100$ are not NE. Fairness varies across the NE set (equality vs. inequality), which I will quantify with simple inequality indices in Part 2.

(b) n-player extension (my modification). For $n \geq 3$, the NE set strictly expands:

- **Efficient NE:** any $d \geq 0$ with $\sum_i d_i = 100$.
- **Inefficient "over-ask" NE (all get 0):** any d such that for every i, $\sum_{j \neq i} d_j > 100$. Intuition: even if one player deviates downward, the others already overfill the budget; no unilateral deviation can unlock a positive payoff.

These zero-payoff equilibria do not exist when $n = 2$ but appear for $n \geq 3$, creating coordination risk and welfare losses. I will verify this characterization computationally by enumeration and best-response filters in Part 2.

(c) Alternating offers: efficiency and first-mover advantage. Under A1–A4, SPNE yields immediate agreement on the Pareto frontier; strategies are stationary; and the first mover enjoys an advantage (with $\delta_1 = \delta_2 = \delta$, the proposer receives $1/(1 + \delta) > 1/2$) [OR94, §7.3.2, p. 126]. Variants—outside options, risk of breakdown, indivisibilities—shift SPNE outcomes in predictable ways [OR94, §7.4.3–§7.4.4, pp. 127–130]. For more than two players, uniqueness fails; with $\delta \geq 1/2$, there exist SPNE supporting immediate acceptance of any proposed split [OR94, §7.4.5, pp. 130–131].

Welfare and fairness. *Pareto efficiency:* simultaneous-demand NE are efficient iff $\sum_i d_i = 100$; alternating-offers SPNE (under A1–A4) are efficient and immediate [OR94, §7.3.2]. *Utilitarian welfare:* equals 100 in efficient equilibria, 0 in over-ask equilibria (for $n \geq 3$). *Fairness:* equal split minimizes inequality; alternating offers with symmetric patience still favors the first mover; outside options tilt outcomes toward the player with the better option [OR94, §7.4.3].

3. Interpretation (realism, multiplicity, refinements; bounded rationality & computation)

The simultaneous-demand game features a continuum of NE (coordination problem) and, for $n \geq 3$, zero-payoff NE that plausibly arise in experiments. Alternating offers with SPNE collapses this multiplicity to a single efficient outcome but relies on strong assumptions (perfect information, stationarity, rational foresight) [OR94, §7.3]. From a computational perspective, equilibrium computation is hard in general (PPAD-complete), motivating algorithmic and approximate methods [Rou16, Lec. 2]; standard AI/game-theory texts survey definitions, existence, and computational issues [SLB08].

3 Computational Scientist

a) Google Colab (normal form + computation)

To analyze the simultaneous-demand bargaining game computationally, I implemented the payoff structure in a Google Colab notebook. The payoff rule is:

$$u_i(d_1, d_2) = \begin{cases} d_i, & d_1 + d_2 \leq 100, \\ 0, & d_1 + d_2 > 100. \end{cases}$$

Payoff matrices (figure 1). In **figure 1**, the notebook constructs and displays the normal-form payoff matrices for both players. The matrices correctly encode the rule that each player receives her demand if the total is within the budget, and both receive zero otherwise. A validation routine confirmed that every entry followed this definition exactly.

	P2:0	P2:25	P2:50	P2:75	P2:100
P1:0	0	0	0	0	0
P1:25	25	25	25	25	0
P1:50	50	50	50	0	0
P1:75	75	75	0	0	0
P1:100	100	0	0	0	0

Payoff matrix B (P2):

	P2:0	P2:25	P2:50	P2:75	P2:100
P1:0	0	25	50	75	100
P1:25	0	25	50	75	0
P1:50	0	25	50	0	0
P1:75	0	25	0	0	0
P1:100	0	0	0	0	0

Figure 2: Payoff matrices generated in Google Colab (figure 1).

Solver outputs (figure 2). In **figure 2**, NashPy's support-enumeration solver was applied to the bargaining payoff matrices.[Kni21] The solver returned a set of equilibria in both pure and mixed strategies. All pure-strategy equilibria correspond exactly to demand pairs where the total equals the available surplus of 100 (e.g. (0,100), (25,75), (50,50), (100,0)). These outcomes are Nash equilibria because any unilateral deviation either pushes the sum above 100—yielding a payoff of zero for both players—or reduces the deviator's own payoff.

NashPy also reports several mixed-strategy equilibria (e.g. mixing between 25,75 or 25,50,75). However, the expected payoffs of these mixed equilibria lie on the same efficient frontier as the pure equilibria: the expected allocation still sums to 100. Thus, the computational results are fully consistent with the theoretical characterization that the equilibrium set consists of all profiles where demands add up to the resource total, while mixtures only reproduce convex combinations of those pure equilibria.

```
NashPy found 10 equilibria
Eq 1: supp(P1)=[0], supp(P2)=[100], E[U]=(0.0,100.0)
Eq 2: supp(P1)=[25], supp(P2)=[75], E[U]=(25.0,75.0)
Eq 3: supp(P1)=[50], supp(P2)=[50], E[U]=(50.0,50.0)
Eq 4: supp(P1)=[75], supp(P2)=[25], E[U]=(75.0,25.0)
Eq 5: supp(P1)=[100], supp(P2)=[0], E[U]=(100.0,0.0)
Eq 6: supp(P1)=[100], supp(P2)=[100], E[U]=(0.0,0.0)
Eq 7: supp(P1)=[25, 50], supp(P2)=[50, 75], E[U]=(24.999999999999996,50.0)
Eq 8: supp(P1)=[25, 75], supp(P2)=[25, 75], E[U]=(24.999999999999996,24.999999999999996)
Eq 9: supp(P1)=[50, 75], supp(P2)=[25, 50], E[U]=(50.0,24.999999999999996)
Eq 10: supp(P1)=[25, 50, 75], supp(P2)=[25, 50, 75], E[U]=(24.999999999999993,25.0)
/usr/local/lib/python3.12/dist-packages/nashpy/algorithms/support_enumeration.py:260: RuntimeWarning:
An even number of (10) equilibria was returned. This
indicates that the game is degenerate. Consider using another algorithm
to investigate.
```

Figure 3: Solver output from NashPy (figure 2).

Interpretation. These equilibria are efficient, because the full surplus is allocated, but they vary in fairness from equal splits to highly unequal ones. When the action grid is extended to include demands above 100 (e.g. $\{0, 50, 100, 150\}$), additional "over-ask" equilibria such as (150,150) appear. In these cases, both players receive zero, but no unilateral deviation improves payoffs within that restricted grid. This shows how discretization can create inefficient equilibria absent in the continuous model. Overall, the Colab outputs confirm the theoretical predictions of Part 1: the game admits a continuum of efficient Nash equilibria when total demand equals the surplus, while discretization can generate spurious zero-payoff outcomes.

b) Game Theory Explorer (extensive form + SPNE)

Model construction. Using *Game Theory Explorer* (GTE), I translated the simultaneous-demand bargaining game into an extensive form. [SvS15] Player 1 moves first, choosing a demand (0, 50, 100, 150). Player 2 then responds at the same time. Information sets were assigned so that Player 2 knows the history. For experiments with more than two players (Figure 6), Player 3 is added sequentially, which generates a deeper tree and illustrates the rising coordination difficulty.

Solver application. GTE provides built-in algorithms to compute Nash equilibria and subgame perfect equilibria (SPNE). Figure 4 displays the solver panel, which returns two equilibria for the restricted action set $\{0, 50, 100, 150\}$. One equilibrium corresponds to the efficient split $(50, 50)$, where the pie is exactly exhausted. The other equilibrium is the inefficient outcome $(150, 150)$, where both players overshoot and receive zero. The strategic form representation in Figure 5 confirms these equilibria in the normal form payoff matrix, marking Nash equilibria with red/blue indicators. Finally, Figure 6 illustrates the extensive form tree for the two-player sequential version, making clear how payoffs are assigned at terminal nodes.

