How Well Do LLMs Adapt Their Play? A Controlled Evaluation with Classic, Misère, and Novel Board Games

Anonymous authors
Paper under double-blind review

Abstract

Board games provide a controlled setting for testing whether large language models adapt their behavior when task objectives or rules change. We study two forms of adaptation within a General Game Playing (GGP) framework. First, classic–*misère* pairs invert the success criterion while preserving the board, actions, legality rules, and state transitions. Second, Centralis, an original game designed for this study, introduces a new spatial structure, action semantics, and victory condition. All actions are validated by a common GDL-based referee. We evaluate Qwen3-8B and Qwen3-14B in Instruct and Reasoning configurations against four calibrated GGP reference opponents. The Reasoning configurations show fewer model-generated forfeits and greater recovery from invalid responses, but differences in competitive performance depend on the game and opponent and require substantially more tokens and execution time. The classic–*misère* comparisons reveal an objective-adaptation gap: win rates decrease or remain low in several pairings after the success criterion is inverted, despite unchanged game mechanics. In contrast, Qwen3-14B Reasoning achieves high win rates against all four reference opponents in Centralis, while Qwen3-8B Reasoning generally completes matches but records few wins. These results distinguish adaptation to an inverted objective from play under a new rule system and show that operational robustness, competitive performance, and inference cost characterize different aspects of model behavior.

1 Introduction

The evaluation of language models in games has moved beyond benchmarks centered on imitation or surface accuracy toward settings that examine procedural reasoning and sequential decision-making in greater detail (Ruoss et al., 2024; Srivastava et al., 2023). At the same time, many evaluations still rely on a small set of widely known domains and aggregate measures such as win/loss or correct/incorrect outcomes. Under these conditions, it is difficult to separate adaptation to rules and objectives from familiarity with patterns acquired during pre-training. A model may produce legally valid actions without following a policy that is aligned with the active game objective. Recent work has documented this gap between formal validity and strategic behavior in game settings, including failures to exploit immediately available winning opportunities (Lin et al., 2025) and broader limitations in planning, memory, and sequential control (Ying et al., 2026).

A related question is whether models adjust their behavior when the objective changes without a substantial change in the underlying game dynamics. Recent work on procedural reasoning shows that training on rule-governed systems can improve performance in these settings, while also documenting failures when evaluation conditions depart from established conventions (Bateni et al., 2025). In games, this question can be examined in a controlled way through domain pairs that preserve most of their structure while inverting the victory condition (Morillo et al., 2025). If a model continues to behave as if it were playing the standard variant after the objective has changed, then performance in familiar domains provides only a partial view of how the model responds to the task at hand.

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

We examine how language models adjust their action selection when a game's objective changes or when they are given the rules of an original game. We use adaptation in this behavioral sense, rather than to refer to learning over successive matches. Our main contributions are as follows:

(i) we propose an evaluation design with two classic–*misère* pairs that preserve the board, action space, legality rules, and transitions while inverting the success criterion, complemented by Centralis, an original game designed for this study with its own mechanics and victory condition;

(ii) we define a GGP-based protocol for evaluating heterogeneous agents under shared game semantics, with separate analyses of competitive performance, operational robustness, and inference cost; and

(iii) we evaluate Qwen3-8B and Qwen3-14B in Instruct and Reasoning configurations against four calibrated GGP reference opponents, comparing behavioral adaptation, operational robustness, and competitive performance by model size, inference configuration, game family, and opponent.

2 Related Work

Games have become a common setting for studying reasoning in language models beyond surface-level accuracy (Topsakal et al., 2025; van Wetten et al., 2025). AI GameStore (Ying et al., 2026) evaluates frontier models in a broad collection of games and reports low performance in tasks involving planning, memory, and world-model learning. GTBENCH (Duan et al., 2024), GAMEBoT (Lin et al., 2025), and TTT-Bench (Mishra et al., 2025) examine strategic behavior more directly and show that LLMs often fall short of classical search methods, including in small deterministic games. Other work also shows that performance varies with game structure and payoff specification (Akata et al., 2025), and surveys describe a gap between natural-language competence and formal strategic reasoning (Sun et al., 2025).

A related line of work studies whether models adjust to changes in rules or objectives Gandhi et al. (2023). Fine-tuning on alternative rule systems can improve procedural reasoning (Bateni et al., 2025), while other studies report failures under unconventional specifications or dynamic-rule environments (Malmqvist, 2025; Gevers & Daelemans, 2025; van Wetten et al., 2025). These results motivate evaluation settings in which the objective can vary while much of the domain structure remains fixed.

Work on evaluation methodology has also emphasized that aggregate performance alone may be insufficient. Benchmark^2 (Qian et al., 2026) highlights discriminability and consistency as properties of useful evaluations, while (Collins et al., 2025) shows that overall performance measures can diverge from more fine-grained strategic judgments. This paper is closer to that line than to game-specific performance benchmarks, but differs in its use of a controlled set of related game domains. By varying the objective while preserving the underlying game structure, the evaluation examines behavioral changes associated with objective inversion without attributing them to a specific underlying mechanism. A shared GGP referee applies each game's legality rules, transitions, and terminal conditions consistently to all agents.

