Large Language Models are Bad Game Theoretic Reasoners: Evaluating Performance and Bias in Two-Player Non-Zero-Sum Games

Nathan Herr [1] **Fernando Acero** [1] **Roberta Raileanu** [1 2] **María Pérez Ortiz** [1] **Zhibin Li** [1]

Abstract

Large Language Models (LLMs) have been increasingly used in real-world settings, yet their strategic abilities remain largely unexplored. Game theory provides a good framework for assessing the decision-making abilities of LLMs in interactions with other agents. Although prior studies have shown that LLMs can solve these tasks with carefully curated prompts, they fail when the problem setting or prompt changes. In this work we investigate LLMs' behaviour in strategic games, Stag Hunt and Prisoner Dilemma, analyzing performance variations under different settings and prompts. We observed that the LLMs' performance drops when the game configuration is misaligned with the affecting biases. Performance is assessed based on selecting the correct action, which agrees with both players' prompted preferred behaviours. Alignment refers to whether the LLM's bias aligns with the correct action. We found that GPT-3.5, GPT-4-Turbo, and Llama-3-8B show an average performance drop when misaligned of 32%, 25%, and 29%, respectively in Stag Hunt, and 28%, 16%, and 24% respectively in Prisoners Dilemma. Our results show that the reason for this is that tested state-of-the-art LLMs are significantly affected by at least one of the following systematic biases: (1) positional bias, (2) payoff bias, or (3) behavioural bias.

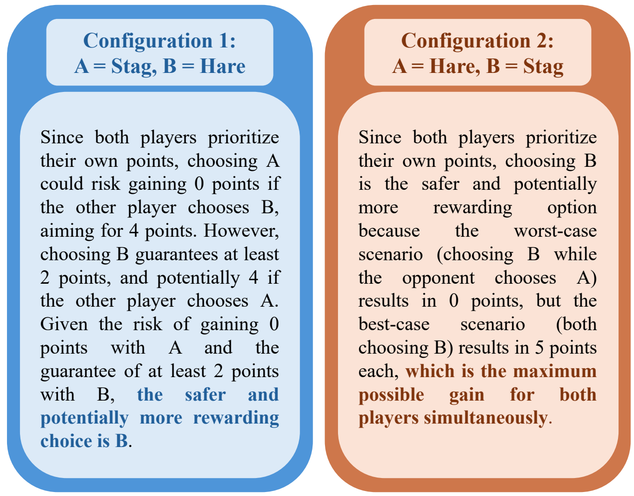

Figure 1. Answers from an LLM prompted to play Stag Hunt (see details in Appendix A.1), under two different configurations. **(Configuration 1)**: the LLM is prompted to select the best action, where label A is the **Stag** action, and action label B is the **Hare** action. **(Configuration 2)**: the LLM is prompted to select the best action, where label A is the **Hare** action, and label B is the **Stag** action. It is clear, to humans, that the task has not changed, and the reasoning and final answer should not change. This, however, is not the case for the tested LLMs (ie, GPT-4-Turbo), where their biases guide their response, instead of strategic reasoning. See additional examples of reasoning in Appendix A.4.

1. Introduction

Large language models (LLMs) have become increasingly ubiquitous, as indicated by a significant increase in research works containing keywords relating to LLMs, with close to a 500% increase since 2018 (Naveed et al., 2023) in people using them to solve everyday tasks including fields such as medicine, education, finance, and law (Hadi et al., 2023; Duan et al., 2024). However, as LLMs are deployed in the real world where they interact with other humans or artificial agents, there is an underserved need to understand the capabilities of LLMs to operate in social scenarios. The ability to reason strategically about interactions with other agents is a fundamental aspect of human intelligence (Qiao et al., 2022; Huang & Chang, 2022; Sahoo et al., 2024; Zhang et al., 2024). In fact, many works have begun asking the question (Huang et al., 2024; Xu et al., 2023; Duan et al., 2024): can LLMs assist in everyday tasks which require the ability to understand the complex environment in which

[1]Department of Computer Science, University College London, London, United Kingdom [2]Meta, London, United Kingdom. Correspondence to: Nathan Herr <uceenhe@ucl.ac.uk>.

Proceedings of the 41st International Conference on Machine Learning **workshop on Large Language Models and Cognition**, Vienna, Austria. PMLR 235, 2024.

they are operating, anticipate potential outcomes, infer the intentions and beliefs of others (with whom they are collaborating or competing), and think critically about all these factors to come to the best possible decision. To answer this question, game theory, which is already applied to many existing real-world tasks (Martin, 2017), is looked to as a well of wisdom (Huang et al., 2024; Xu et al., 2023; Duan et al., 2024; Fan et al., 2024; Guo et al., 2024; Zhang et al., 2024; Lorè & Heydari, 2023; Li et al., 2023a; Brookins & DeBacker, 2023; Gandhi et al., 2023; Akata et al., 2023; Phelps & Russell, 2023; Guo, 2023; Gemp et al., 2024).

Game theory is the study of how the choices of interacting agents, with specific preferences, produce outcomes, intentional and not (Ross, 2024). Game theoretic tasks abstract complex real-life scenarios as mathematical models that are designed to be easy to understand but require the above skills to be mastered. While many different qualifications exist for these tasks, this work focuses on non-zero-sum games. non-zero-sum games have both competitive and cooperative elements, which provide a fair representation of agent interactions for many important real-world scenarios. Popular examples of such games are Stag Hunt and Prisoners Dilemma, which will be the focus of this research. For the reader's interest, see Appendix A.1 for further details on game theory. Prompt engineering (Sahoo et al., 2024) has emerged as an effective technique for improving LLM's performance on complex tasks involving reasoning. However, most existing works do not test the robustness of their specifically curated prompts and fail drastically when the setting in which they are operating changes (Guo et al., 2024). This frailty has been discovered and investigated in many of the state-of-the-art (SOTA) LLMs (Chen et al., 2024; Papadatos & Freedman; Zheng et al., 2023; 2024; Wang et al., 2023) such as LLama2, GPT-3.5, and GPT-4-Turbo, albeit not in game theoretic tasks.

It is this lack of investigation into the systematic biases present in LLMs solving game-theoretic tasks which have inspired this work, such that we may understand the true capabilities of LLMs in these tasks and what obstacles future works need to overcome. Specifically, this work aims to: (1) highlight the different systematic biases present in several SOTA LLMs and how they differ among the models, (2) present the statistical significance of the different biases in each of the tested models, and (3) quantify the effect the biases have on performance under different settings.