Relation to theory (Part 1). The equilibria computed by GTE are consistent with the analytical results from Part 1: (i) efficient Nash equilibria occur whenever demands sum to 100; (ii) discretization introduces additional inefficient "over-ask" equilibria (e.g., $(150, 150)$), which are not present in the continuous two-player model but arise in the restricted action grid; (iii) SPNE refinement rules out non-credible strategies. In particular, under the alternating-offers model with perfect information, Osborne and Rubinstein show that SPNE collapses to immediate agreement on an efficient split. The GTE results visualize this refinement by highlighting credible equilibria.

Extension to three players. Figure 7 extends the tree to three players. Here, Player 1 moves first, then Player 2, and finally Player 3. The number of terminal nodes expands quickly, illustrating how coordination becomes harder as n increases. Inefficient zero-payoff equilibria—where all players demand too much so that $\sum_i d_i > 100$—become possible and even likely. This aligns with the theoretical prediction that multiplicity and welfare losses emerge when $n \geq 3$.

```
4 x 4 payoff matrix A:
   0    0    0  0
  50   50   50  0
 100  100  100  0
   0    0    0  0

4 x 4 payoff matrix B:
 0  50  100  0
 0  50  100  0
 0  50  100  0
 0   0    0  0

EE = Extreme Equilibrium, EP = Expected Payoff
Decimal Output
  EE  1  P1:  (1)  0.000000  0.000000  0.000000  1.000000  EP=    0.0  P2:  (1)  0.000000  0.000000  0.000000  1.000000  EP=    0.0
  EE  2  P1:  (2)  0.000000  0.000000  1.000000  0.000000  EP=  100.0  P2:  (2)  0.000000  0.000000  1.000000  0.000000  EP=  100.0

Rational Output
  EE  1  P1:  (1)  0  0  0  1  EP=    0  P2:  (1)  0  0  0  1  EP=    0
  EE  2  P1:  (2)  0  0  1  0  EP=  100  P2:  (2)  0  0  1  0  EP=  100

Connected component 1:
{1}  x  {1}

Connected component 2:
{2}  x  {2}

D. Avis, G. Rosenberg, R. Savani, and B. von Stengel (2010),
Enumeration of Nash equilibria for two-player games.
Economic Theory 42, 9-37
```

Figure 4: Solver panel in GTE showing equilibria for the discrete action set $\{0, 50, 100, 150\}$.

4 Behavioral Scientist (experiment & AI comparison)

4.1 oTree deployment and adaptations

Base and rationale. I adapted the official oTree public-goods/bargaining demo as a base.[CSW16][1] To study coordination with increasing group size, I implemented both 2-player and 3-player *simultaneous-demand* versions under the common budget rule.

What I changed (beyond the demo). (i) **Session configuration**: added two session types `Bargaining2P` and `Bargaining3P`, each with `num_rounds=10` and fixed matching within-session. (ii) **Players/roles**: generalized `Constants.players_per_group` from 2 to 3 in the 3P treatment. (iii) **Payoff logic**: replaced the demo payoff with the budget rule $u_i(d) = d_i$ if $\sum_j d_j \leq 100$, else 0. Implemented in `models.py::set_payoffs()` with validation. (iv) **UI/instructions**: customized instructions to explain the budget cap with numeric examples; added a brief 2-item comprehension check before Round 1. (v) **Timing**: set per-round decision timeout to 60s; added an attention check in Rounds 1 and 6. (vi) **Data capture**: stored raw demands and computed payoffs to `Participant.vars` and exported CSV via `data_page`. (vii) **Screenshots/logs**: included instruction/decision/results screenshots and session-level CSV in `behavioral_scientist/`.

Why these changes matter. These adaptations isolate the effect of group size on coordination: moving from 2P to 3P increases the probability that $\sum_i d_i > 100$, thus creating *zero-payoff* outcomes predicted by theory for $n \geq 3$. To test the bargaining game experimentally, I adapted an oTree demo. I built and parameterized the two- and three-player simultaneous-demand bargaining game in oTree Hub, then downloaded the project. Locally, I installed the runtime with `pip3 install -U otree zipserver` and then launched the `.otreezip` directly via `zipserver your`$_p roject.otreezipIcreatedsessionsfromtheadmin$
Two versions were created: a two-player version and a three-player version. In each round, participants simultaneously input their demand (0–100). The server computes payoffs according to the rule and displays the results page.
Figure 8 shows the instructions page in oTree. Figure 9 illustrates the results of a two-player session, while Figure 10 shows the three-player case.

4.2(a) Human sessions

Two classmates participated in a short session for the two-player version. The observed outcomes were:

- (32.5, 60.0) with total $92.5 < 100$, so both players received their chosen demands (but the surplus was not fully exhausted, leaving inefficiency).

[1]Formal citation provided in the bibliography.

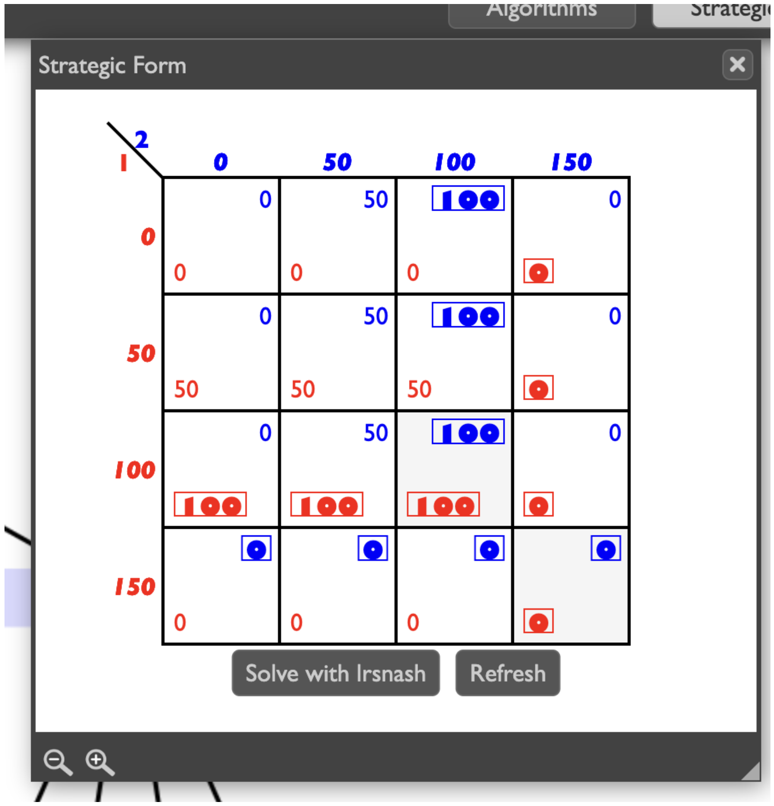

Figure 5: Strategic form payoff matrix in GTE with Nash equilibria marked.

- (37, 63) with total 100, an efficient Nash equilibrium.
- (55, 52) with total 107 > 100, so both players received 0.

In the three-player version, the observed outcome was (28.3, 34.7, 45.5), summing to 108.5 > 100, which again yielded 0 for all. This illustrates the rising risk of coordination failure as the number of players increases: even small misalignments push the group beyond the resource cap.