3 Games

TicTacToe is played on a 3×3 board in which players alternate marking empty cells using a `mark` action with row and column coordinates. Suicide keeps the same board and action format, but inverts the victory condition so that completing a line results in a loss for the player who forms it. Connect4 is played on a board of 6 rows and 8 columns using `drop` actions, where each piece falls to the lowest available position in the selected column. NotConnect4 preserves this mechanic while treating a four-piece connection as a losing condition. The GDL descriptions for these four games were taken from the Stanford Gamemaster environment (Stanford Logic Group, 2026), an online platform for General Game Playing competition, execution, and development. Centralis, by contrast, was implemented from scratch in GDL for this study. This places all domains under the same logical formalization and allows them to be executed and validated by the same referee.

Centralis is the only original game in the evaluation. Its board has a radial layout with 16 sections, 3 peripheral rings, and a blocked central cell. Players, `x` and `o`, begin by marking peripheral cells

through `mark(section, ring)` actions. The central cell cannot be occupied directly. To unlock it, a player must form a path with three of their own marks, one in each ring, aligned in one of three ways: radially, by keeping the same section between rings; clockwise diagonally, by shifting the section consecutively from one ring to the next; or counter-clockwise diagonally, following the opposite pattern. Unlocking the center does not immediately end the game. The player who unlocks it must occupy the center on their next turn to win; otherwise, no victory is awarded and play continues. If the peripheral cells are filled and the center remains uncaptured, the game ends in a draw. Examples of the board representations and move formats are provided in Appendix A.1.

3.1 WHY MISÈRE VARIANTS?

The *misère* variants modify the victory condition while preserving the traditional game structure. This makes it possible to examine whether an agent adjusts to the objective stated in the rules or continues to behave as if it were playing the standard version. In TicTacToe and Connect4, inverting the objective does not introduce a new action space, a different board, or a different transition system. What changes is the condition under which a position becomes favorable or terminal. This design supports comparisons of closely related domain pairs without introducing a different environment altogether. If an agent follows a policy that remains suited to the standard variant after the objective has been inverted, that mismatch can be observed without changing the action interface or the core mechanics of the game. For this reason, the *misère* variants provide a controlled way of examining sensitivity to the active objective.

3.2 WHY A NOVEL GAME?

Centralis is an original game designed for this study to provide a setting with no established gameplay history and to reduce the likelihood that model behavior is driven by prior exposure during training. Its radial board departs from the rectangular layouts used in the classic games, although each peripheral action is still represented compactly as a tuple identifying a section and a ring. The game also introduces a two-stage victory condition: a player must first form one of the allowed three-mark paths to unlock the center and must then occupy the center on the following turn to win. Unlocking the center alone is therefore insufficient, allowing us to distinguish recognition of a goal-enabling state from execution of the action required to complete the objective. Whereas the classic–*misère* pairs test adaptation to an inverted objective while preserving the underlying game structure, Centralis examines how models apply a new rule system with an unfamiliar spatial organization, action semantics, and terminal condition. All games are executed and validated through the same GDL-based framework.

4 PLAYERS

The evaluation includes two types of players: Large Language Models and General Game Playing agents. The LLM players comprise Qwen3-8B and Qwen3-14B, each evaluated in `instruct` and `reasoning` configurations. These models interact with the games through the textual interface described in Section 5.1. The GGP players, by contrast, operate directly on the formal game specification and serve as reference opponents.

The evaluation includes four GGP reference opponents: `legal`, `random`, `greedy`, and `mcs`, taken from the Gamemaster environment (Stanford Logic Group, 2026). Their purpose is not to provide compute- or information-matched baselines for the language models, nor to establish whether GGP agents outperform LLMs. Instead, all LLM conditions are evaluated against the same set of opponents, providing common reference points with different strategic behaviors and empirical levels of difficulty.

Although the two player types receive different forms of access to the game, both are governed by the same referee, game rules, state transitions, terminal conditions, and legality criteria. This keeps the game semantics fixed within each domain while allowing different LLM configurations to be evaluated against the same opponents.

4.1 LLM PLAYERS

For the Qwen3 models, the `instruct` and `reasoning` configurations use the same model size and differ in their inference setup, allowing a within-model comparison between the two generation modes. The `instruct` condition uses a maximum generation budget of 128 tokens with `reasoning_effort` set to `none`, whereas the `reasoning` condition uses a maximum generation budget of 1,024 tokens with `reasoning_effort` set to `low`. For both conditions, decoding parameters were fixed to reduce sampling variability: temperature was set to 0, `top_p`=1, `presence_penalty`=0, and `frequency_penalty`=0. Further implementation and model-specific configuration details are provided in the reproducibility appendix A.

4.2 GGP REFERENCE OPPONENTS

The `legal` agent selects a legal action from the current state without an additional strategic criterion, while `random` samples uniformly from the available legal actions. The `greedy` agent expands a state tree within the available time and selects actions according to estimated utility. The `mcs` agent uses Monte Carlo Search, evaluating successor states through simulations and selecting the action with the highest estimated average utility within the time limit. Because `greedy` and `mcs` have access to search procedures that are not available to the LLM agents, their results are interpreted as reference-opponent outcomes rather than as matched comparisons between agent classes. Each GGP agent is initialized with its assigned role, the game specification, and the corresponding start and play clocks. For stochastic agents, the seed is derived from the match identifier and differs by pairing. This provides a fixed randomization condition for each match while avoiding identical stochastic trajectories across repeated games. Exact reproducibility nevertheless depends on the pseudorandom-number handling of the underlying execution environment.