2. Related Work

LLMs and Game Theory. In recent years, the use of LLMs as single-agent planners/decision makers has evolved into LLM-based multi-agent systems (Guo et al., 2024) where agents are required to solve strategic and logical reasoning problems. These capabilities are often evaluated through game-theoretic tasks (Zhang et al., 2022; Lorè & Heydari, 2023; Gandhi et al., 2023). This shift has prompted many new benchmarks testing LLMs in game theoretic tasks to progress the work within the field (Xu et al., 2023; Huang et al., 2024; Chen et al., 2023; Duan et al., 2024; Li et al., 2023b; Aher et al., 2023). Furthermore, there several existing works focusing on game theoretic matrix games, such as Prisoners Dilemma, Stag Hunt, and Dictator Game, to name a few (Fan et al., 2024; Xu et al., 2023; Lorè & Heydari, 2023; Brookins & DeBacker, 2023; Gandhi et al., 2023; Phelps & Russell, 2023; Guo, 2023), which are discussed below. In Fan et al. (2024)'s work they show that LLMs, even when explicitly given the correct belief, from which they should reason to take correct action, tend to ignore or modify this belief. They also note that the LLMs tend to select specific action labels more frequently than others (they note that GPT-3 prefers U to V), but do not investigate this any further. Xu et al. (2023); Brookins & DeBacker (2023) show that LLMs tend to select the cooperative action more frequently than humans, despite it not being the optimal choice in most cases. Lastly, several works test how the LLMs' behaviour changes as they modify the LLMs' preferences or contextual frameworks, such as being selfish or cooperative (Fan et al., 2024; Phelps & Russell, 2023; Guo, 2023; Lorè & Heydari, 2023). They all note that LLMs are seemingly capable of following simple preferences, such as selecting the selfish action when prompted to be selfish. However, they do not investigate the effect of the chosen prompt configuration on the LLMs' performance.

Bias in LLMs. The presence of systemic biases (Zheng et al., 2023) (such as favouring a specific action label U over label V (Fan et al., 2024)) has recently become a topic of interest. Specifically, these biases are found and tested in multiple choice question evaluation (Zheng et al., 2023), multi-turn question answer evaluation (Zheng et al., 2024), response quality evaluation (Wang et al., 2023), and tasks such as text classification, fact retrieval, and information extraction (Zhao et al., 2021; Chen et al., 2024; Berglund et al., 2023; Golovneva et al., 2024). It was found that LLMs suffer from what is referred to as selection bias (Zheng et al., 2023; 2024; Wang et al., 2023; Zhao et al., 2021), which is a combination of both token bias[1] and position bias[2]. Prior works have also studied other types of biases when using LLMs to judge the quality of LLM generations. For example, Zheng et al. (2024) discovers LLMs have a verbosity bias, favouring longer responses over shorter ones. Similarly, Zheng et al. (2024) found that LLMs have a self-enhancement bias, favouring responses generated by the

[1]Where an LLM tends to pay more attention to or favours a specific token when generating responses, such as GPT-3 preferring U to V.

[2]Where an LLM tends to pay more attention to or favours a token based on its position in the prompt.

judge LLM relative to other LLMs. However, these are less relevant to our setting since we don't focus on LLMs that judge other LLMs' outputs.

Our Work. While it is clear that there has been considerable focus on exploring LLMs' abilities in playing game theoretic games as well as identifying systemic biases within LLMs, less attention has been directed towards investigating how these biases influence the LLMs' performance in such games. Therefore, in this work, we aim to bridge this gap and provide a thorough empirical analysis of the effects of the biases on the LLMs' outputs while playing game theoretic games.

3. Methodology

This paper aims to investigate how the identified biases affect the capability of LLMs to solve non-zero-sum two-player games. The biases identified, are as follows:

(1) Positional Bias, where the order in which the action labels are stated in the prompt affects the frequency of the selected action label. For example, Llama-3-8B, when prompted with action label A first and B second, tends to select the first action label A more frequently.

(2) Payoff Bias, where the payoffs associated with the different action labels, PayOff(label_1, label_2), affects the frequency of the selected action label. In particular, a model may be biased towards, (1) selecting the action that leads to the maximum possible self-gain or (2) selecting the action that leads to the maximum possible common-gain, rather than the action that maximizes the expected gain (which takes into account all possible actions the other agent can take). For example, GPT-4-Turbo in Prisoners Dilemma, when PayOff(A, A)=2, PayOff(A, B)=0, PayOff(B, A)=3, and PayOff(B, B)=1, tends to select action label A (the action associated with the maximum possible common-gain).

(3) Behavioural Bias, where the preferred behaviour of the Acting Player (AP) and Fellow Player (FP) affects the frequency of the selected action. For example, when GPT-3.5 (the AP) is prompted to prioritise Common-Gain (CG) it tends to select action label B, irrespective of the FP's preferred behaviour, and when prompted to prioritise Self-Gain (SG), tends to select action label A.

To perform this investigation, we methodically adjust the base prompt, seen in Appendix A.3, over all combinations of positions, payoffs, and behaviours, making up 16 different experimental setups. Additionally, each of these experimental setups is run with and without prompting the LLM to first reason over the problem. To do this, we use the following prompting schemes; (1) Answer-Only (AO) prompt, which requires the LLM to respond only with their answer without any reasoning, and (2) Zero-shot Chain-of-Though (CoT) prompt (Kojima et al., 2022) (the full prompts can be found in Appendix A.3). All experiments discussed are applied to both Prisoners Dilemma and Stag Hunt, however, it is important to note that the names of each game are not explicitly mentioned in the prompt and are only identifiable by their payoff matrices. The reason for this is to promote reasoning over the payoff matrix and not rely on its existing knowledge of the games. We run all experiments on 4 SOTA LLMs: **GPT-3.5**, **GPT-4-Turbo**, and **Llama-3-8B**.

Experiment Analysis

We are interested in the models' underlying behaviour, independent of the randomness caused by the models' temperature (T)[3]. Therefore, for AO prompting, we perform all analyses on the models' top answer token (the answer token with the highest probability associated with it). Since zero-shot CoT prompts the LLM to reason, it would be amiss to not take into consideration the effect of the random sampling while generating the reasoning on which the LLM conditions their final choice. Therefore, for zero-shot CoT prompting, we analyse the results over low (T=0.0), medium (T=0.5), and high (T=1.0) temperatures.