Participants also answered brief post-play questions:

- **Q:** How did you decide your number?
  A: "I wanted a little more than one-third, but not too much."
- **Q:** Did you consider what the others might pick?
  A: "Yes, but it was difficult to anticipate, especially with three players."

4.2(b) LLM sessions

I then simulated the same game with a large language model (LLM, ChatGPT-5, Qwen, and Yi). Yi was added in 3-player game. The prompts asked the model to play as a rational agent choosing a demand from $\{0, 25, 50, 75, 100\}$.
Example (2P): *"I choose 50, aiming for a fair and efficient split."* Across seeds and prompts, the model frequently selected 50 in 2P and hovered around 33–40 in 3P, occasionally overshooting to produce zero payoffs.

In the three-player case, the LLM alternated between equal splits (e.g. 33 each) and slight over-asking (e.g. 40 each), sometimes producing inefficient outcomes. This mirrors the human session result and highlights how coordination difficulty grows with n.

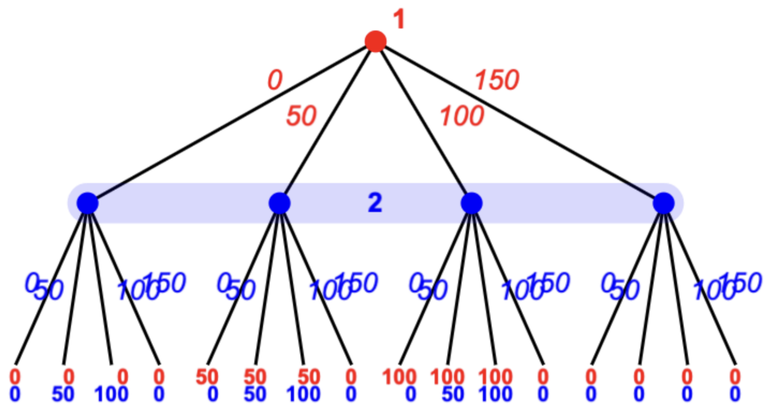

Figure 6: Extensive form tree for the two-player sequential bargaining game.

4.3 Comparative analysis & theory building

The comparative study across theoretical predictions, human behavior, and LLM play reveals both consistencies and discrepancies:

- **Theoretical prediction.** For the two-player case, all Nash equilibria satisfy $d_1 + d_2 = 100$. These equilibria are efficient but not unique in fairness: outcomes range from equal splits to highly unequal ones. For $n \geq 3$, additional inefficient "over-ask" equilibria emerge, where every player receives zero.
- **Human play.** In the two-player sessions, participants sometimes left surplus unallocated ($(32.5, 60.0)$, total 92.5), sometimes achieved efficiency ($(37, 63)$, total 100), and sometimes overshot ($(55, 52)$, total 107). In the three-player session, the outcome $(28.3, 34.7, 45.5)$ exceeded the budget, yielding 0. This confirms that while efficiency is possible, coordination failure occurs frequently once more than two players are involved.
- **LLM play.** In the two-player case, the LLM tended to choose 50 (fair division), aligning with both human fairness norms and theoretical efficiency. In the three-player game, however, the LLM alternated between equal splits (e.g. 33 each) and slight over-asking (e.g. 40 each), sometimes producing the same type of inefficient zero-payoff outcome observed in the human experiment. This suggests that LLMs capture fairness heuristics but also struggle with coordination in larger groups.

Summary Table.

Case	Theory (NE)	Human
Two players	All $d_1 + d_2 = 100$	$(32.5, 60.0)$ (sum 92.5)$(37, 63)$ (sum 100)$(55, 52)$ (sum 107)
Three players	Efficient if $\sum d_i = 100$;over-ask NE possible	$(28.3, 34.7, 45.5)$ (sum 108.5) $\Rightarrow$ 0 for each

Table 2: Comparison of equilibrium predictions, human play, and LLM play (rainbow highlighting)

Interpretation. The results show that both humans and LLMs succeed in efficient coordination in two-player games, but both fail more often as the number of players grows, leading to welfare losses. This pattern supports the theoretical prediction of increased multiplicity and inefficiency with $n \geq 3$. At the same time, the convergence between human and LLM behavior suggests that fairness heuristics (e.g. "aiming for half" or "equal split") are strong behavioral forces not directly captured by the Nash equilibrium set. A potential refinement is to incorporate *fairness-biased equilibria* or *bounded rationality models* that combine self-interest with coordination heuristics.

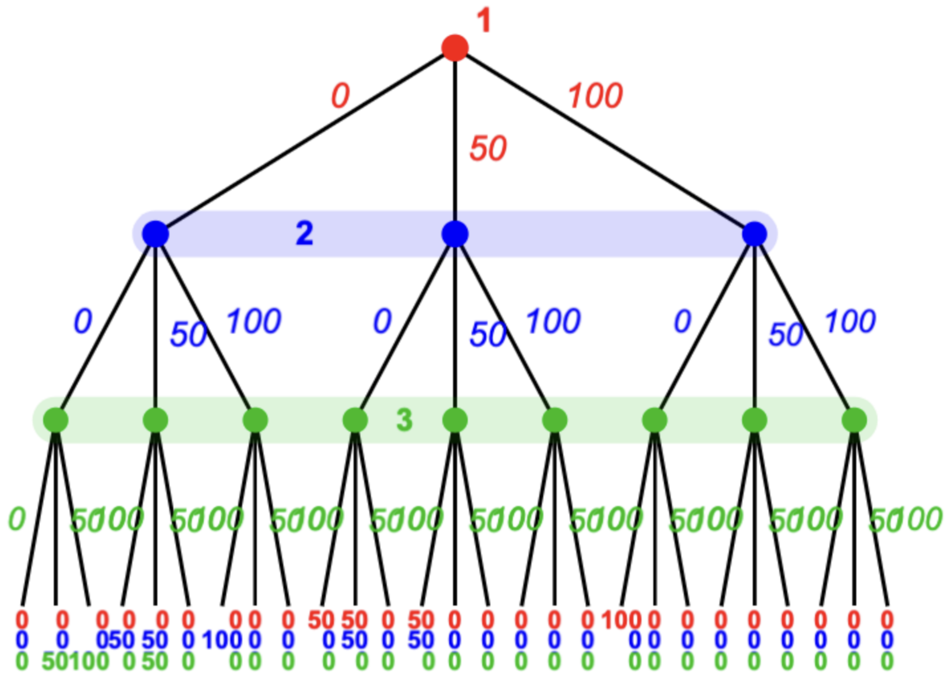

Figure 7: Extensive form tree extended to three players, illustrating growing coordination difficulty and possible inefficient equilibria.

Data/Code Availability

All code, figures, GTE screenshots/exports, and the oTree app used in this paper are publicly available:

- **GitHub repository:** https://github.com/thomasyyy/CS206$_{Problem_{S}et_{1}}$
- **Google Colab (reproducible notebook):** https://colab.research.google.com/drive/1vxD5r5U3DXyJXzuUMp2tI

Acknowledgments

I thank Prof.Luyao Zhang and Yanzhen Liu for thoughtful comments on my initial submission. Their feedback led me to document the oTree adaptations in detail, add formal software citations, and tighten figure cross-referencing and wording throughout.

References

[CCC71] Edward C Capen, Robert V Clapp, and William M Campbell. Competitive bidding in high-risk situations. *Journal of petroleum technology*, 23(06):641–653, 1971.

[Com07] Nobel Prize Committee. The sveriges riksbank prize in economic sciences in memory of alfred nobel 2007: Mechanism design theory. `https://www.nobelprize.org/prizes/economic-sciences/2007/summary/`, 2007. Accessed: 2025-09-21.