5 EXPERIMENTAL PROTOCOL

5.1 TEXT-BASED INTERACTION

Interaction with the language models is entirely text-based. At the beginning of each match, the model receives a system prompt describing the game rules, its assigned role, and examples of the expected action format. At each turn, the model receives the current board state in text form, a reminder of its assigned role (e.g., `x` or `o`), and a request for the next move. The logical state representation is not exposed to the model, and the full interaction history is not accumulated as additional context. This keeps the interaction structure fixed across games and inference conditions.

5.2 MOVE PARSING AND LEGALITY CHECKING

Each model response is processed in two stages. First, the text output is parsed into a candidate move according to the expected format of the game. Second, the candidate move is checked against the legal action set of the current state using a General Game Playing referee. Legality is therefore determined by the referee rather than by manual inspection or template matching. If a response cannot be parsed, it is recorded as a formatting error, and the model is queried again. If the parsed move is not legal in the current state, it is recorded as an illegal move, and a new response is requested. To prevent long error sequences, the match ends in a forfeit if, after the first invalid response, the model accumulates ten illegal or incorrectly formatted outputs. This procedure is applied consistently across all games, model sizes, and inference configurations.

5.3 MATCH SETUP, DATA LOGGING, AND CALIBRATION

The main evaluation matches each LLM condition against the four GGP reference opponents in all five games. The LLM conditions are Qwen3-8B and Qwen3-14B, each evaluated in `instruct` and `reasoning` configurations. The GGP reference opponents are `legal`, `random`, `greedy`, and `mcs`. For each combination of LLM condition, opponent, and game, 60 matches are included in the analysis, with player order balanced so that the LLM moves first in 30 matches and second in the remaining 30.

After execution, matches affected by infrastructure failures, such as API or backend errors, are identified and excluded from the analysis. These executions are not counted as model-generated invalid responses or forfeits and do not contribute to the target number of matches. The corresponding condition and player order are rerun until 30 completed matches unaffected by infrastructure failures are obtained for each order. The final analyzed dataset therefore contains 1,200 matches per LLM condition, 960 matches per game, and 4,800 LLM-versus-GGP matches in total.

Each executed match produces two JSONL files. A `matches` file stores match-level metadata, including the match identifier, game, players, model condition, and final outcome, following the naming pattern `matches_[ID]_[game]_[player1]_[player2].jsonl`. A `moves` file stores the turn-level sequence, following the naming pattern `moves_[ID]_[game]_[player1]_[player2].jsonl`. A consolidated CSV file is generated from the retained matches for aggregate analysis. Match-level records are used to summarize outcomes by game, opponent, and model condition, while move-level records support the analysis of invalid responses, retries, and interaction duration.

In addition to the LLM evaluation, we run a calibration phase among the four GGP reference opponents. For each game, 100 matches are played for each ordered pairing of GGP agents, including self-play, yielding 1,600 matches per game. These experiments characterize the behavior of the reference opponents within each domain and provide an empirical indication of their relative difficulty. They are used to contextualize the LLM results rather than to establish a matched comparison between LLM and GGP agent classes.

5.4 Metrics

The reported metrics fall into three groups. The first measures competitive performance through wins, losses, draws, and model-generated forfeits. The second assesses operational robustness through the mean number of invalid attempts per move, the proportion of moves requiring at least one retry, and the frequency of matches ending after the retry budget is exhausted.

Invalid responses include outputs that cannot be parsed into the expected move format, parsed actions that are illegal in the current state, and empty outputs associated with generation truncation. A truncation-related empty response is identified when the backend returns an empty `raw_response` while the reported completion-token count reaches the configured generation limit, indicating that generation terminated at the output budget before a usable move was returned. Such responses consume an attempt in the retry procedure and may lead to a model-generated forfeit if the retry budget is exhausted. Recovery is measured as the proportion of moves that, after at least one invalid response, produce a valid action before exhausting the retry budget.

The third group characterizes inference cost using generated tokens and end-to-end match time. Metrics are reported by game, game family, GGP opponent, and model condition, allowing competitive performance, operational robustness, and inference cost to be examined separately rather than combined into a single aggregate score. For Qwen3-8B and Qwen3-14B, the design supports within-model comparisons between `instruct` and `reasoning` configurations in terms of valid action generation, recovery from invalid responses, competitive performance against the different reference opponents, and inference cost. Infrastructure failures are excluded following the procedure described in Section 5.3 and therefore do not contribute to any of the reported metrics. Comprehensive details on the experimental setup, prompting strategy, model configurations, and reproducibility are provided in Appendix A. The anonymized code, game specifications, prompts, and scripts used to reproduce the experiments are available at `https://anonymous.4open.science/r/llm_vs_ggp-orchestrator-05C5/README.md`.

6 Results

6.1 Calibration of the game set with GGP agents

The GGP round-robin calibration shows consistent differences among reference opponents, although their relative performance varies by game (Table 1). In TicTacToe, Suicide, Connect4, and NotConnect4, `Greedy` and `MCS` generally record higher win rates than `Legal` and `Random`. Their ordering is not fixed, however: `MCS` outperforms `Greedy` in Connect4 and NotConnect4, whereas

in Centralis `Greedy` wins 0.86 of their direct matches. This supports the use of the four agents as reference opponents with game-dependent competitive profiles rather than as a fixed difficulty hierarchy. Centralis yields 0.47 wins for `x`, 0.44 for `o`, and 0.10 draws, with a mean match length of 23.7 turns and a median of 20. Additional calibration results are reported in Appendix A.