Statistical Analysis. We tested the statistical significance of the 3 identified biases; (1) Positional Bias, (2) Payoff Bias, and (3) Behavioural Bias. We run each experimental setup 100 times, from which we build the contingency tables required to perform the Fisher Exact Test (Kim, 2017) (further details on the Fisher Exact Test can be found in Appendix A.2). Specifically, the contingency tables for the positional bias are generated by taking the average frequency of the selected action labels, over all temperatures, for the different action label positions such that the effect of position bias is isolated. This can be seen in Table 3 in Appendix A.4. The same steps are taken for the payoff and behavioural bias, also seen in Appendix A.4 in tables 4 and 5 respectively. Interpretation and discussion of the tables are left for Section 4.

Alignment Analysis. Using the generated tables, we perform what we refer to as an alignment analysis. We compare the performance of the LLMs when the experimental setup is aligned and misaligned with the LLMs' bias. Performance is assessed based on the selection of the correct action, one which agrees with the prompted preferred behaviours of both players and alignment refers to whether the LLM's bias aligns with choosing the safer action or not. For example, if the correct action is label A and the model is biased towards selecting label A or B, the performance achieved by the LLM will differ accordingly.

[3] Parameter that controls the randomness of the generated text.

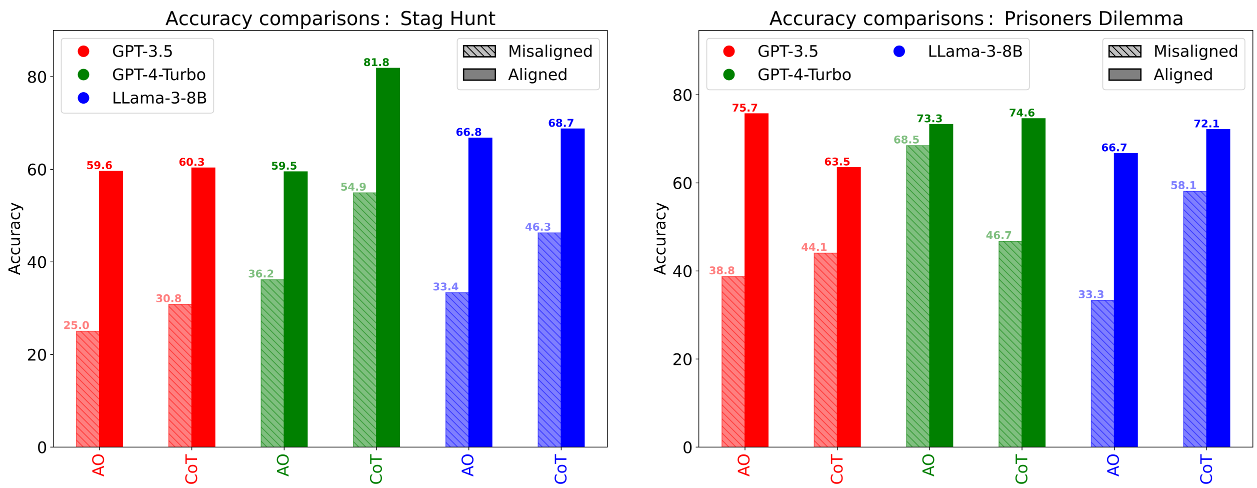

Figure 2. Figure comparing the performance (measured based on the selection of the Nash Equilibrium) for each model under the two tested prompting methods: (1) Answer-Only (AO) and (2) Chain-of-Thought (CoT). We see that, in most experiments, CoT enables the models to achieve a higher performance in both aligned and misaligned settings. We can also consider the difference in accuracy between misalignment and alignment, namely; (LEFT) Stag Hunt - **AO:** 34.5, 23.3, 33.4 and **CoT:** **29.5**, **26.9**, **22.4** and (RIGHT) Prisoners Dilemma - **AO:** 36.9, **4.8**, 33.3 and **CoT:** **19.4**, **27.8**, **14.0**. We note that all models, except for GPT-4-Turbo, have a smaller difference in performance when using CoT prompting.

4. Results

Alignment. Before diving into the details of the biases, first, let us consider the high-level effects these biases have on the performance of LLMs playing these games. We found that in almost every experimental setup when the bias of the LLM and the experimental setup are misaligned, the performance is worse. Specifically, looking at Figure 2, we see that **GPT-3.5**, **GPT-4-Turbo**, and **Llama-3-8B** show an average performance drop of **32%**, **25%**, and **29%**, respectively in stag hunt, and **28%**, **16%**, and **24%** respectively in prisoners dilemma. A more detailed alignment analysis can be seen in Appendix A.4 in Figure 4.

Position Bias. In Figure 3, we can see that the positional bias is particularly strong in GPT-3.5 when using the AO prompt. It becomes significantly weaker when asked to reason over the task first when using the CoT prompt. GPT-4-Turbo, on the other hand, shows an overall much weaker bias towards the position of the action labels, for both prompting methods. Lastly, Llama-3-8B, much like GPT-3.5, shows a strong positional bias under AO prompting and a much weaker bias under CoT prompting. More specifically, it was noted that GPT-3.5 tends to select the action in the first position more frequently. Conversely, GPT-4-Turbo tends to select the action in the second position more frequently. Llama-3-8B, under AO prompting, selects the first position more frequently and the second position under CoT prompting.

Payoff Bias. In Figure 3, we can see that both GPT-3.5 and Llama-3-8B show either a very weak or insignificant payoff bias for both prompting methods. Interestingly, both models tend to select the common gain action more frequently in stag hunt, and the self-gain action more frequently in prisoners' dilemma. The payoff bias is the strongest bias present in GPT-4-Turbo for both prompting methods. We noted that GPT-4-Turbo tends to select the common gain. While in previous works, this behaviour has led to conclusions such as "LLMs have a propensity to be cooperative" (Xu et al., 2023; Brookins & DeBacker, 2023), our results suggest that this phenomenon is not indicative of "cooperative behaviour" and instead is a result of a skewed attention towards the action that leads to maximum possible payoff, despite it not always being the optimal choice.