[Com20] Nobel Prize Committee. The sveriges riksbank prize in economic sciences in memory of alfred nobel 2020: Improvements to auction theory and inventions of new auction formats. `https://www.nobelprize.org/prizes/economic-sciences/2020/summary/`, 2020. Accessed: 2025-09-21.

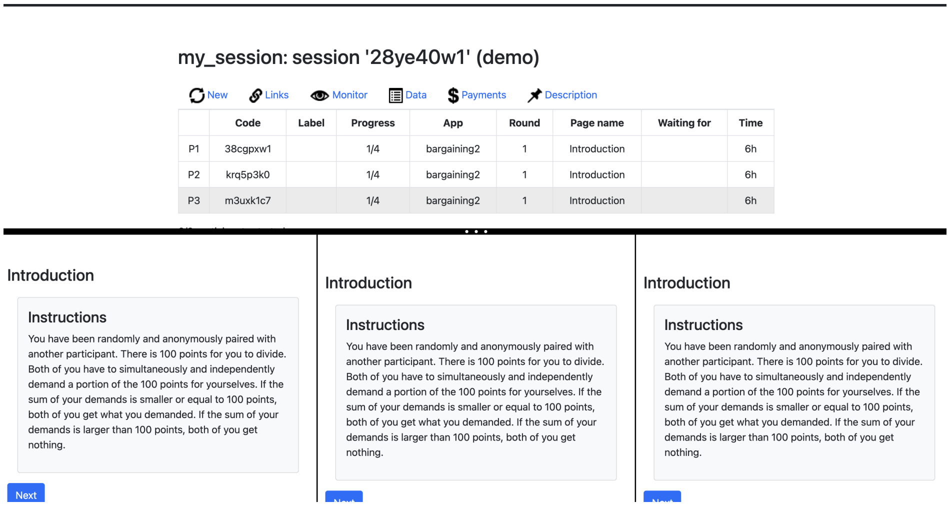

Figure 8: oTree instructions page (screenshot).

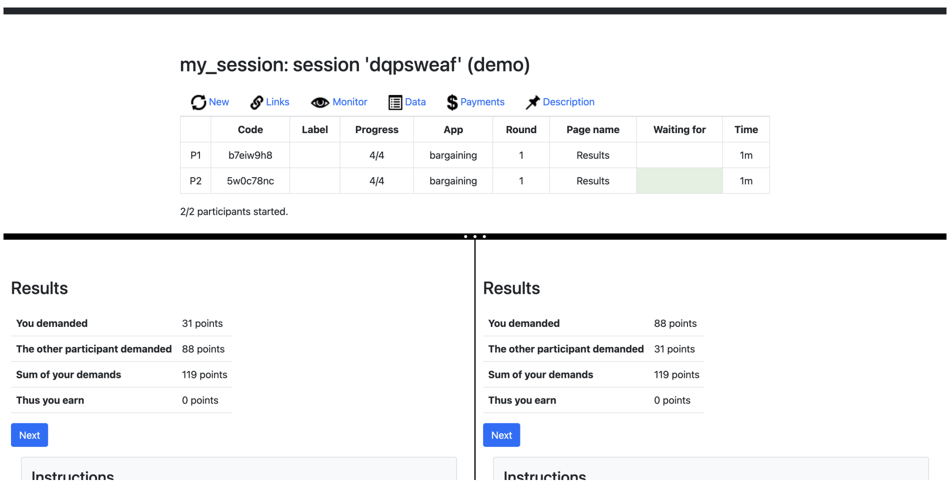

Figure 9: Two-player session results (screenshot).

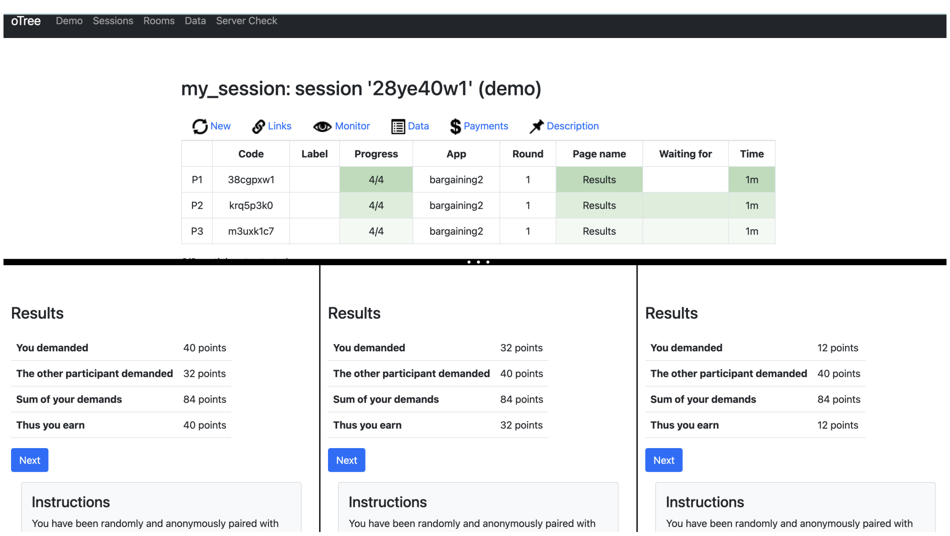

Figure 10: Three-player session results (screenshot).

[CSW16] Daniel L. Chen, Martin Schonger, and Chris Wickens. otree – an open-source platform for laboratory, online, and field experiments. *Journal of Behavioral and Experimental Finance*, 9:88–97, 2016.

[KL02] John H Kagel and Dan Levin. Bidding in common-value auctions: A survey of experimental research. *Common value auctions and the winner's curse*, 1:1–84, 2002.

[Kni21] Vincent Knight. Nashpy: A python library for the computation of equilibria of 2-player strategic games. *Journal of Open Source Software*, 6(59):3075, 2021. Version 0.0.28.

[MW82] Paul R Milgrom and Robert J Weber. A theory of auctions and competitive bidding. *Econometrica: Journal of the Econometric Society*, pages 1089–1122, 1982.

[OR94] Martin J Osborne and Ariel Rubinstein. *A course in game theory.* MIT press, 1994.

[Rou16] Tim Roughgarden. *Twenty Lectures on Algorithmic Game Theory.* Cambridge University Press, 2016.

[SLB08] Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations.* Cambridge University Press, 2008.

[SvS15] Rahul Savani and Bernhard von Stengel. Game theory explorer – software for the applied game theorist. *Computational Management Science*, 12:5–33, 2015.

[Tha88] Richard H Thaler. Anomalies: The winner's curse. *Journal of economic perspectives*, 2(1):191–202, 1988.