Table 1: GGP cross-play win rate matrix: each cell shows the win rate of the row agent against the column opponent on the indicated game (200 matches per pairing). Opponent abbreviations: L = Legal, R = Random, G = Greedy, M = MCS.

	TicTacToe				**Suicide**				**Connect4**				**NotConnect4**				**Centralis**			
Agent	L	R	G	M	L	R	G	M	L	R	G	M	L	R	G	M	L	R	G	M
Legal	–	0.57	0.01	0.00	–	0.37	0.05	0.00	–	0.85	0.00	0.03	–	0.14	0.02	0.00	–	0.00	0.00	0.00
Random	0.39	–	0.00	0.01	0.60	–	0.07	0.01	0.14	–	0.06	0.01	0.85	–	0.10	0.00	0.76	–	0.01	0.03
Greedy	0.99	0.88	–	0.20	0.94	0.78	–	0.49	1.00	0.94	–	0.17	0.97	0.90	–	0.04	1.00	0.99	–	0.86
MCS	1.00	0.97	0.00	–	1.00	0.94	0.24	–	0.97	0.99	0.82	–	1.00	1.00	0.95	–	0.99	0.97	0.14	–

Figure 1 reports the win rates of Qwen3-8B and Qwen3-14B in *instruct* and *reasoning* configurations against the four GGP reference opponents. Table 2 provides the corresponding distributions of wins, draws, losses, and model-generated forfeits. Each game–opponent condition contains 60 analyzed matches, balanced by player order.

6.2 Match Outcomes by Game Family

In the classic games, the *reasoning* configurations generally record higher win rates than their *instruct* counterparts, particularly against `legal` and `random`. In TicTacToe, this pattern appears for both model sizes. In Connect4, *reasoning* also records higher win rates against these two opponents, although the differences between Qwen3-8B and Qwen3-14B vary by pairing. Against `greedy` and `mcs`, several conditions remain at low win rates under both inference configurations.

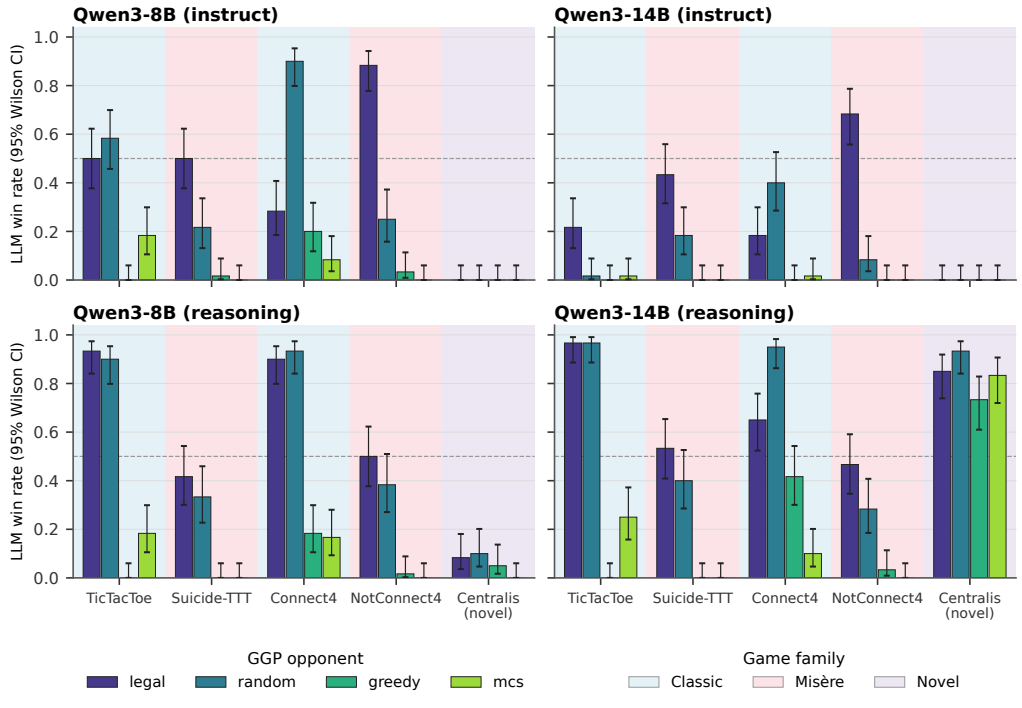

Figure 1: Win rates of Qwen3-8B and Qwen3-14B in *instruct* and *reasoning* configurations against the four GGP reference opponents across the five games. Error bars show 95% Wilson confidence intervals; background shading identifies classic, *misère*, and novel game families.

For the *reasoning* configurations, the classic–*misère* comparisons show lower or unchanged win rates when moving from TicTacToe to Suicide and from Connect4 to NotConnect4 for each GGP opponent. The differences are more evident against `legal` and `random`, while win rates against `greedy` and `mcs` are already low in several classic-game conditions and remain close to zero in the corresponding *misère* variants. The *instruct* configurations show a less consistent pattern: some pairings decrease after objective inversion, whereas others, including several comparisons against `legal`, increase.