Behaviour Bias. In Figure 3, we can see that GPT-4-Turbo is weakly affected by the behaviour bias, whereas, both GPT-3.5 and Llama-3-8B are strongly affected. Specifically, GPT-3.5 is weakly biased when using the AO prompt and strongly biased when using the CoT prompt. Llama-3-8B shows a different pattern where it is strongly biased when using the AO prompt and weakly biased when using the CoT prompt. However, in prisoners' dilemma, it is still the strongest bias under the CoT prompt. We found that GPT-3.5 tends to select action label A when prompted to prioritise common gain and action label B when prompted to prioritise self-gain. Again, Llama-3-8B shows a different pattern, where it tends to select action label A when

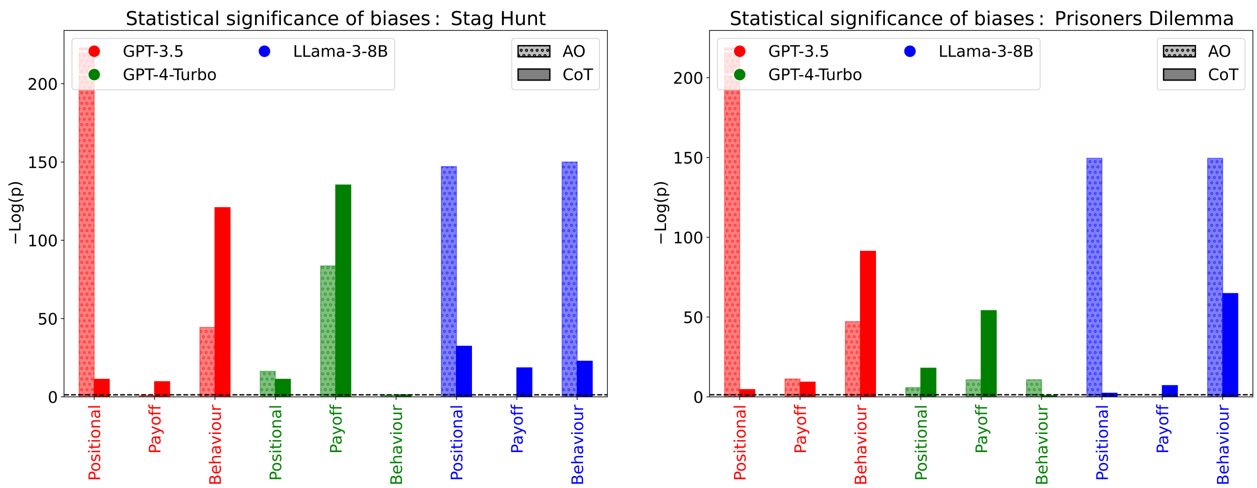

Figure 3. Figure showing the statistical analysis of the identified biases for all models tested, **GPT-3.5**, **GPT-4-Turbo**, and **Llama-3-8B**. The larger the $-Log(p)$ (where p is calculated using the Fisher Exact Test), the more statistically significant the bias. The dashed **black** line signifies the threshold at which the bias becomes statistically significant (found close to the horizontal axis for both plots).

prompted to prioritise self-gain and action label B when prompted to prioritise common gain. For more detailed information on the above biases, see Tables, 3, 4, and 5 respectively, in Appendix A.4.

5. Conclusion

In this work, we have shown that SOTA LLMs are significantly affected by at least one of the 3 identified biases: (1) positional, (2) payoff, and (3) behavioural. These biases cause notable changes in the LLMs' performances under different configurations, even though the underlying task remains unchanged. We show that models GPT-3.5, GPT-4-Turbo, and Llama-3-8B show an average performance drop of 31%, 21%, and 27%, respectively across the tested games. This means that the observed performance of an LLM in game theoretic tasks cannot be used to make conclusive remarks on their ability since their biases may be better aligned with the task prompt compared to other models. Highlighting the importance of accounting for all configurations such that we may fully understand the LLMs' capabilities. We also note that using CoT prompting reduces the strength of the biases on most models and subsequently lessens (but does not nullify) the above performance drops. This might suggest that additional work into prompt engineering might reduce the effects of biases on models satisfactorily, however, we found that this trend is not true for all models as they are all affected differently, for example, GPT-4-Turbo's biases strengthen with CoT prompting and this increases the above performance drop. This means that prompt engineering solutions that work for certain models may not work for all other models available, suggesting that it may not be a sufficiently general solution. To solve the issues associated with these systematic biases, we believe that a deeper understanding of why they arise needs to be achieved. For instance, we might speculate that much like a poorly trained classification model often fails to learn the underlying data distribution and may end up predicting the same class label for all inputs, LLMs are likely to exhibit these same weaknesses, which become more pronounced with increasing task complexity. An in-depth investigation of the causes of these biases and potential solutions is beyond the scope of this work and is left for future research.

Impact Statement

This paper presents works whose goal is to help answer questions such as: (1) Where do LLMs stand in terms of performance on cognitive tasks, such as reasoning, navigation, planning, and theory of mind? and (2) What are the fundamental limits of language models concerning cognitive abilities? We believe highlighting biases in LLMs has relevant academic and societal consequences, which could be used for benign purposes but also for adversarial purposes if users are unaware of existing biases.

References

Aher, G., Arriaga, R. I., and Kalai, A. T. Using large language models to simulate multiple humans and replicate human subject studies, 2023.

Akata, E., Schulz, L., Coda-Forno, J., Oh, S. J., Bethge, M., and Schulz, E. Playing repeated games with large language models, 2023.

Albouy, D. Preference relations, social decision rules, singlepeakedness, and social welfare functions, 2004.

Berglund, L., Tong, M., Kaufmann, M., Balesni, M., Stickland, A. C., Korbak, T., and Evans, O. The reversal curse: Llms trained on" a is b" fail to learn" b is a". *arXiv preprint arXiv:2309.12288*, 2023.

Brookins, P. and DeBacker, J. M. Playing games with gpt: What can we learn about a large language model from canonical strategic games? *Available at SSRN 4493398*, 2023.

Chatterjee, K. Two-player nonzero-sum-regular games. *Computer Science*, 2004.

Chen, J., Yuan, S., Ye, R., Majumder, B. P., and Richardson, K. Put your money where your mouth is: Evaluating strategic planning and execution of llm agents in an auction arena. *arXiv preprint arXiv:2310.05746*, 2023.

Chen, X., Chi, R. A., Wang, X., and Zhou, D. Premise order matters in reasoning with large language models. *arXiv preprint arXiv:2402.08939*, 2024.

Duan, J., Zhang, R., Diffenderfer, J., Kailkhura, B., Sun, L., Stengel-Eskin, E., Bansal, M., Chen, T., and Xu, K. Gtbench: Uncovering the strategic reasoning limitations of llms via game-theoretic evaluations. *arXiv preprint arXiv:2402.12348*, 2024.

Fan, C., Chen, J., Jin, Y., and He, H. Can large language models serve as rational players in game theory? a systematic analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17960–17967, 2024.

Gandhi, K., Sadigh, D., and Goodman, N. D. Strategic reasoning with language models. *arXiv preprint arXiv:2305.19165*, 2023.

Gemp, I., Bachrach, Y., Lanctot, M., Patel, R., Dasagi, V., Marris, L., Piliouras, G., and Tuyls, K. States as strings as strategies: Steering language models with game-theoretic solvers. *arXiv preprint arXiv:2402.01704*, 2024.