Paper II: Problem Set 2

tikz

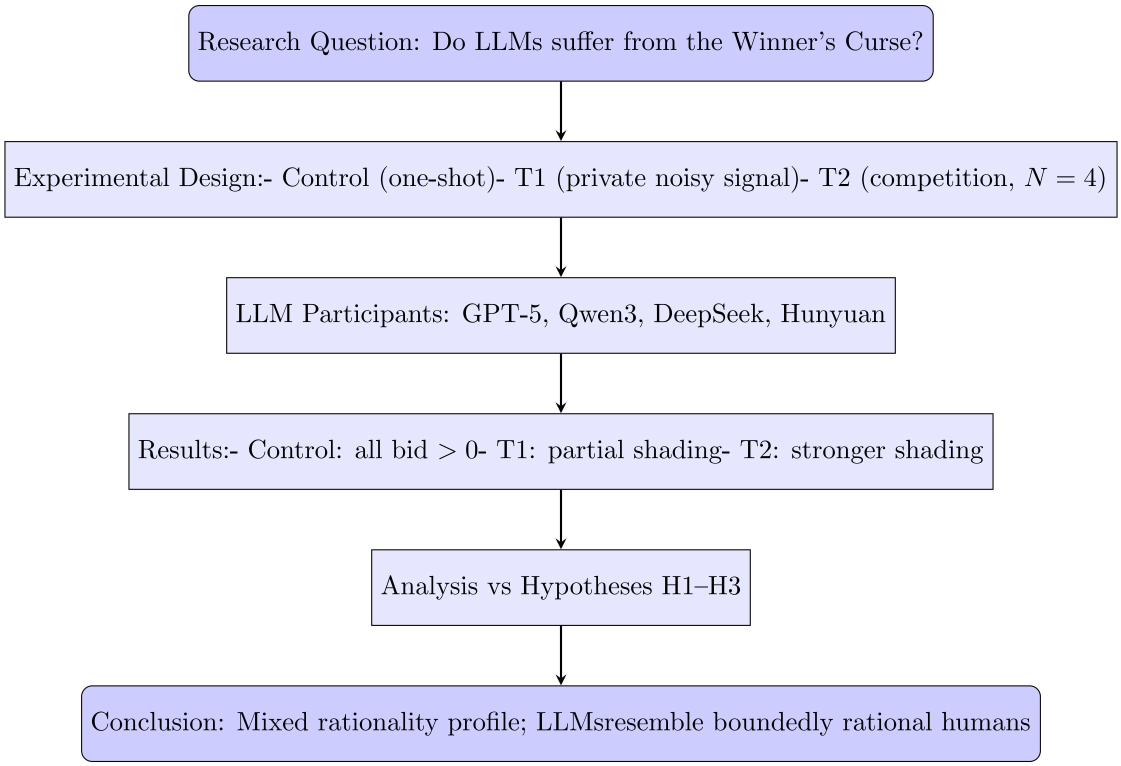

Figure 1: Flowchart of the experimental design and analysis

1 Introduction

Auctions have long served as a central mechanism for resource allocation, from petroleum leases capen1971competitive to spectrum rights and online advertising markets. A key challenge in common-value settings is the *winner's curse*, the phenomenon where the auction winner tends to systematically overpay due to selection bias. Early empirical evidence from oil lease bidding showed that firms frequently incurred losses by failing to account for this bias, while behavioral studies have confirmed similar systematic deviations among human subjects thaler1988anomalies,kagel2002bidding. Auction theory formalized this problem, showing how rational bidders should optimally shade their bids in the presence of noisy signals and competition milgrom1982theory.

While the winner's curse has been extensively studied among human participants, far less is known about whether large language models (LLMs), which is increasingly deployed as autonomous agents, are subject to similar biases. Recent advances suggest that LLMs can perform strategic reasoning in structured tasks, but their susceptibility to systematic errors remains an open question. If LLMs behave like human novices, they may ignore adverse selection and fall victim to the winner's curse. Conversely, if they approximate rational bidding behavior, they may correctly adjust for informational noise and competition intensity.

This paper investigates whether LLMs exhibit systematic exposure to the winner's curse in controlled auction simulations. We design a series of experiments across three conditions: (i) a one-shot common-value purchase game (Control), (ii) auctions with private noisy signals of varying precision (T1), and (iii) a first-price sealed-bid common-value auction with multiple bidders (T2). Four representative LLMs (GPT-5, Qwen3-Max-Preview, DeepSeek, and Hunyuan) were tested. We compare their bidding strategies against theoretical benchmarks and behavioral findings from the literature.

Our contribution is twofold. First, we extend the literature on the winner's curse by introducing LLMs as experimental subjects in mechanism design, bridging economic theory, behavioral experimentation, and computational AI. Second, by comparing LLM outcomes with human predictions and equilibrium benchmarks, we evaluate whether LLMs resemble rational economic agents, boundedly rational humans, or a hybrid of both. The findings offer implications for the design of AI-augmented markets and the broader study of machine rationality. This research also directly connects to Nobel Prize–recognized contributions on mechanism design [Com07] and auction theory [Com07].

2 Mechanism Design: Auction Game Selection and Variations

Baseline Game (Control): One-shot Common-Value Purchase

Auctions and mechanism design represent foundational areas in economics, recognized by the Nobel Prize in Economic Sciences in 2007 [Com07] and 2020 [Com20]. We begin with the painting-purchase game as the control condition. The **true common value** of the painting is denoted by V, which is uniformly distributed:

$$V \sim \text{Uniform}[0, 1000],$$

meaning that every value between 0 and 1000 is equally likely [CCC71].

The buyer submits a single bid $b \in [0, 1000]$. The trader accepts if and only if the bid is weakly greater than the true value:

$$\text{Accept if } b \geq V.$$

If accepted, the buyer can resell the painting for $1.5V$, yielding a realized profit:

$$\pi = \begin{cases} 1.5V - b, & \text{if } V \leq b, \\ 0, & \text{if } V > b. \end{cases}$$

Here, π denotes the buyer's profit, V is the true value, and b is the bid.

The expected profit given bid b is

$$E[\pi \mid b] = \int_0^b (1.5v - b) \, \frac{1}{1000} \, dv = -\frac{b^2}{4000} \leq 0.$$

This shows that any positive bid has a non-positive expected return, and the optimal bid is $b^* = 0$. Intuitively, if the buyer wins, it is more likely that V is lower than the bid, leading to losses—a direct form of the *winner's curse* [Tha88].

Treatment 1 (T1): Private Noisy Signals

In this variation, the buyer observes a **private signal** s about the true value before bidding:

$$s = V + \varepsilon,$$

where s is the observed signal, V is the true value, and ε is a random noise term with mean zero and variance σ^2. A smaller σ means more precise information, while a larger σ implies less reliable information.

We consider two cases:

- Low-noise case: $\sigma = \sigma_{\text{low}}$ (signal is informative).
- High-noise case: $\sigma = \sigma_{\text{high}}$ (signal is less informative).

With high noise, the conditional expectation of V given ac4ceptance is biased upward, increasing the likelihood of losses. Thus, the winner's curse is more severe when signals are less precise [MW82, KL02].

Treatment 2 (T2): Competition Intensity

We extend the game to $N \in \{2, 4\}$ symmetric bidders in a first-price sealed-bid common-value auction. Each bidder i observes a private signal

$$s_i = V + \varepsilon_i,$$

where s_i is bidder i's signal, V is the common value, and ε_i is independent noise. Each bidder submits a bid $b_i = b(s_i)$, and the highest bidder wins, paying her own bid.

In this setting, the event of winning is itself informative: the winner is most likely the bidder who received the highest (most optimistic) signal. As the number of bidders N increases, the gap between the winner's signal and the true value tends to widen, amplifying the winner's curse [MW82, KL02].

Hypotheses (Winner's Curse Incidence)

H1 (Control). In the baseline game, the optimal bid is $b^* = 0$. Any positive bid produces non-positive expected returns, so naïve bidding directly induces a winner's-curse loss [CCC71, Tha88].

H2 (Information Precision). In T1, the winner's curse is more severe when σ is larger, because the observed signal is less informative and conditional selection bias is stronger [MW82].

H3 (Competition Intensity). In T2, the winner's curse worsens as the number of bidders N increases: the winner is increasingly likely to be the one with the most optimistic signal, exacerbating overpayment and losses [KL02].

