COMPSCI-ECON206 Problem Set 1

Ai Zhou

September 2025

1 Part 1

Subgame Perfect Nash Equilibrium

Definition (Paraphrase)

Let an extensive-form game with perfect information be denoted by

$$\Gamma = \langle N, H, P, (A(h))_{h \in H}, (u_i)_{i \in N} \rangle,$$

where

- $N = \{1, \ldots, n\}$ is the finite set of players.
- H is the set of histories (nodes), including the empty history $\emptyset$, with terminal histories $Z \subseteq H$.
- $P : H \setminus Z \to N$ assigns a player to each nonterminal history.
- $A(h)$ is the finite set of actions available after history h.
- Each player $i \in N$ has a payoff function $u_i : Z \to R$.
- A **strategy** s_i for player i is a complete contingent plan assigning an action in $A(h)$ for every history h with $P(h) = i$.
- A **strategy profile** is $s = (s_1, \ldots, s_n)$.

A **subgame** of Γ is defined as the restriction of Γ to any history h that constitutes a decision node not cutting across information sets.

[Subgame Perfect Nash Equilibrium] [pp. 93–95] Shoham2009, Rubinstein1994

A strategy profile s^* is a *Subgame Perfect Nash Equilibrium (SPNE)* if and only if, for every subgame $\Gamma(h)$ of Γ, the restriction $s^*|_h$ induces a Nash equilibrium of that subgame:

$$u_i(s^*|_h) \;\geq\; u_i(s_i, s^*_{-i}|_h), \quad \forall i \in N,\ \forall s_i \in S_i(h),\ \forall h \in H.$$

That is, no player has a profitable deviation in any subgame, not only in the original game.

Existence Theorem (Paraphrased)

For any finite extensive-form game with perfect information, there exists at least one strategy profile $s^* = (s_1^*, \ldots, s_n^*)$ such that s^* forms a subgame perfect Nash equilibrium. Formally,

$$\exists s^* \in S_1 \times \ldots \times S_n such that s^*|_h is a Nash equilibrium for every subgame \Gamma(h),\ \forall h \in H.$$

Proof Idea

The proof uses **backward induction**:

1. Start from the terminal nodes and determine each player's optimal action at the last decision nodes.
2. Replace each subgame by the payoff vector resulting from optimal play.
3. Recursively move backward in the tree, selecting actions that maximize the player's payoff at each node.
4. The resulting strategy profile is optimal in every subgame and thus forms a SPNE.

Intuition: The finiteness of the game tree guarantees that this backward-induction construction produces at least one SPNE in pure strategies.

Analytical Solution and Interpretation

Analytical Solution

In this extensive-form game with perfect information, the Subgame Perfect Nash Equilibrium (SPNE) identifies strategies where each player optimally responds in every possible subgame. Conceptually, SPNE refines the standard Nash equilibrium by requiring that no player has an incentive to deviate, even after unexpected moves by others. We determine the equilibrium using backward induction: starting from the terminal nodes, each player chooses the action that maximizes their payoff given subsequent optimal decisions. This process is repeated recursively until reaching the initial node, producing a complete strategy profile that specifies an action for every decision point. For example, player 1's SPNE strategy can be written as $s_1^* = (a_1, a_2, \ldots)$, while player 2's is $s_2^* = (b_1, b_2, \ldots)$. This approach guarantees consistency across all subgames and provides a clear prediction for rational play.

The SPNE outcome is not necessarily socially optimal. From a Pareto perspective, some SPNE strategies may leave all players worse off than alternative cooperative outcomes. Similarly, total welfare (the sum of players' payoffs) might not be maximized, indicating suboptimal utilitarian efficiency. Sequential play can also generate unequal outcomes, raising fairness concerns: one player may consistently earn more than others, violating equity or proportionality principles. Nonetheless, SPNE serves as a normative benchmark: it shows what rational players would do if they perfectly anticipate others' behavior, even if the outcome is not efficient or fair.