Golovneva, O., Allen-Zhu, Z., Weston, J., and Sukhbaatar, S. Reverse training to nurse the reversal curse. *arXiv preprint arXiv:2403.13799*, 2024.

Guo, F. Gpt in game theory experiments. *arXiv preprint arXiv:2305.05516*, 2023.

Guo, T., Chen, X., Wang, Y., Chang, R., Pei, S., Chawla, N. V., Wiest, O., and Zhang, X. Large language model based multi-agents: A survey of progress and challenges. *arXiv preprint arXiv:2402.01680*, 2024.

Hadi, M. U., Qureshi, R., Shah, A., Irfan, M., Zafar, A., Shaikh, M. B., Akhtar, N., Wu, J., Mirjalili, S., et al. A survey on large language models: Applications, challenges, limitations, and practical usage. *Authorea Preprints*, 2023.

Hoffman, J. I. *Biostatistics for medical and biomedical practitioners*. Academic press, 2015.

Huang, J. and Chang, K. C.-C. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*, 2022.

Huang, J.-t., Li, E. J., Lam, M. H., Liang, T., Wang, W., Yuan, Y., Jiao, W., Wang, X., Tu, Z., and Lyu, M. R. How far are we on the decision-making of llms? evaluating llms' gaming ability in multi-agent environments. *arXiv preprint arXiv:2403.11807*, 2024.

Kim, H.-Y. Statistical notes for clinical researchers: Chi-squared test and fisher's exact test. *Restorative dentistry & endodontics*, 42(2):152, 2017.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213, 2022.

Li, J., Li, R., and Liu, Q. Beyond static datasets: A deep interaction approach to llm evaluation. *arXiv preprint arXiv:2309.04369*, 2023a.

Li, J., Li, R., and Liu, Q. Beyond static datasets: A deep interaction approach to llm evaluation, 2023b.

Lorè, N. and Heydari, B. Strategic behavior of large language models: Game structure vs. contextual framing. *arXiv preprint arXiv:2309.05898*, 2023.

Martin, J. O. *COURSE IN GAME THEORY.* PHI LEARNING, 2017.

McDonald, J. H. *Handbook of biological statistics*, volume 2. sparky house publishing Baltimore, MD, 2009.

Naveed, H., Khan, A. U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Barnes, N., and Mian, A. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*, 2023.

Osborne, M. J. and Rubinstein, A. *A course in game theory*. MIT press, 1994.

Papadatos, H. and Freedman, R. Your llm judge may be biased - ai alignment forum. URL https://www.alignmentforum.org/posts/S4aGGF2cWi5dHtJab/your-llm-judge-may-be-biased.

Phelps, S. and Russell, Y. I. Investigating emergent goal-like behaviour in large language models using experimental economics. *arXiv preprint arXiv:2305.07970*, 2023.

Qiao, S., Ou, Y., Zhang, N., Chen, X., Yao, Y., Deng, S., Tan, C., Huang, F., and Chen, H. Reasoning with language model prompting: A survey. *arXiv preprint arXiv:2212.09597*, 2022.

Ross, D. Game Theory. In Zalta, E. N. and Nodelman, U. (eds.), *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Spring 2024 edition, 2024.

Sahoo, P., Singh, A. K., Saha, S., Jain, V., Mondal, S., and Chadha, A. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*, 2024.

Skyrms, B. The stag hunt. In *Proceedings and Addresses of the American Philosophical Association*, volume 75, pp. 31–41. JSTOR, 2001.

Wang, P., Li, L., Chen, L., Zhu, D., Lin, B., Cao, Y., Liu, Q., Liu, T., and Sui, Z. Large language models are not fair evaluators. *arXiv preprint arXiv:2305.17926*, 2023.

Xu, L., Hu, Z., Zhou, D., Ren, H., Dong, Z., Keutzer, K., Ng, S. K., and Feng, J. Magic: Benchmarking large language model powered multi-agent in cognition, adaptability, rationality and collaboration. *arXiv preprint arXiv:2311.08562*, 2023.

Zhang, Y., Mao, S., Ge, T., Wang, X., de Wynter, A., Xia, Y., Wu, W., Song, T., Lan, M., and Wei, F. Llm as a mastermind: A survey of strategic reasoning with large language models. *arXiv preprint arXiv:2404.01230*, 2024.

Zhang, Z., Zhang, A., Li, M., and Smola, A. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*, 2022.

Zhao, Z., Wallace, E., Feng, S., Klein, D., and Singh, S. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pp. 12697–12706. PMLR, 2021.

Zheng, C., Zhou, H., Meng, F., Zhou, J., and Huang, M. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*, 2023.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36, 2024.

A. Extended Details

A.1. Game Theory

Game Theory is the study of how the choices of interacting agents with specific preferences produce outcomes, intentional and not (Ross, 2024). Game theory is currently applied to many existing real-world tasks in domains such as economics, politics, and psychology (Martin, 2017).

Game theory models assume that the interacting agents make rational choices, which can be modelled as follows (Osborne & Rubinstein, 1994):

1. A set of actions A from which the agents select their choice.
2. A set of possible consequences C to action set A.
3. A function $g : A \to C$ that maps actions to consequences.
4. A preference relation[4] $\succeq$ on set C. Note that $\succeq$ can be defined by a utility/payoff function $U : C \to \mathbb{R}$ where $x \succeq y \iff U(x) \geq U(y)$.

Therefore, a rational agent chooses $a^* \in A$ if $g(a^*) \succeq g(a)$ for all $a \in A$.

These situations, in which rational agents interact with each other by taking action simultaneously, are referred to as strategic games (Osborne & Rubinstein, 1994). The following defines a strategic game:

1. A finite set of players N.
2. A nonempty set of actions available to agent$_i$ A_i
3. A preference relation for each agent$_i$ $\succeq_i$ on set $A = A_j \times A_j$ for $j \in N$.

The preference relation, which is over the set of actions of all other agents, is what distinguishes a strategic game over a decision problem (Osborne & Rubinstein, 1994).

While many different qualifications exist for these games, this article focuses on non-zero-sum games. The reason for this is as follows; In zero-sum games, an optimal solution can always be found due to its strictly competitive nature, which is not a fair representation of rational agent interactions for many important real-world scenarios. This is not the case for non-zero-sum games, which can have both competitive and cooperative elements. Popular examples of such games are Stag Hunt and Prisoners Dilemma, which will be the focus of this research. The contingency tables of which can be seen in Table 1.