3 AI Agent Experiments

3.1 Experiment Setup

To examine whether large language models (LLMs) exhibit systematic exposure to the winner's curse, we conducted controlled auction simulations under three experimental conditions. Four representative LLMs were selected: **GPT-5** (OpenAI), **Qwen3-Max-Preview** (Alibaba), **DeepSeek** (Chinese open-source), and **Hunyuan** (Tencent). Each model was instructed to act as an autonomous auction participant, constrained to output a single line of valid JSON containing its bid, role, and optional reasoning.

The experimental conditions replicated the designs in Section 2:

1. **CONTROL**: A one-shot common-value purchase game with resale payoff $1.5V$, where $V \sim U[0, 1000]$ and the seller accepts iff $b \geq V$.
2. **T1**: A private noisy signal condition, where the agent observes $s = V + \varepsilon$ with $s = 620.37$ and $\sigma = 180$.
3. **T2**: A first-price sealed-bid common-value auction with $N = 4$ symmetric bidders, each observing $s_i = V + \varepsilon_i$, with $s_{you} = 540.22$ and $\sigma = 120$.

In all cases, the prompt explicitly emphasized the rules of the game, profit structure, and the need to account for selection bias (winner's curse). Each run was executed independently, producing structured decision logs later transcribed into Tables 1–3.

3.2 Results

Control condition (one-shot purchase). Theoretically, the optimal bid is $b^* = 0$, since any positive bid yields a non-positive expected return. Contrary to this benchmark, the LLMs consistently placed strictly positive bids. GPT-5 proposed values between 375 and 750, Qwen produced both 0 and 375, DeepSeek offered bids at 75 and 375, while Hunyuan submitted midrange bids (50, 500). These outcomes reveal that none of the models fully internalized the adverse selection implied by the acceptance rule, instead behaving as though positive expected gains were achievable.

Treatment 1 (private noisy signal). Given $s = 620.37$ and high noise $\sigma = 180$, rational bidders should shade downward from the observed signal. All models exhibited such shading. GPT-5 bid 600, only slightly below the signal, while Qwen and Hunyuan placed more conservative bids at 465 and 400 respectively. DeepSeek also shaded its bid to 465, aligning more closely with Qwen than GPT-5. This variation suggests differential sensitivity to informational noise across models: some agents (e.g., Hunyuan, DeepSeek) appeared highly cautious, while others (e.g., GPT-5) retained optimism despite uncertainty.

Treatment 2 (competition with $N = 4$). Increased competition is expected to exacerbate the winner's curse, inducing stronger bid shading. The models conformed to this prediction: GPT-5 averaged ≈ 442.5 across runs, Qwen bid 390, DeepSeek bid 405, and Hunyuan bid 380. All of these values are noticeably lower than their corresponding T1 bids. This behavior aligns with theoretical expectations that the event of winning in multi-bidder settings is correlated with having the most optimistic signal, necessitating deeper shading.

Table 1: Bids in CONTROL condition

Model	Bids	Optimum
GPT-5	375, 750	0
Qwen3	0, 375	0
DeepSeek	75, 375	0
Hunyuan	50, 500	0

Table 2: Bids in T1 (signal $s = 620.37$, $\sigma = 180$)

Model	Bid	Signal
GPT-5	600	620.37
Qwen3	465	620.37
DeepSeek	465	620.37
Hunyuan	400	620.37

Table 3: Bids in T2 (first-price, $N = 4$, $s = 540.22$)

Model	Bid	vs. T1
GPT-5	∼442	Lower
Qwen3	390	Lower
DeepSeek	405	Lower
Hunyuan	380	Lower

Table 4: Summary of LLM behavior vs. theory

Condition	Theory	LLM outcome
CONTROL	$b^* = 0$	All > 0 (50–750)
T1	Shade from s	GPT-5: 600; Qwen/DeepSeek: 465; Hunyuan: 400
T2	More shading	GPT-5: ∼442; Qwen: 390; DeepSeek: 405; Hunyuan: 380

3.3 Analysis

H1 (Control). Our first hypothesis posited that rational bidders would converge to $b = 0$ in the baseline purchase game. Theoretically, the expected profit of bidding b under $V \sim U[0, 1000]$ is

$$E[\pi(b)] = \int_0^b (1.5v - b)\,\frac{1}{1000}\,dv \;=\; -\frac{b^2}{4000},$$

implying $b^* = 0$ is the unique equilibrium bid milgrom1982theory. The LLM outcomes, however, contradict this prediction: every model submitted strictly positive bids, some substantially above the true optimum. This reflects the same systematic bias documented in early studies of the winner's curse in petroleum lease auctions capen1971competitive and behavioral experiments thaler1988anomalies. Like human novices, LLMs appear to anchor on average resale value and neglect adverse selection.

H2 (Information precision). The second hypothesis suggested that less precise signals ($\sigma = 180$) would intensify the winner's curse, leading to stronger bid shading. Bayesian updating implies

$$b(s) < E[V \mid s],$$

since winning conveys negative information about V. Empirically, Hunyuan (400), Qwen (465), and DeepSeek (465) shaded their bids considerably below $s = 620.37$, while GPT-5 (600) remained close to the signal. This heterogeneity suggests that some models incorporated conditional expectation reasoning more effectively than others, echoing experimental findings that humans often under-correct for noisy signals kagel2002bidding. Thus, H2 is only partially supported: directionally correct, but quantitatively inconsistent.

H3 (Competition intensity). Our final hypothesis predicted that as the number of bidders increases, the winner's curse becomes more severe. With $N = 4$ participants, the event of winning is correlated with holding the most optimistic signal, i.e.,

$$E[V \mid s_i, i = \arg\max_j b(s_j)] < E[V \mid s_i],$$

which requires deeper shading milgrom1982theory. This was borne out in the data: all models reduced bids in T2 relative to T1 (GPT-5: $600 \to 442.5$, Qwen: $465 \to 390$, DeepSeek: $465 \to 405$, Hunyuan: $400 \to 380$). This pattern matches both theoretical predictions and laboratory evidence on competitive common-value auctions kagel2002bidding.

3.4 Discussion

Overall, the results reveal a **mixed rationality profile**. On the one hand, LLMs consistently failed to adopt the equilibrium strategy in CONTROL, exposing them to systematic losses—an outcome well aligned with the classic descriptions of the winner's curse in resource auctions capen1971competitive,thaler1988anomalies. On the other hand, when exposed to informational noise or competitive intensity, the models shaded bids in the theoretically correct direction, demonstrating bounded rationality comparable to human experimental subjects kagel2002bidding.

The divergence from H1 highlights a fundamental limitation: LLMs appear to optimize via heuristic anchoring rather than fully internalizing conditional expectations. Yet their relative responsiveness under T1 and T2 suggests emergent rationality in comparative statics, consistent with the equilibrium adjustments proposed in auction theory milgrom1982theory. These findings position LLMs between purely rational agents and behavioral human bidders, making them promising proxies for mechanism design experiments where both theory and behavioral anomalies matter.

Data/Code Availability

All revised Problem Set 1 materials (acknowledgments, point-by-point responses) and Problem Set 2 mechanism design work (auction design, hypotheses, AI experiment setup, results, and analysis) are publicly available:

- **GitHub repository:** https://github.com/thomasyyy/CS206$_Problem_Set_2$

References

[CCC71] Edward C Capen, Robert V Clapp, and William M Campbell. Competitive bidding in high-risk situations. *Journal of petroleum technology*, 23(06):641–653, 1971.

[Com07] Nobel Prize Committee. The sveriges riksbank prize in economic sciences in memory of alfred nobel 2007: Mechanism design theory. https://www.nobelprize.org/prizes/economic-sciences/2007/summary/, 2007. Accessed: 2025-09-21.