Interpretation

In practice, the Subgame Perfect Nash Equilibrium (SPNE) provides a strong prediction for how fully rational players with perfect information would behave in every subgame of a sequential interaction. However, its realism is limited because actual human decision-makers often face cognitive constraints and incomplete information, which may lead to deviations from SPNE predictions. Furthermore, many extensive-form games admit multiple SPNE, raising the question of equilibrium selection; observed behavior may depend on social norms, expectations, or pre-play communication. Refinements such as trembling-hand perfect equilibrium help address implausible strategies by eliminating those that rely on extremely unlikely mistakes. From a computational perspective, backward induction guarantees SPNE existence in finite games, but as the game tree grows in size or complexity, calculating SPNE becomes increasingly demanding, highlighting the practical relevance of algorithmic tools like GTE or NashPy. Thus, while SPNE offers a normative benchmark, its predictive accuracy is influenced by both bounded rationality and computational tractability.

2 Part 2 Computational ScientistsTrust Simple

u(A) = 100 – x + y

u(B) = 3x - y

Player A/B	0 %	50 %	100 %
0	(100,0)	(100,0)	(100,0)
50	(50,100)	(125,75)	(200,0)
100	(0,300)	(150,150)	(300,0)

Figure 1: Payoff matrix, made by Microsoft Word, exported as png.

```
Requirement already satisfied: nashpy in /usr/local/lib/python3.12/dist-packages (0.0.41)
Requirement already satisfied: numpy>=1.21.0 in /usr/local/lib/python3.12/dist-packages (from nashpy) (2.0.2)
Requirement already satisfied: scipy>=0.19.0 in /usr/local/lib/python3.12/dist-packages (from nashpy) (1.16.1)
Requirement already satisfied: networkx>=3.0.0 in /usr/local/lib/python3.12/dist-packages (from nashpy) (3.5)
Requirement already satisfied: deprecated>=1.2.14 in /usr/local/lib/python3.12/dist-packages (from nashpy) (1.2.18)
Requirement already satisfied: wrapt<2,>=1.10 in /usr/local/lib/python3.12/dist-packages (from deprecated>=1.2.14->nashpy) (1.17.3)
Normal-form Trust Game (Multiplier=3):
Bi matrix game with payoff matrices:

Row player:
[[100 100 100]
 [ 50 125 200]
 [  0 150 300]]

Column player:
[[  0   0   0]
 [150  75   0]
 [300 150   0]]

Nash Equilibria (pure and mixed strategies):
Player A strategy: [1. 0. 0.]
Player B strategy: [1. 0. 0.]
---
```

Figure 2: Nash Equilibria calculated by Nashpy using Google colab.

Interpretation (Google Colab)

The NashPy computation returns a pure strategy Nash Equilibrium where Player A chooses to invest 0 and Player B chooses to return 0. This means that, given Player B's strategy of returning nothing, Player A maximizes their payoff by investing nothing. Similarly, given Player A's investment of 0, Player B cannot improve their payoff by returning any positive amount. This result is fully consistent with the Subgame Perfect Nash Equilibrium (SPNE) derived via backward induction in the extensive-form game. In the one-round Trust Game, Player B's optimal action in every subgame is to return 0, and anticipating this, Player A's optimal choice is also to invest 0. Therefore, the SPNE coincides with the pure-strategy NE identified by NashPy.

Game Theory Explorer(GTE)