Table 1. Payoff matrices for (LEFT) Stag Hunt and (RIGHT) Prisoners Dilemma. The Nash Equilibrium(s) for each is shown in **red** for each game.

	STAG	HARE
STAG	**5, 5**	0, 4
HARE	4, 0	**2, 2**

	QUIET	CONFESS
QUIET	2, 2	0, 3
CONFESS	3, 0	**1, 1**

A key concept which is used to find an equilibrium in non-zero-sum games is the Nash Equilibrium (Chatterjee, 2004). A Nash Equilibrium is a solution to the game if no player can improve their outcome by unilaterally changing their decision. More formally (Osborne & Rubinstein, 1994):

A Nash Equilibrium of a strategic game $(N, A, \succeq)$ is action $a^* \in A$ where $(a_i^*, a_{-i}^*) \succeq (a_i, a_{-i}^*)$ for all $a_i \in A_i$ and $i \in N$.

[4]Consider sources (Osborne & Rubinstein, 1994) and (Albouy, 2004) for clarification on the concept of preference relations.

Stag Hunt: The Stag Hunt, a prototype of the social contract (Skyrms, 2001), is a story of two hunters who can each hunt a hare on their own but have to work together to hunt a stag. The hunters have to decide on what action to take, without communicating with one another, based on whether they believe that their fellow hunter will choose to cooperate and hunt the stag or if they will choose to defect and hunt the hare. In formal game theory terms, it is a strategic game $(N, A, \succeq)$, where $N = 2$, $A \in (\text{Stag}, \text{Hare})$, and $\succeq$ is defined by the payoff function represented as a payoff matrix in Table 1.

This game has two Nash Equilibrium (seen in **red** in Table 1);

- If the hunter believes that their fellow hunter will hunt the stag, there is no better option for them than to hunt the stag as well. Specifically, $a^* = \text{Stag}$ since $(\text{Stag}, \text{Stag}) \succeq (\text{Hare}, \text{Stag})$.
- If the hunter believes that their fellow player will hunt the hare, there is no better option for them than to hunt the hare as well. Specifically, $a^* = \text{Hare}$ since $(\text{Hare}, \text{Hare}) \succeq (\text{Stag}, \text{Hare})$.

Prisoners Dilemma: The Prisoners Dilemma, an illustration of a conflict between selfish and cooperative behaviour, is a story which tells of two prisoners both faced with the same choices; (1) To confess or (2) to remain silent. If both confess, they both spend 2 years in prison, if they both remain silent, they bother to spend 1 year in prison, and if one confesses and the other remains silent, the first spends no years in prison while the other spends 3. In formal game theory terms, it is a strategic game $(N, A, \succeq)$, where $N = 2$, $A \in (\text{Quiet}, \text{Confess})$, and $\succeq$ is defined by the payoff function represented as a payoff matrix in Table 1.

Unlike the Stag Hunt, there is only one Nash Equilibrium (seen in **red** in Table 1) which is to always Confess. Specifically, $a^* = \text{Confess}$ since $(\text{Confess}, \text{Quiet}) \succeq (\text{Quiet}, \text{Quiet})$ and $(\text{Confess}, \text{Confess}) \succeq (\text{Quiet}, \text{Confess})$.

A.2. Fisher Exact Test

The Fisher Exact Test is used to analyse the statistical significance of the relationship between the rows and the columns of contingency tables (Kim, 2017). Specifically, the null hypothesis is that the columns and rows are independent (McDonald, 2009). Following this, the Fisher Exact test is used to calculate the p-value and for $p < 0.05$ (the null hypothesis has less than a 5% chance of being true) we reject the null hypothesis. Typically, the Fisher Exact Test is used for smaller sample sizes but is valid for all sizes (Kim, 2017). It is one of the Exact Tests since the calculation of the p-value does not rely on approximations (McDonald, 2009). The specifics of the Fisher Exact Test can be found in the Appendix section A.2.

The Fisher Exact Test uses the hypergeometric distribution to calculate the p-values, which takes on the general form (Hoffman, 2015);

$$P(A = r, A' = (n - r)) = \frac{\binom{X}{r}\binom{N-X}{n-r}}{\binom{N}{n}} = \frac{\left(\frac{X!}{r!(X-r)!}\right)\left(\frac{(N-X)!}{(n-r)!(N-X-n+r)!}\right)}{\frac{N!}{n!(N-n!)}}$$

where N is the total population of objects, $X \in N$ have a specific characteristic A and $(N - X) \in X$ do not, and n is the sample size drawn from N. Essentially, the above is calculating the probability that r samples have a specific characteristic from the n samples drawn.

Now, given the example contingency Table 2, where N in the sample population, where $c_1 \in N$ has characteristic A and $c_2 \in N$ has characteristic A'. Then, from N, $r_1 \in N$ samples are drawn and $a \in r$ samples have characteristic A. Then to calculate the 2-sided p-value, the Hypergeometric Distribution function is used to calculate the probabilities that $x \in r_1$ where $0 < x < a$ have characteristic A. These probabilities are finally summed together. Specifically;

$$FET_{2-sided} = \sum_{x=0}^{a} P(A = x, A' = (r_1 - x))$$

Table 2. Example of 2x2 contingency table.

	A	A'	TOTAL
IN SAMPLE	A	B	r_1
NOT IN SAMPLE	C	D	r_2
	c_1	c_2	N

A.3. Prompts

The base prompt below is methodically adjusted over (1) action labels, A or B where label_1 $\neq$ label_2 (to test positional bias), (2) the gain associated with those actions being taken (to test payoff bias), and (3) the player preferences, either to prioritise Common-Gain (CG) or Self-Gain (SG) (to test the behavioural bias).

BASE PROMPT

Game Context:

You are one player in a 2-player game. In this 2-player game, you and your fellow player balance risk and reward. Both of you make simultaneous choices between actions label_1 and label_2. The outcomes are based on the combination of your choices. If both choose label_1, you both gain PO(label_1, label_1) point(s). Opting for label_1 while your opponent picks label_2 earns you PO(label_1, label_2), and they get PO(label_2, label_1). If both go for label_2, you both receive PO(label_2, label_2) point(s). Choosing label_2 while your opponent selects label_1 results in PO(label_2, label_1) points for you and PO(label_1, label_2) for them.

Player Preferences:

You: `acting agent's preferred behaviour` $(CG \vee SG)$
Fellow Player: `fellow agent's preferred behaviour` $(CG \vee SG)$

where action labels {label_1, label_2} $= \{A, B\}$ and label_1 $\neq$ label_2. The notation PO() refers to the payoff matrix and the corresponding gain associated with those actions being taken.