Centralis shows a marked difference between the two *reasoning* conditions. Qwen3-8B *reasoning* records few wins and none against `mcs`, whereas Qwen3-14B *reasoning* reaches win rates between 0.73 and 0.93 against the four reference opponents. Neither *instruct* condition records wins in Centralis, and their outcome distributions contain a high proportion of forfeits. Qwen3-14B *reasoning* therefore shows a different outcome profile in Centralis from that observed in the two *misère* games, particularly against `greedy` and `mcs`. The match outcomes vary by model size, inference configuration, game, and opponent. The *reasoning* configurations record higher win rates in several classic-game conditions and differ substantially between Qwen3-8B and Qwen3-14B in Centralis, while the *misère* variants show lower win rates in multiple pairings.

Table 2: Outcome distribution per game × GGP opponent for Qwen3-8B and Qwen3-14B, in both Instruct and Reasoning variants. Each cell reports the number of matches (out of 60) ending in win (W), draw (D), loss (L), or LLM forfeit (F).

		Instruct																				**Reasoning**											
Model	Opponent	TicTacToe				Suicide				Connect4				NotConnect4				Centralis				TicTacToe				Suicide				Connect4			
		W	D	L	F	W	D	L	F	W	D	L	F	W	D	L	F	W	D	L	F	W	D	L	F	W	D	L	F	W	D	L	F
Qwen3-8B	Legal	30	0	30	0	30	0	30	0	17	0	43	0	53	0	7	0	0	0	0	60	56	0	4	0	25	11	24	0	54	0	6	0
	Random	35	5	20	0	13	8	39	0	54	0	6	0	15	0	45	0	0	0	0	60	54	5	1	0	20	12	28	0	56	0	4	0
	Greedy	0	13	47	0	1	5	54	0	12	0	48	0	2	0	58	0	0	0	15	45	0	48	12	0	0	10	50	0	11	0	49	0
	MCS	11	1	47	1	0	0	60	0	5	0	55	0	0	0	60	0	0	0	4	56	11	38	11	0	0	2	58	0	10	0	50	0
Qwen3-14B	Legal	13	6	6	35	26	0	20	14	11	0	31	18	41	0	8	11	0	1	0	59	58	0	2	0	32	3	25	0	39	0	21	0
	Random	1	2	2	55	11	0	13	36	24	0	5	31	5	0	23	32	0	0	1	59	58	2	0	0	24	16	20	0	57	0	3	0
	Greedy	0	0	8	52	0	1	21	38	0	0	16	44	0	0	25	35	0	0	1	59	0	43	17	0	0	2	58	0	25	0	35	0
	MCS	1	1	8	50	0	2	27	31	1	0	25	34	0	0	28	32	0	0	0	60	15	36	9	0	0	2	58	0	6	0	54	0

		Reasoning															
Model	Opponent	NotConnect4				Centralis											
		W	D	L	F	W	D	L	F								
Qwen3-8B	Legal	0	0	0	60	5	55	0	0								
	Random	0	0	0	60	6	9	45	0								
	Greedy	0	0	15	45	3	0	57	0								
	MCS	0	0	4	56	0	0	60	0								
Qwen3-14B	Legal	0	1	0	59	51	9	0	0								
	Random	0	0	1	59	56	1	3	0								
	Greedy	0	0	1	59	44	0	16	0								
	MCS	0	0	0	60	50	0	10	0								

6.3 Operational Robustness and Invalid Responses

The four Qwen3 conditions exhibit distinct operational profiles. For Qwen3-8B, invalid responses under *instruct* are almost entirely associated with illegal action selection, whereas the *reasoning* configuration produces substantially fewer retries in TicTacToe, Suicide, Connect4, and NotConnect4. Centralis remains different from the paired games: Qwen3-8B *instruct* requires retries in more than 70% of turns against every opponent and frequently exhausts the retry budget, while *reasoning* still requires retries but completes the matches without model-generated forfeits. Figure 2 summarizes these differences by game and opponent.

Qwen3-14B presents a different failure profile. Under *instruct*, retry rates are high in all five games, and forfeits occur under every opponent. The recorded failures include truncation-related responses and mixed sequences containing more than one type of invalid response, rather than being dominated by illegal actions as in Qwen3-8B *instruct*. Under *reasoning*, retries remain infrequent in the four classic–*misère* games and no forfeits are observed. In Centralis, retry rates increase to approximately 16–26%, but the model still completes the matches within the allowed retry budget. These results distinguish the frequency of invalid responses from the ability to recover from them. Both *reasoning* configurations still produce invalid responses in Centralis, but these responses do not lead to forfeits in the analyzed matches. The *instruct* configurations, in contrast, reach similar match-level failures through different mechanisms: repeated illegal actions for Qwen3-8B and truncation-related or mixed invalid responses for Qwen3-14B.

6.4 Inference Cost

The *reasoning* configurations require more generated tokens and longer end-to-end execution times than their corresponding *instruct* configurations for both model sizes. The magnitude of this difference varies by game and opponent, with larger costs in several NotConnect4 and Centralis conditions. Token consumption does not increase monotonically with model size: Qwen3-14B *reasoning*