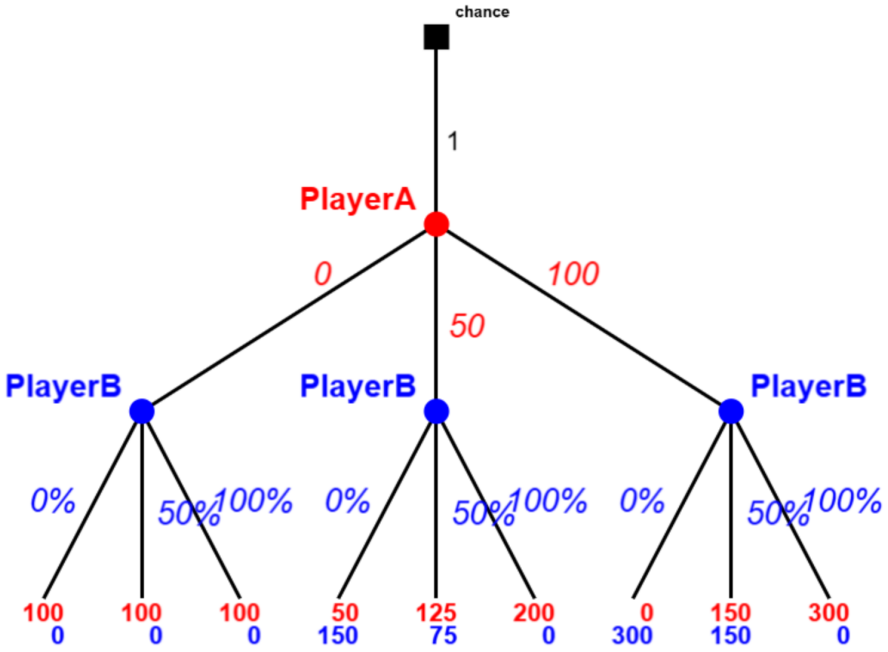

Figure 3: Extensive-form version of Trust Simple in GTE.

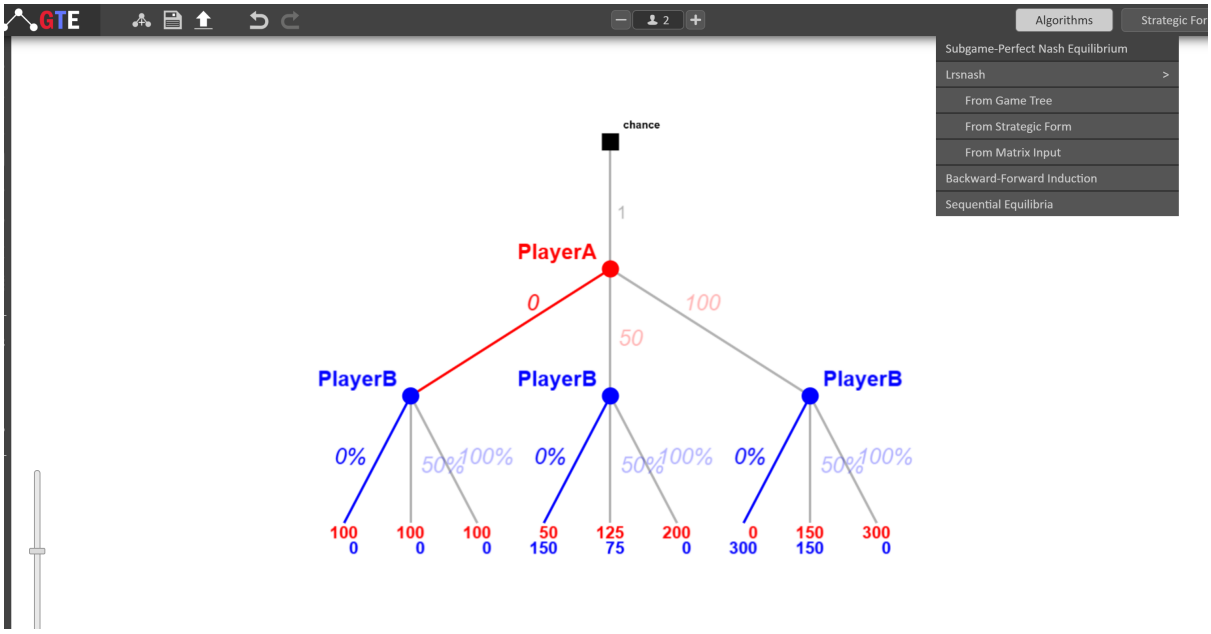

Figure 4: Solving the extensive-form version of Trust Simple with SPNE in GTE.