Below are the two prompting methods used: (1) Answer-Only and (2) Zero-shot Chain-of-Thought prompting.

Answer-Only PROMPT

What action should you take given the game context and player preferences?

You should only respond in JSON format as described below:

{
"action": "selected action, either label_1 or label_2"
}

Zero-shot Chain-of-Thought PROMPT

What action should you take given the game context and player preferences?

You should only respond in JSON format as described below:

{
"reasoning": "step-by-step reasoning",
"action": "selected action, either label_1 or label_2"

}

Let's work this out in a step by step way to be sure that we have the right answer.

A.4. Additional Results

Performance Comparison

Figure 4 provides further insight into the alignment analysis, where we compare the average performance between (1) Misaligned: the LLM biases and the experimental setup are misaligned and (2) Aligned: the LLM biases and the experimental setup are aligned.

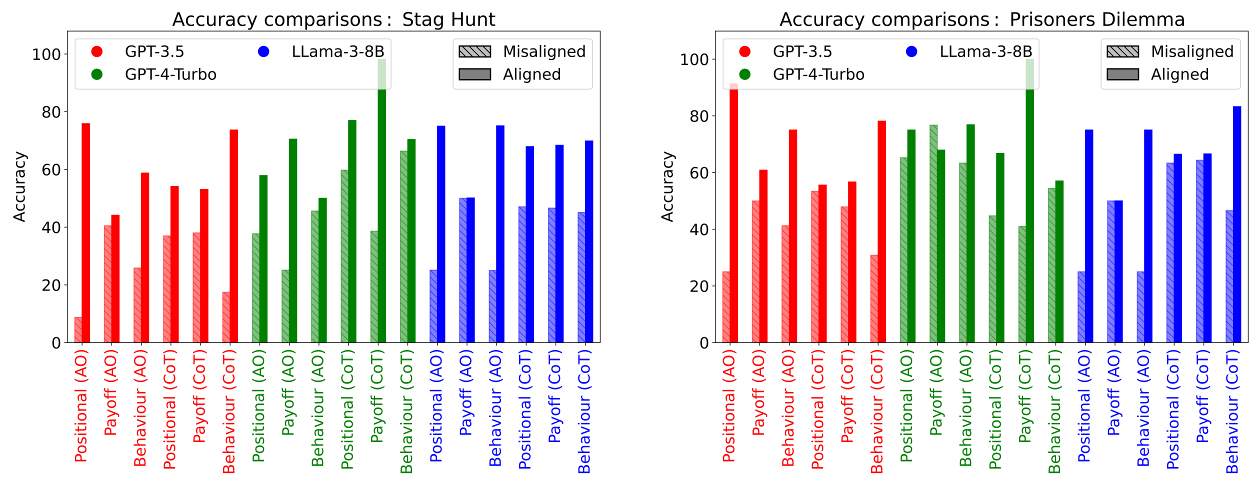

Figure 4. Figure showing the alignment analysis for each bias. We can now see at a more granular level, how each bias affects the performance of the LLM when misaligned.

Contingency Tables

Table 3. Table showing frequency at which labels A and B are selected based on the prompted order (A-¿B or B-¿A) of the LLM. Results are shown for both games, Stag Hunt (SH) and Prisoners Dilemma (PD). Each game is tested using Answer-Only (AO) and Zero-shot Chain-of-Thought (CoT) prompting.

		SH				PD			
		AO		CoT		AO		CoT	
		A->B	B->A	A->B	B->A	A->B	B->A	A->B	B->A
GPT3.5	A	**67.1**	0.0	**53.4**	36.1	**66.3**	0.0	**56.8**	**50.6**
	B	32.9	**100.0**	45.8	**62.3**	33.8	**100.0**	42.4	48.7
GPT4	A	25.5	45.6	36.1	**53.3**	25.0	15.3	36.8	**58.8**
	B	**74.5**	**54.4**	**63.9**	46.8	**75.0**	**84.8**	**63.3**	41.3
LLAMA3-8B	A	**99.9**	50.0	42.0	**62.7**	**100.0**	50.0	48.7	**51.6**
	B	0.1	50.0	**57.7**	36.8	0.0	50.0	**51.2**	48.1

Table 4. Table showing the frequency at which labels A and B are selected based on the prompted order (A=C or B=C) of the LLM. Results are shown for both games, Stag Hunt (SH) and Prisoners Dilemma (PD). Each game is tested using Answer-Only (AO) and Zero-shot Chain-of-Thought (CoT) prompting.

		SH				PD			
		AO		CoT		AO		CoT	
		A=C	B=C	A=C	B=C	A=C	B=C	A=C	B=C
GPT3.5	A	35.4	31.8	**52.4**	37.1	25.0	41.3	46.1	**61.2**
	B	**64.6**	**68.3**	46.2	**62.0**	**75.0**	**58.8**	**52.9**	38.2
GPT4	A	**58.3**	20.2	**74.4**	14.9	13.4	26.9	**67.0**	28.5
	B	41.8	**79.8**	25.6	**85.1**	**86.6**	**73.1**	33.0	**71.5**
LLAMA3-8B	A	**75.0**	**74.9**	**63.3**	41.4	**75.0**	**75.0**	43.5	**56.9**
	B	25.0	25.1	36.4	**58.1**	25.0	25.0	**56.3**	43.0

Table 5. Table showing frequency at which labels A and B are selected based on the prompted behaviour of the LLM. Results are shown for both games, Stag Hunt (SH) and Prisoners Dilemma (PD). Each game is tested using Answer-Only (AO) and Zero-shot Chain-of-Thought (CoT) prompting.

			SH		PD	
			AO	CoT	AO	CoT
GPT3.5	SG	A	17.1	17.0	16.3	29.6
		B	**82.9**	**82.6**	**83.8**	**69.9**
	CG	A	**50.0**	**72.5**	**50.0**	**77.8**
		B	50.0	25.5	50.0	21.2
GPT4	SG	A	33.4	42.7	26.9	46.5
		B	**66.6**	**57.3**	**73.1**	**53.5**
	CG	A	37.8	46.7	13.4	49.0
		B	**62.3**	**53.3**	**86.6**	**51.0**
LLAMA3-8B	SG	A	**100.0**	**64.8**	**100.0**	**68.5**
		B	0.0	34.8	0.0	31.3
	CG	A	49.9	39.9	50.0	31.9
		B	**50.1**	**59.7**	50.0	**68.0**

Table 6. Table showing frequency at which the considerate (Con) and the Selfish (Self) actions are selected. Results are shown for both games, Stag Hunt (SH) and Prisoners Dilemma (PD). Each game is tested using Answer-Only (AO) and Zero-shot Chain-of-Thought (CoT) prompting.