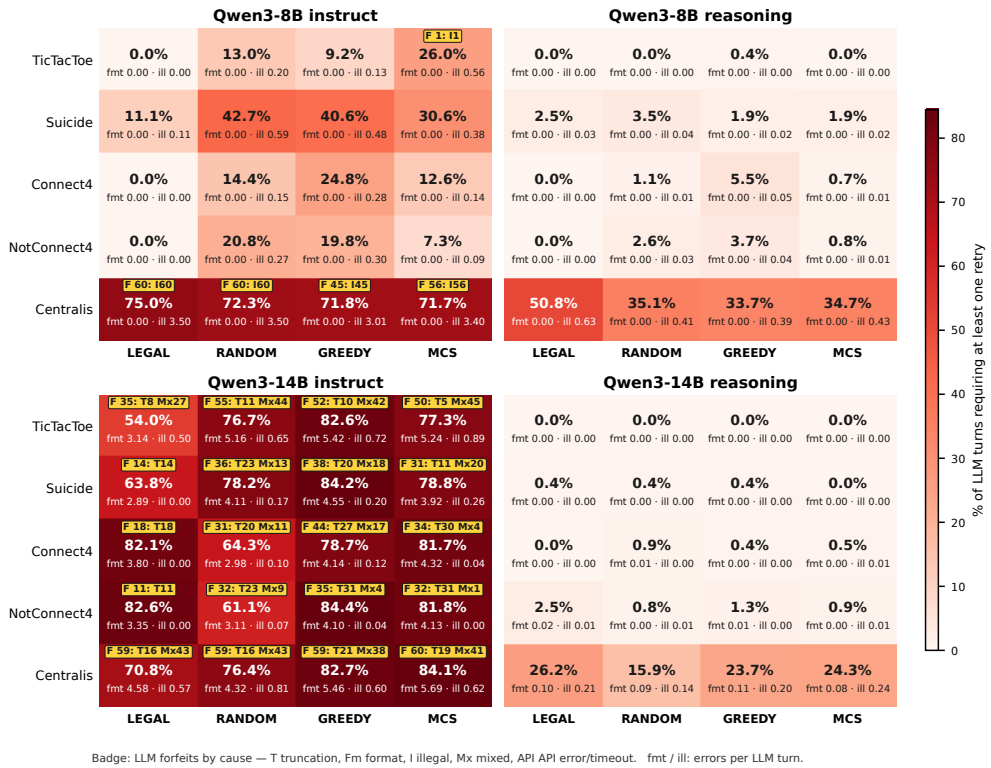

Figure 2: Operational robustness by model condition, game, and GGP opponent. Each cell reports the percentage of LLM turns requiring at least one retry and the mean number of format (`fmt`) and illegal (`ill`) responses per LLM turn. Forfeit markers indicate the number and cause of model-generated forfeits: truncation (T), illegal actions (I), and mixed invalid-response sequences (Mx).

uses fewer completion tokens than Qwen3-8B *reasoning* in several pairings. Because match time also includes game length, repeated model calls, and GGP-opponent execution, we interpret it as an operational cost rather than as direct LLM inference latency. Detailed statistics are reported in Appendix C.

7 Analysis and Discussion

The results distinguish competitive performance, operational robustness, and inference cost as separate aspects of model behavior. The *reasoning* configurations generally produce fewer model-generated forfeits and recover more frequently from invalid responses, but these differences do not correspond to uniform changes in win rate. The relationship also varies with model size. In Centralis, Qwen3-8B *reasoning* completes most matches but records few wins, whereas Qwen3-14B *reasoning* achieves high win rates against the four GGP reference opponents. The two *instruct* configurations also exhibit different failure profiles: Qwen3-8B is affected mainly by illegal action selection in Centralis, while Qwen3-14B produces frequent truncation-related and mixed invalid responses.

The classic–*misère* pairs provide a controlled comparison of changes in the success criterion. For the *reasoning* configurations, win rates decrease or remain low in several pairings after the objective is inverted, despite preserving the board, actions, legality rules, and transitions. We refer to this behavioral difference as an objective-adaptation gap. The experiments establish the observed change in behavior but do not identify its underlying mechanism. Reliance on policies associated with the standard game, difficulty applying the inverted objective, prompt salience, or a combination of these factors remain compatible explanations. Centralis provides a different type of comparison because it introduces an original rule system, spatial organization, and terminal condition. The

contrast between Qwen3-8B and Qwen3-14B *reasoning* shows that producing valid actions and completing matches is not sufficient to characterize competitive performance. Qwen3-8B generally remains within the rules and completes the game but records few wins, whereas Qwen3-14B attains substantially higher win rates under the same game representation and referee. At the same time, the lower win rates of Qwen3-14B *reasoning* in several *misère* pairings show that performance under an original rule system and adaptation to an inverted objective do not vary in the same way.

The operational results also show that similar match outcomes can arise from different failure mechanisms. Qwen3-8B *instruct* produces almost exclusively illegal-action errors, with repeated invalid moves accounting for its Centralis forfeits. Qwen3-14B *instruct*, instead, frequently reaches the generation limit before returning a usable move, producing truncation-related and mixed invalid-response sequences. This difference cannot be attributed to a different output budget between model sizes: both Qwen3-8B and Qwen3-14B use the same 128-token limit in the *instruct* configuration. The observed difference therefore reflects how the two models behave under the same generation constraint rather than unequal token budgets. These patterns are consistent with prior work reporting a distinction between valid action generation and strategic behavior in game-based evaluations (Duan et al., 2024; Lin et al., 2025; Mishra et al., 2025). Our evaluation extends this distinction by considering competitive outcomes together with invalid-response recovery and inference cost under a common referee. The *reasoning* configurations require more generated tokens and longer match execution times, while the associated differences in win rate depend on the game and opponent. For both model sizes, *instruct* uses a 128-token generation limit with `reasoning_effort` set to `none`, whereas *reasoning* uses a 1,024-token limit with `reasoning_effort` set to `low`. Comparisons between these configurations therefore characterize the evaluated inference settings and should not be interpreted as an isolated causal effect of reasoning.