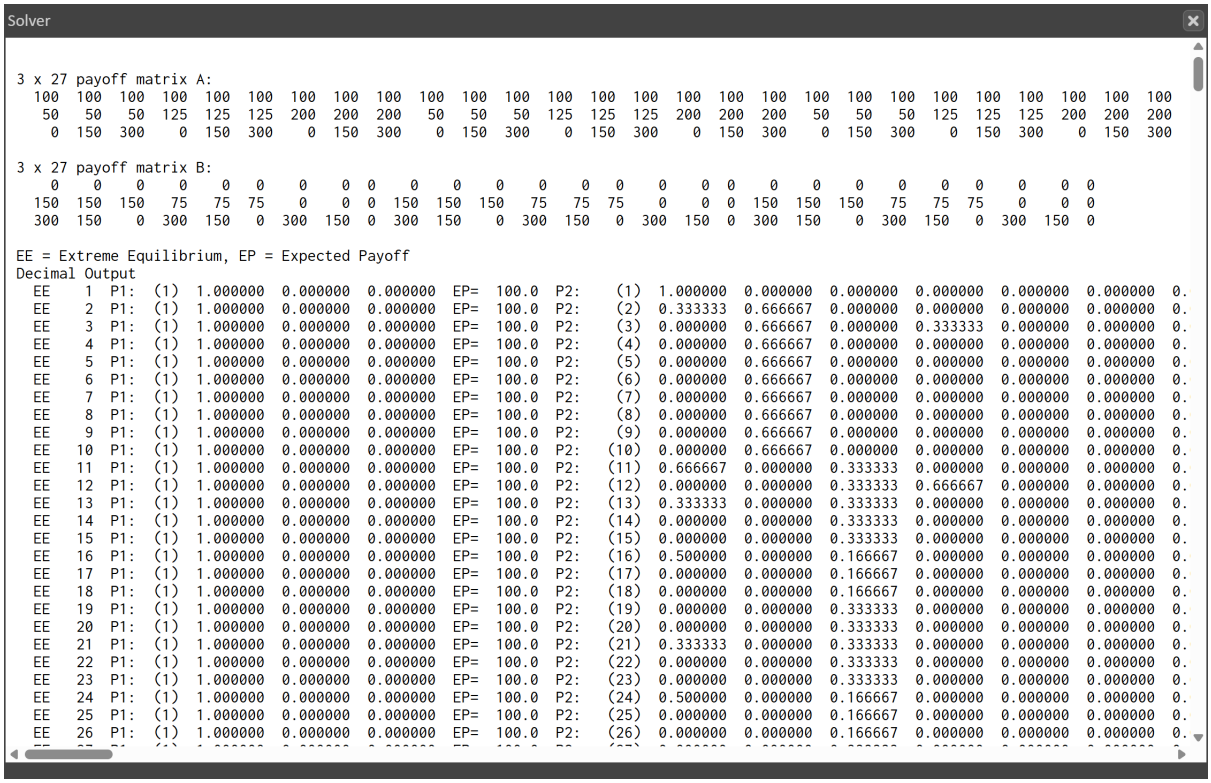

Figure 5: Solving the extensive-form version of Trust Simple with using payoff matrix in GTE, the solving process is a bit complicated.

SPNE and Its Relation to Part 1 and Normal Form

The Subgame Perfect Nash Equilibrium (SPNE) computed in GTE confirms the theoretical analysis presented in Part 1. In the one-round Trust Game, backward induction shows that Player B's optimal action in every subgame is to return 0, and anticipating this, Player A optimally invests 0. This outcome matches the SPNE identified in the extensive-form game.

When we translate the game into simultaneous normal form, as done with the NashPy payoff matrices, the same equilibrium emerges: the pure-strategy Nash Equilibrium is for Player A to invest 0 and Player B to return 0. In other words, the SPNE of the extensive-form game corresponds exactly to the NE of the simultaneous normal-form representation. This illustrates that, for a one-shot, perfect-information Trust Game, SPNE and pure-strategy NE are equivalent, and both capture the strategic incentives of the players consistently.