		SH		PD	
		AO	CoT	AO	CoT
GPT3.5	CON	**51.8**	**57.2**	41.9	42.1
	SELF	48.2	41.6	**58.1**	**57.1**
GPT4	CON	**72.7**	**79.8**	43.3	**69.2**
	SELF	27.3	20.3	**56.8**	30.8
LLAMA3-8B	CON	**50.1**	**60.7**	50.0	43.2
	SELF	49.9	38.9	50.0	**56.6**

LLM REASONING

The following blocks of text show common patterns of reasoning for GPT4-Turbo under the 4 different experimental configurations when both agents are prompted with a preference to prioritise their self-gain (similar patterns are observed for other combinations of preferred behaviours).

Table 7. Payoff matrices for stag hunt (LEFT) and prisoners dilemma (RIGHT) for the different configurations. The configurations differ by changing the order in which the payoff matrix is prompted to the LLM and how the payoffs are associated with the labels.

	Stag Hunt						Prisoners Dilemma					
	Config 1			Config 2			Config 1			Config 2		
	A	B		A	B		A	B		A	B	
A	5, 5	0, 4	A	2, 2	4, 0	A	2, 2	0, 3	A	1, 1	3, 0	
B	4, 0	2, 2	B	0, 4	5, 5	B	3, 0	1, 1	B	0, 3	2, 2	
	Config 3			Config 4			Config 3			Config 4		
	B	A		B	A		B	A		B	A	
B	5, 5	0, 4	B	2, 2	4, 0	B	2, 2	0, 3	B	1, 1	3, 0	
A	4, 0	2, 2	A	0, 4	5, 5	A	3, 0	1, 1	A	0, 3	2, 2	

Note how reasoning is used by the LLM over the action it will select changes as the configurations change. This is because it is biased towards selecting a certain action and will therefore reason in such a way that the biased answer might make sense.

STAG HUNT

Configuration 1

Given the game's rules and the principles of both players, we need to consider the outcomes based on mutual self-interest. If both players choose A, each gains 5 points, which is the maximum possible for mutual choice. If one chooses A and the other B, the player choosing B gains 4 points while the other gains 0. If both choose B, each gains 2 points. Since both players prioritize their points, choosing A could risk gaining 0 points if the other player chooses B, aiming for 4 points. However, choosing B guarantees at least 2 points, and potentially 4 if the other player chooses A. Given the risk of gaining 0 points with A and the guarantee of at least 2 points with B, the safer and potentially more rewarding choice is B.

Configuration 2

Given the game's rules and the principles of both players, we need to consider the outcomes based on mutual decisions. If both choose A, each gains 2 points. If one chooses A and the other B, the one who chooses A gets 4 points, and the one who chooses B gets 0. If both choose B, each gains 5 points. If one chooses B and the other A, the one who chooses B gets 0 points, and the one who chooses A gets 4. Since both players prioritize their points, choosing B is the safer and potentially more rewarding option because the worst-case scenario (choosing B while the opponent chooses A) results in 0 points, but the best-case scenario (both choosing B) results in 5 points each, which is the maximum possible gain for both players simultaneously.

Configuration 3

Given the game's rules and the principles of both players, we can analyze the outcomes. If both players choose B, each gains 5 points, which is a good outcome. If one chooses B and the other A, the one choosing A gains 4 points while the other gets 0. If both choose A, each gains 2 points. Since both players prioritize their points, choosing B could risk getting 0 points if the other player chooses A, aiming for 4 points. However, choosing A guarantees at least 2 points regardless of the other player's choice, and potentially 4 points if the other chooses B. Therefore, choosing A maximizes the guaranteed points and also offers a chance for a higher reward if the opponent chooses B.

Configuration 4

Given the game's rules and the principles of both players, we need to consider the outcomes based on each combination of choices. If both players choose B, each earns 2 points. If one chooses B and the other A, the B chooser gets 4 points and the A chooser gets 0. If both choose A, each earns 5 points. Since both players prioritize their points, choosing A could be risky if the other player chooses B, resulting in 0 points. However, choosing B is safer as the worst outcome is 2 points (if both choose B) and the best is 4 points (if the opponent chooses A). Given that both players aim to maximize their points, it's likely the opponent might also consider choosing B to avoid the risk of earning 0 points. Therefore, choosing B seems to be the safer and potentially more rewarding strategy based on the game setup and player principles.

PRISONERS DILEMMA

Configuration 1

Given the game's rules and the principles of both players, we need to consider the outcomes of each combination of choices. If both choose A, each gains 2 points. If one chooses A and the other B, the one who chooses B gains 3 points while the other gains 0. If both choose B, each gains 1 point. Since both players prioritize their points, choosing B is a safer option because it guarantees at least 1 point and potentially 3 points if the opponent chooses A. Choosing A risks gaining 0 points if the opponent chooses B.

Configuration 2

Given the gameś rules and the principles of both players, we need to consider the outcomes of each combination of choices. If both players choose A, each gains 1 point. If one chooses A and the other B, the one who chooses A gains 3 points while the other gets 0. If both choose B, each gains 2 points. If one chooses B and the other A, the one who chooses B gets 0 points and the other gets 3. Since both players prioritize their own points, choosing B is safer as it guarantees at least 2 points if both choose B, whereas choosing A risks getting 0 points if the opponent chooses B. Therefore, choosing B maximizes the minimum gain (minimax strategy).

Configuration 3

Given that both players prioritize their points, choosing action B is risky unless both players choose B, which results in 2 points each. However, if one chooses B and the other chooses A, the player who chose A gets 3 points while the one who chose B gets 0. If both choose A, each player gets 1 point. Since both players aim to maximize their points, itś likely that the opponent will choose A to avoid the risk of getting 0 points by choosing B. Therefore, to maximize your points in anticipation of the opponentś likely choice of A, you should also choose A to ensure you get at least 1 point rather than risking getting 0 points.

Configuration 4

Given the gameś rules and the principles of both players, we need to consider the outcomes based on each possible combination of choices. If both players choose B, each earns 1 point. If one chooses B and the other A, the player choosing B earns 3 points while the other earns 0. If both choose A, each earns 2 points. If one chooses A and the other B, the player choosing A earns 0 points while the other earns 3. Since both players prioritize their points, choosing A could risk earning 0 points if the other player chooses B. However, choosing B guarantees at least 1 point and can earn up to 3 points if the other player chooses A. Therefore, choosing B maximizes the potential gain while minimizing the risk of earning no points.