8 Conclusions and Limitations

We examine how Qwen3-8B and Qwen3-14B behave when the objective of a known game is inverted and when they are presented with the rules of an original game. The *reasoning* configurations generally show fewer model-generated forfeits and greater recovery from invalid responses, while requiring more generated tokens and execution time. Competitive performance, however, varies by game, opponent, and model size. In Centralis, Qwen3-8B *reasoning* completes most matches but records few wins, whereas Qwen3-14B *reasoning* achieves high win rates against the four GGP reference opponents. This difference illustrates why operational robustness and competitive performance should be considered separately. The classic–*misère* comparisons reveal an objective-adaptation gap when the success criterion is inverted while the board, actions, legality rules, and transitions remain unchanged. This pattern is also observed for Qwen3-14B *reasoning*, despite its higher win rates in Centralis. Performance under an original rule system therefore does not imply similar behavior when the objective of a known game is reversed.

The experiments do not identify the mechanism underlying the classic–*misère* gap. Reliance on policies associated with the standard game, difficulty applying the inverted objective, prompt salience, or a combination of these factors remain compatible explanations. The comparison between *instruct* and *reasoning* should also not be interpreted as an isolated effect of reasoning, since the configurations differ in both reasoning effort and generation budget.

The generation limits are nevertheless held constant across model sizes within each configuration, allowing Qwen3-8B and Qwen3-14B to be compared under the same output constraints. The evaluation is limited to two related game families, one original turn-based game, and two model sizes from the same model family. Future work can extend the protocol to additional game structures, larger and frontier models, and models from different families to examine whether the same behavioral patterns persist. The use of GDL specifications and a common referee provides a reusable basis for such comparisons while preserving the same validation and measurement procedure.

AI Use Statement

Generative AI tools were used to support language editing, grammar correction, reference discovery, and programming tasks related to the experimental workflow. All suggested references were

verified against their original sources, and AI-assisted code was reviewed, executed, and tested by the authors.

Generative AI was not used to determine the scientific claims, experimental conclusions, or interpretation of the results. All analyses, reported results, and conclusions were reviewed and validated by the authors, who take full responsibility for the final content of the paper.

ETHICS STATEMENT

This work does not involve human subjects, crowdsourcing, animal experiments, or the collection of personally identifiable information. The experiments are conducted in controlled board-game environments using language-model agents and General Game Playing baseline agents. The generated data consist of game states, model actions, legality labels, match outcomes, timing information, token counts, and interaction logs produced during automated gameplay.

The study evaluates rule following, legality, robustness, and performance of language models in synthetic game domains. It does not release a high-risk generative model, scraped dataset, or user-generated private data. The released assets are intended to support reproducibility and further research on the evaluation of LLM agents in structured decision-making tasks.

Although the games are synthetic and low-stakes, the results may inform broader evaluations of autonomous or semi-autonomous LLM agents. Therefore, the findings should be interpreted within the controlled setting of the paper and should not be taken as evidence of suitability for high-stakes decision-making. The authors are responsible for the design, validation, documentation, and interpretation of the experiments and released artifacts.

REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce the experimental setting in the main paper and appendix. Section 5 describes the game domains, GDL-based referee, LLM and GGP agents, text-based interaction protocol, parsing and legality checks, match setup, retry and forfeit rules, logged files, and evaluation metrics. Appendix A provides additional implementation details, including the prompting setup, move parsing procedure, legality validation, API backend. Appendices B and C report additional calibration and inference cost results, supporting verification of the main findings.

REFERENCES

Elif Akata, Laura Schulz, Julian Coda-Forno, Seong Joon Oh, Matthias Bethge, and Eric Schulz. Playing repeated games with large language models. *Nature Human Behaviour*, 9(7):1380–1390, 2025.

Bijan Bateni, Brandon Pratt, and Jim Whitehead. LLM game rule understanding through out-of-distribution fine-tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 21, pp. 2–11, November 2025.

Katherine M. Collins, Charles E. Zhang, Graham Todd, Lexin Ying, Marco B. da Costa, Ryan Liu, and Thomas L. Griffiths. Evaluating language models' evaluations of games, 2025.

Jinhao Duan, Rui Zhang, James Diffenderfer, Bhavya Kailkhura, Lichao Sun, Elias Stengel-Eskin, and Kaidi Xu. GTBench: Uncovering the strategic reasoning capabilities of LLMs via game-theoretic evaluations. In *Advances in Neural Information Processing Systems*, volume 37, pp. 28219–28253, 2024.

Kanishk Gandhi, Dorsa Sadigh, and Noah D. Goodman. Strategic reasoning with language models, 2023.

Isabelle Gevers and Walter Daelemans. Do you get the hint? benchmarking LLMs on the board game concept, 2025.

Weizhe Lin, Jonathan Roberts, Yang Yang, Samuel Albanie, Zongqing Lu, and Kai Han. GAME-BoT: Transparent assessment of LLM reasoning in games. In *Proceedings of the 63rd Annual*

Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 7656–7682, July 2025.

Linus Malmqvist. Winning at all cost: A small environment for eliciting specification gaming behaviors in large language models. In *International Conference on Applied Cryptography and Network Security*, pp. 52–69, Cham, June 2025. Springer Nature Switzerland.