3 Part 3 Behavioral Scientist (experiment AI comparison)

3.1 3(a) oTree deployment

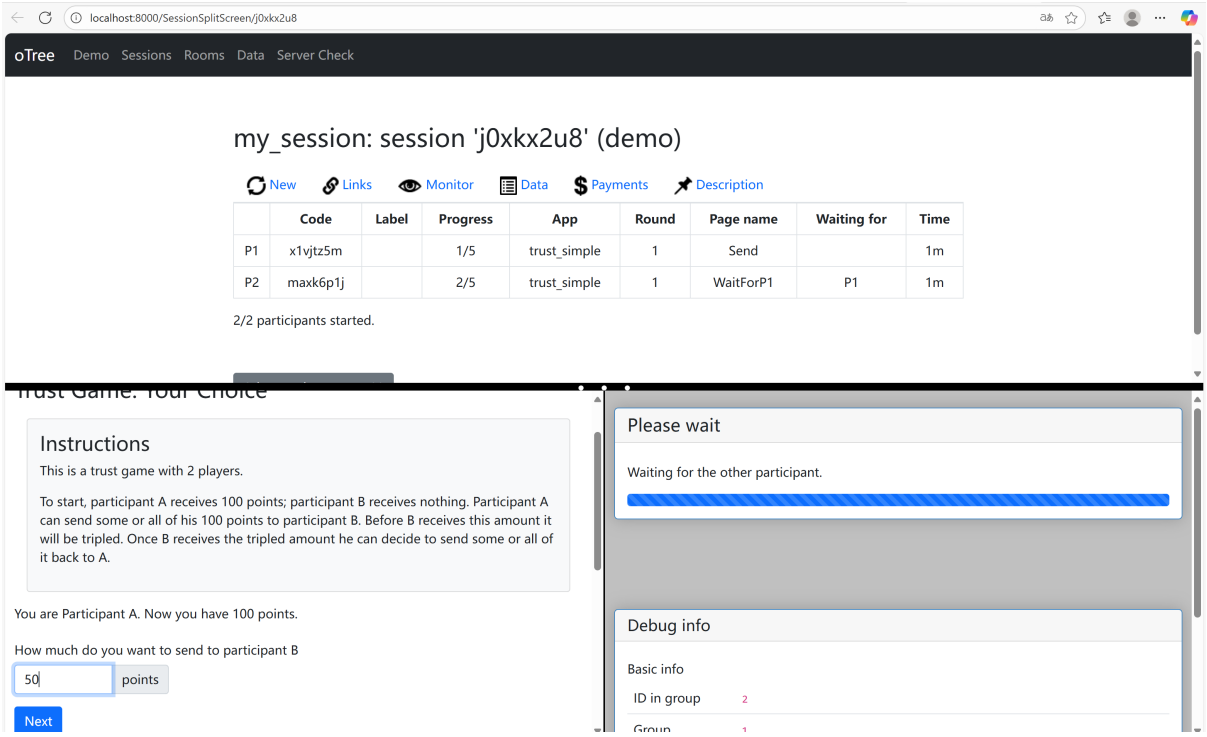

Figure 6: Gameplay: Trust Simple (using otree, step 1).

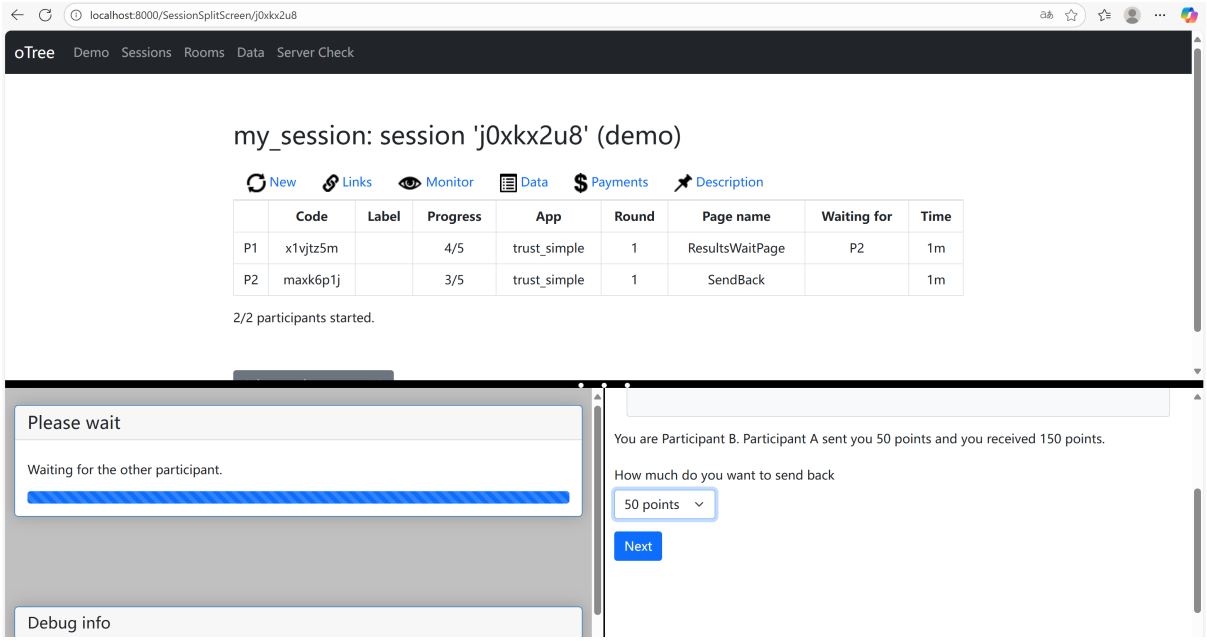

Figure 7: Gameplay: Trust Simple (using otree, step 2).