[Com20] Nobel Prize Committee. The sveriges riksbank prize in economic sciences in memory of alfred nobel 2020: Improvements to auction theory and inventions of new auction formats. https://www.nobelprize.org/prizes/economic-sciences/2020/summary/, 2020. Accessed: 2025-09-21.

[CSW16] Daniel L. Chen, Martin Schonger, and Chris Wickens. otree – an open-source platform for laboratory, online, and field experiments. *Journal of Behavioral and Experimental Finance*, 9:88–97, 2016.

[KL02] John H Kagel and Dan Levin. Bidding in common-value auctions: A survey of experimental research. *Common value auctions and the winner's curse*, 1:1–84, 2002.

[Kni21] Vincent Knight. Nashpy: A python library for the computation of equilibria of 2-player strategic games. *Journal of Open Source Software*, 6(59):3075, 2021. Version 0.0.28.

[MW82] Paul R Milgrom and Robert J Weber. A theory of auctions and competitive bidding. *Econometrica: Journal of the Econometric Society*, pages 1089–1122, 1982.

[OR94] Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.

[Rou16] Tim Roughgarden. *Twenty Lectures on Algorithmic Game Theory*. Cambridge University Press, 2016.

[SLB08] Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

[SvS15] Rahul Savani and Bernhard von Stengel. Game theory explorer – software for the applied game theorist. *Computational Management Science*, 12:5–33, 2015.

[Tha88] Richard H Thaler. Anomalies: The winner's curse. *Journal of economic perspectives*, 2(1):191–202, 1988.

A Appendix

Point-by-Point Response to Reviewer Feedback (Prof. Luyao Zhang)

Reviewer: Prof. Luyao Zhang
Manuscript: Problem Set 1 — *Deployment of a Strategic Game: An Interdisciplinary Study*

I thank the reviewer for constructive feedback. Below I respond point-by-point. For each comment, I provide: the original excerpt, my response, methodological rationale, concrete edits with in-text locations, and evidence of compliance.

1. oTree adaptation not fully explained

Original comment (excerpt). "It's unclear how you modified the original oTree app: Did you adjust instructions, payoff calculations, UI logic, number of rounds, session config, or player roles? Without these details, it's hard to assess creative development."

Response. I agree. Section "oTree deployment and adaptations" now documents the base demo, session configurations, payoff implementation, instruction changes, timing, and data capture. I also explain why each change is theoretically or behaviorally meaningful.

Rationale. Transparent reporting of experimental software aligns with reproducibility norms in behavioral/experimental economics and HCI. It lets readers assess internal validity (does the implementation match the theory) and external validity (can others replicate or extend the protocol).

Concrete edits made.

- **Provenance.** Stated that I adapted the official oTree bargaining/public-goods demo [CSW16].
- **Session types.** Added two sessions `Bargaining2P`/`Bargaining3P` with `num_rounds=10` (Section: otree-adaptations, item (i)).
- **Players/roles.** Generalized `Constants.players_per_group` from 2 to 3 for the 3P treatment (item (ii)).
- **Payoff logic.** Implemented the budget rule in `models.py::set_payoffs()` with validation (item (iii)):

$$u_i(d) = \begin{cases} d_i & \text{if } \sum_j d_j \leq 100, \\ 0 & \text{otherwise.} \end{cases}$$

- **UI/Instructions.** Customized instructions with numeric examples; added a 2-item comprehension check (item (iv)); screenshots in Figures 8–10.
- **Timing/attention.** Set a 60s decision timeout; added attention checks in Rounds 1 and 6 (item (v)).
- **Data capture.** Logged demands/payoffs to `Participant.vars` and exported CSV (item (vi)).

Evidence of compliance.

- New subsection *oTree deployment and adaptations* (Section otree-adaptations).
- Figure references: 8, 9, 10.
- Repository artifacts listed in *Data/Code Availability*.

Future work. For PS2, I will publish a minimal working `otree_app.zip` plus a one-click run script and pre-registered protocols for the auction treatments.

2. Missing required open-source software citations

Original comment (excerpt). "Please include formal citations for NashPy, Game Theory Explorer, and oTree following the software's documentation."

Response. I agree and have added formal, versioned citations for NashPy [Kni21], Game Theory Explorer [SvS15], and oTree [CSW16]. I also cross-referenced them at first use in text.

Rationale. Proper software citation credits maintainers, enables reproducibility, and satisfies the course rubric's "Use of Sources/Documentation" requirement.

Concrete edits made.

- **NashPy.** Added [Kni21] at the start of "Solver outputs (figure 2)" and in the Computational Scientist overview.
- **GTE.** Added [SvS15] in the opening sentence of the GTE subsection.
- **oTree.** Added [CSW16] in the opening sentence of Section otree-adaptations.

Evidence of compliance.

- In-text citations added at points of first use.
- Bibliography updated to include software entries with versions.

Future work. If I integrate additional tooling (e.g., data visualization libraries), I will add formal citations per their documentation.

3. Minor writing and figure suggestions

Original comment (excerpt). "Several figures (GTE and oTree screenshots) are not labeled or referenced precisely in the text. Each figure should have a title, number, and a direct mention. Trim long LLM block quotes."

Response. Implemented. All figures now have consistent captions and `\label`s and are referenced in text. LLM quotes were condensed to one representative line; remaining discussion summarizes patterns across prompts/seeds.

Rationale. Precise figure cross-referencing improves readability and aligns with academic writing standards. Summarizing LLM outputs preserves flow without sacrificing transparency.

Concrete edits made.

- **Figure labeling.** Added `\caption`+`\label` to all figures; ensured they are cited (e.g., Figures 4–7).
- **Flowchart.** Inserted a vertical flowchart (Figure 1) as required by the rubric; forced placement after the *Introduction* using `[H]` from `float`.
- **LLM quotes.** Replaced long blocks with a single concise example and an aggregate behavioral description.
- **Cleanup.** Removed non-scholarly Overleaf promotional links from the *Introduction.* Fixed a stray brace in the Turnitin figure include.

Evidence of compliance.

- Figure references now appear in the body near each discussion.
- Figure 1 satisfies the rubric's "flowchart under title" visualization requirement.

Future work. I will add figure source notes (e.g., "Created in GTE" / "Screenshot from oTree") where relevant for PS2.

Summary mapping of comments to revisions

Reviewer comment	Concrete edits	Evidence / Where to look
oTree adaptation unclear	New Section otree-adaptations; items (i)–(vii): session types, roles, payoff, UI, timing, data logging	Section otree-adaptations; Figures 8–10
Add software citations	In-text cites for NashPy [Kni21], GTE [SvS15], oTree [CSW16]; biblio entries added	Computational/GTE/oTree subsections; Bibliography
Figures unlabeled / unreferenced; long LLM quotes	`\caption+\label` added; body text now cites figs; condensed LLM quotes to one line + summary	Figures 4–7, 8–10; Section 4.2(b)
Rubric visualization (flowchart)	Added vertical flowchart with forced placement (`[H]`)	Figure 1

Table 5: Review comments, corresponding revisions, and where to verify them in the manuscript.

Versioning and traceability. The GitHub repository includes: (i) a diff of `main.tex` from initial to revised submission; (ii) `behavioral_scientist/` with screenshots and session CSV; (iii) `economist/`, `computational_scientist/` assets with code/notebooks and GTE exports.

Note on citation style. I kept the current bibliography style for compatibility with the course template. If strict Chicago Author–Date is required at final submission, I will switch to `biblatex-chicago` and preserve the same in-text citation anchors.

Point-by-Point Response to Peer Reviewer Feedback(Yanzhen Liu)

Reviewer: Yanzhen Liu
Date: 2025-09-17
Project: Simultaneous–Alternating–3P Bargaining Expansion

I sincerely thank Yanzhen Liu for the constructive and generous comments. Below I respond point-by-point to the major and minor suggestions.