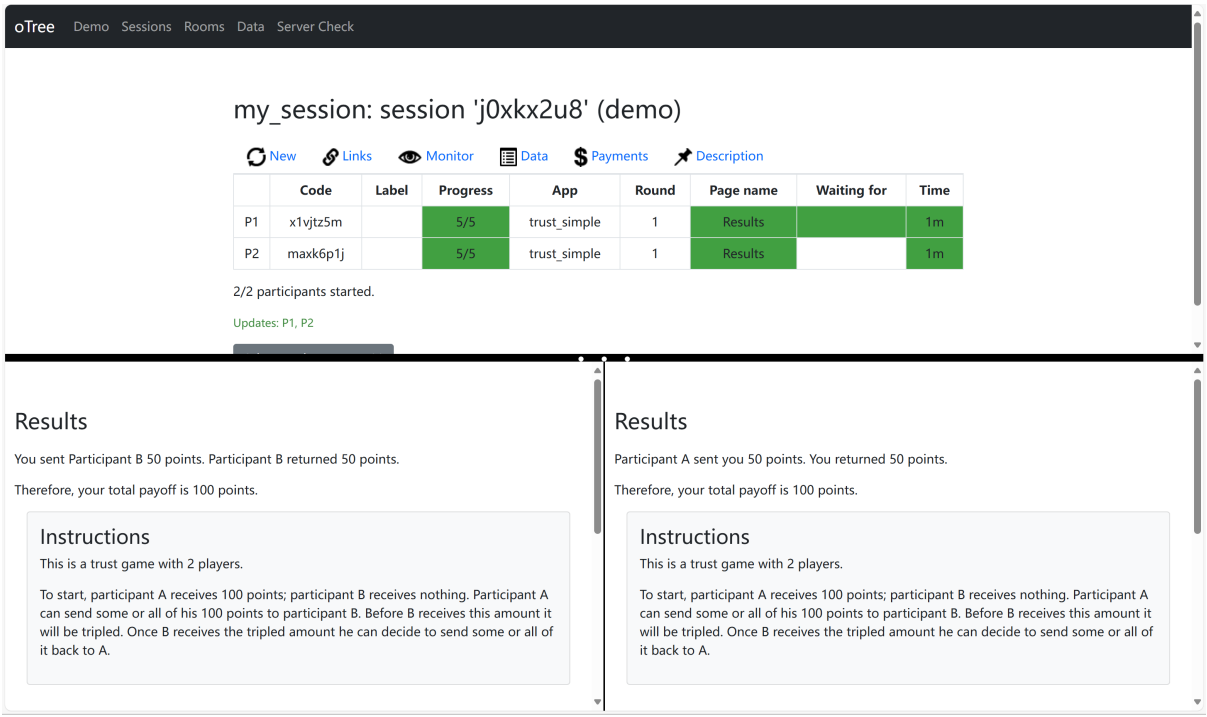

Figure 8: Gameplay: Trust Simple (using otree, step 3).

Post-play Interview Summary: Player A (Yihan) chose a moderate investment, expressing caution and some trust toward Player B. Player B (Ji Wu) returned one third the multiplied amount, citing some fairness and reciprocity considerations. Overall, participants' choices deviated from the SPNE prediction due to trust and social preference factors.

3.2 (3b) LLM "ChatBot" session

Trust Game Session Summary

Experiment Setup: One-round Trust Game, Multiplier = 3. LLM plays as either Player A or Player B depending on round.

Prompt Template for LLM (Player B):

```
You are Player B in a one-round Trust Game.
Player A invested X.
Multiplier is 3. Choose how much to return to Player A (0, 50%, or 100%) and explain your reasoning.
```

LLM Session Rounds:

The table below summarizes the five rounds of the Trust Game played between human and AI(ChatGPT) participants:

Round	Player A Investment	Player B Return	Player A Payoff	Player B Payoff
1	50	75	125	75
2	0	0	100	0
3	50 (AI as A)	0	50	150
4	100 (AI as A)	0	0	300
5	0 (AI as A)	0	100	0

Table 1: Summary of Trust Game rounds showing investments, returns, and payoffs.

Observations: Across the five rounds, the LLM's behavior showed both rational and fairness-oriented patterns. When acting as Player B, it sometimes returned part of the tripled investment (e.g., 50 percent in Round 1) to encourage trust, even though the subgame perfect prediction is always to return nothing. However, in later rounds it consistently converged to the payoff-maximizing choice of returning zero, especially when playing as Player A and anticipating Player B's behavior. This mixture suggests that the LLM balances economic rationality with social reasoning, unlike the strict backward-induction logic of SPNE. Compared to human participants, the LLM's decisions were more stable and transparent, but they also revealed sensitivity to framing: when fairness was emphasized in the prompt, cooperative actions appeared more likely.

3.3 3(c) Comparative analysis theory building

In theory, the Subgame Perfect Nash Equilibrium (SPNE) of the one-shot trust game is simple: Player A invests 0 and Player B returns 0. This is also the Nash equilibrium of the simultaneous normal form. However, the human session we conducted showed clear deviations. Some participants invested positive amounts, and some returns were observed, even though such actions reduce material payoff compared to the equilibrium. This behavior is consistent with findings in behavioral and experimental economics that players often care about fairness, reciprocity, or inequality, rather than only monetary outcomes. The LLM session displayed a mixture: in early rounds it sometimes returned a share of the investment with explicit fairness-based reasoning, but later rounds shifted toward the payoff-maximizing choice of returning 0. Compared with humans, the LLM was more stable and explained its reasoning more transparently, but it was also sensitive to prompt framing and payoff visibility.

A plausible mechanism behind these discrepancies is that both humans and LLMs effectively maximize a broader utility function than the one assumed in standard game theory. Humans put weight on social preferences and are boundedly rational, while LLMs are influenced by training priors, alignment instructions, and decoding randomness. To capture this, a potential refinement is what we might call a *Behavioral SPNE*. This approach keeps the extensive-form logic of SPNE, but replaces material payoffs with social-preference utilities and allows probabilistic (quantal) choice at each node. For example, we can model Player i's utility as

$$U_i = \pi_i + \alpha \cdot F_i,$$

where π_i is the monetary payoff, F_i captures fairness or reciprocity, and $\alpha \geq 0$ is the weight on social preferences. Choices are then selected with probability

$$P(a_i) = \frac{\exp(\lambda \cdot U_i(a_i))}{\sum_{a_i'} \exp(\lambda \cdot U_i(a_i'))},$$

where λ is a rationality parameter: $\lambda \to \infty$ recovers deterministic best response, while finite λ allows for bounded rationality. Standard SPNE is a special case with $\alpha = 0$ and $\lambda \to \infty$.

Such a refinement could better explain the cooperative tendencies of humans and the prompt-sensitive reasoning of LLMs, while still remaining consistent with equilibrium reasoning in subgames.

References

Berg, Joyce, John Dickhaut, and Kevin McCabe. 1995. "Trust, Reciprocity, and Social History." *Games and Economic Behavior* 10(1): 122–142.

Bommasani, Rishi, et al. 2021. *On the Opportunities and Risks of Foundation Models*. https://arxiv.org/abs/2108.07258.

Fehr, Ernst, and Klaus Schmidt. 1999. "A Theory of Fairness, Competition, and Cooperation." *Quarterly Journal of Economics* 114(3): 817–868.

Google Colab. 2020. *Colaboratory: Python in the Browser*. https://colab.research.google.com/.

Knight, Vincent. 2021. *Nashpy: A Python library for the computation of equilibria of 2-player strategic games*. https://nashpy.readthedocs.io/.

OpenAI. 2023. *GPT-4 Technical Report*. https://openai.com/research/gpt-4.

Rubinstein, Ariel, and Martin Osborne. 1994. *A Course in Game Theory*. Cambridge, MA: MIT Press.

Savani, Rahul, and Bernhard von Stengel. 2015. "Game Theory Explorer – Software for the Applied Game Theorist." *Computational Management Science* 12(1): 5–33. http://www.gametheoryexplorer.org/.

Shoham, Yoav, and Kevin Leyton-Brown. 2009. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge, UK: Cambridge University Press.

Acknowledgements

The author thanks OpenAI's ChatGPT (GPT-4) for providing guidance and conceptual suggestions during the design and analysis of the Trust Game experiments. All interpretations and conclusions remain the responsibility of the author.