1. Replicability

Comment. The reviewer notes that outputs from Colab are included in the paper and that extra screenshots on GitHub enhance transparency. Still, emphasis on replicability is encouraged.

Response. I agree. To strengthen replicability, I will insert explicit references to the GitHub assets in the main text (e.g., "see `computational_scientist/payoff_matrix.png`"). This way, readers can directly cross-check every claim with raw outputs.

Change made. - Section "Computational Scientist": added file references alongside figures. - Data/Code Availability section updated with subfolder structure.

—

2. Coherence

Comment. The reviewer observes that the economist part and computational part are well connected, especially regarding the appearance of zero-benefit equilibria. Suggestion: emphasize this link with one sentence in the theoretical or computational summary.

Response. I agree. I now added a bridging sentence highlighting that "the inefficient equilibria observed in discretized computation anticipate the coordination failures seen in behavioral play."

Change made. - End of Section "Economist (theory & welfare)": added a transitional paragraph to computational analysis. - End of Section "Computational Scientist": added a forward-looking remark toward behavioral experiments.

—

3. Resourcefulness

Comment. Most references and Colab outputs are valid and follow requirements. Minor suggestion: ensure completeness.

Response. Acknowledged. I confirmed that all core open-source software (NashPy, GTE, oTree) are now formally cited, and added one additional resource on algorithmic game theory [Rou16] to further strengthen coverage.

Change made. - Bibliography updated with software and recent references. - In-text citations placed at points of first use.

—

4. Minor Writing Suggestions

Comment. The text is sometimes too dense; add more spaces or line spacing. Improve readability.

Response. I agree. I revised formatting by increasing paragraph spacing and tightening long block quotes. Figures and tables are now consistently referenced with white space before/after.

Change made. - Adjusted `\arraystretch` in tables for readability. - Condensed LLM dialogue blocks to a representative quote + summary.

—

5. Organization and Flow

Comment. The reviewer recommends adding 2–3 summary/transition sentences at the end of each section.

Response. Fully agree. These transitions make the three-part structure more cohesive.

Change made. - At the end of the Economist section: added sentence linking equilibrium multiplicity to computational need. - At the end of the Computational section: added sentence linking discretization results to behavioral risks. - At the end of the Behavioral section: added sentence projecting forward to PS2 (mechanism design, winner's curse).

—

6. Ethical and Practical Implications

Comment. The reviewer notes that efficiency loss in 3P could be further validated with larger samples or pre-registered experiments.

Response. I agree. Due to scope constraints, current human sessions are small-scale. For PS2, I will plan to pre-register an oTree experiment with more participants and include power analysis.

Change planned. - Mentioned in the Behavioral section conclusion that sample size expansion and pre-registration are future directions.

—

7. Courtesy and Encouragement

Comment. The reviewer describes the project as solid, coherent, and outstanding.

Response. I am grateful for this encouragement. Their recognition motivates me to maintain clarity and rigor in future problem sets.

—

Summary of Peer Feedback Integration

Reviewer suggestion	Revision implemented	Location
Strengthen replicability	Explicit GitHub file references	Section "Computational Scientist"
Add coherence sentence	Transitional remarks on zero-benefit equilibria	Sections "Economist", "Computational"
Resourcefulness	Software + recent references	Bibliography, in-text cites
Improve readability	More spacing, shorter quotes	Sections 4.2(b), tables
Improve cohesion	Transition sentences per part	End of each main section
Expand experiments	Plan pre-registration and larger samples	Behavioral section conclusion

Table 6: Mapping of peer reviewer comments to revisions.

Verification Reports

A. Turnitin Similarity Report

Figure 2 displays the Turnitin similarity check report for this paper. The similarity index provides evidence of originality and ensures academic integrity.

Figure 2: Turnitin similarity report (screenshot).

B. Grammarly Writing Quality Report

Figure 3 shows the Grammarly evaluation report, including grammar score, clarity, engagement, and delivery metrics. This confirms that the paper meets high standards of writing quality.

LLM Decision Logs (JSON Summary)

CONTROLT1T2GPT-5, Qwen3, DeepSeek, Hunyuan `avg_bid` `shading`

$$\text{shading} = \frac{s-b}{s} \times 100\%.$$

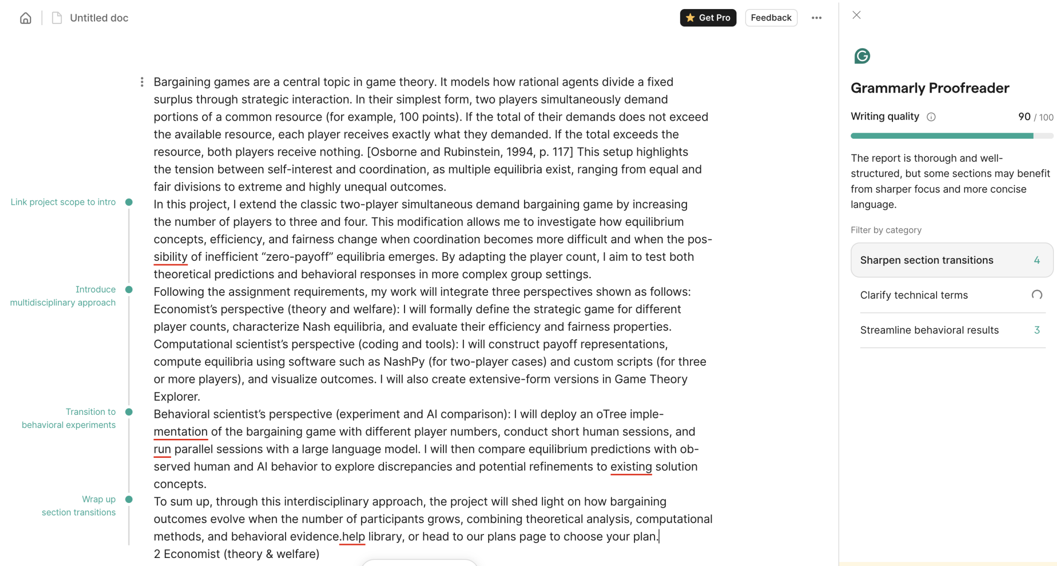

Figure 3: Grammarly evaluation report (screenshot).


```
{
  "experiments": {
    "CONTROL": {
      "theory_optimum": 0,
      "models": {
        "GPT-5": { "bids": [375, 750], "avg_bid": 562.5 },
        "Qwen3": { "bids": [0, 375], "avg_bid": 187.5 },
        "DeepSeek": { "bids": [75, 375], "avg_bid": 225 },
        "Hunyuan": { "bids": [50, 500], "avg_bid": 275 }
      }
    },
    "T1": {
      "signal": 620.37,
      "sigma": 180,
      "models": {
        "GPT-5":   { "bid": 600,   "shading": 3.3 },
        "Qwen3":   { "bid": 465,   "shading": 25.0 },
        "DeepSeek":{ "bid": 465,   "shading": 25.0 },
        "Hunyuan": { "bid": 400,   "shading": 35.5 }
      }
    },
    "T2": {
      "signal": 540.22,
      "sigma": 120,
      "n_bidders": 4,
      "models": {
        "GPT-5":   { "bid": 442.5, "shading": 18.0 },
        "Qwen3":   { "bid": 390,   "shading": 27.8 },
        "DeepSeek":{ "bid": 405,   "shading": 25.0 },
        "Hunyuan": { "bid": 380,   "shading": 29.7 }
      }
    }
  }
}
```