Prakhar Mishra, Jiacheng Liu, Jiayi Wu, Xiaodong Yu, Zichao Liu, and Emad Barsoum. TTT-Bench: A benchmark for evaluating reasoning ability with simple and novel tic-tac-toe-style games. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 2809–2831, November 2025.

Paulina Morillo, Alex Terreros, Cèsar Ferri, and José Hernández-Orallo. LLM GameLab: An interactive platform for testing large language models in board games. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 486–490, Cham, September 2025. Springer Nature Switzerland.

Qingyu Qian, Chao Huang, Jiahao Xu, Cheng Lv, Ming Wu, Wei Liu, and Xiaodan Zheng. Benchmark^2: Systematic evaluation of LLM benchmarks, 2026.

Anian Ruoss, Fabio Pardo, Harris Chan, Bowen Li, Volodymyr Mnih, and Tim Genewein. LMAct: A benchmark for in-context imitation learning with long multimodal demonstrations, 2024.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, and Gary X. Wang. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023.

Stanford Logic Group. General game playing. Stanford University, 2026. URL `http://gamemaster.stanford.edu/homepage/about.php`. Accessed 2026.

Hao Sun, Yuchen Wu, Yuhang Cheng, and Xiaowen Chu. Game theory meets large language models: A systematic survey, 2025.

Oguzhan Topsakal, Emily Colby, and Hannah Jackson. Evaluating the performance of large language models (LLMs) through grid-based game competitions: An extensible benchmark and leaderboard on the path to artificial general intelligence (AGI). *The Journal of Cognitive Systems*, 9(2):8–19, 2025.

Floris van Wetten, Aske Plaat, and Max van Duijn. Baba is LLM: Reasoning in a game with dynamic rules. In *International Joint Conference on Computational Intelligence*, pp. 60–77, Cham, October 2025. Springer Nature Switzerland.

Lexin Ying, Richard Truong, Pratyusha Sharma, Kai I. Zhao, Nicholas Cloos, Kelsey R. Allen, and Joshua B. Tenenbaum. AI GameStore: Scalable, open-ended evaluation of machine general intelligence with human games, 2026.

A Implementation and Reproducibility Details

A.1 Board Representations and Move Formats

Figure 3 illustrates the textual and spatial representations of the three game families used in the evaluation, together with their corresponding move formats. TicTacToe and Suicide share the same 3×3 board, while Connect4 and NotConnect4 use the same gravity-based board. Centralis uses the radial structure introduced for this study.

A.2 Prompting setup

Language-model interaction was implemented with a minimal two-message structure per turn. At match start, the model received a system prompt containing a natural-language description of the game rules, the assigned role, and an example of the expected move format. At each turn, the model then received a user message containing only a textual rendering of the current board state and a

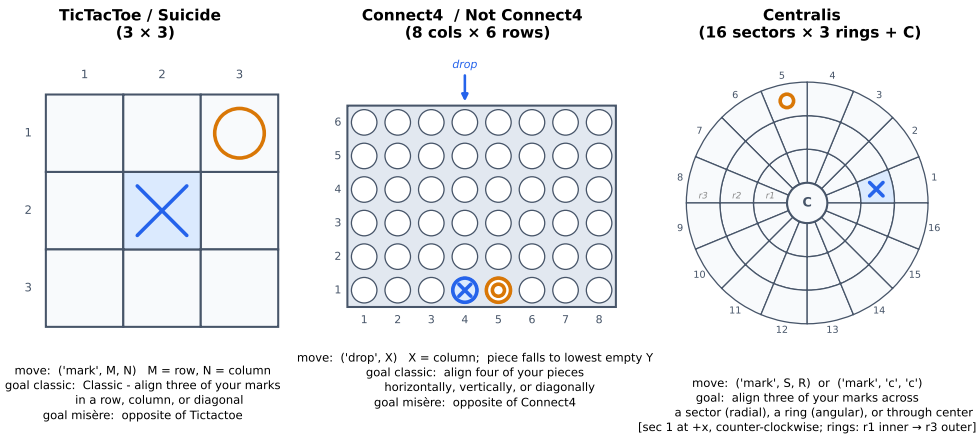

Figure 3: Game board configurations for the three game families. For Centralis, the diagram provides a schematic representation of the radial board and action coordinates. The central action `('mark','c','c')` becomes legal only after a player has formed one of the allowed three-mark paths and must be played on that player's following turn to win. The complete Centralis rules are described in Section 3.

request for the next move. The logical game state was not exposed directly, and the full turn history was not accumulated as additional context. This design kept the interaction format fixed between games and model conditions.

The action formats were game-specific but followed the same general structure of a JSON-like list. For TicTacToe and Suicide, the expected format was `["mark","row","column"]`; for Connect4 and NotConnect4, `["drop","column"]`; and for Centralis, `["mark","section","ring"]`. The system prompt explicitly stated the required action name and provided an example for the corresponding game.

A.3 Move Parsing and Legality Checks

Model outputs were parsed into candidate moves using a rule-based parser. The parser first attempted to read the full response as JSON. If that failed, it searched for list-like spans within the response and attempted to recover a move either through JSON parsing or Python-style literal parsing. Parsed elements were normalized to strings before legality checking. After parsing, the candidate move was validated against the legal action set of the current state using the GGP referee. Legality was therefore determined by the referee and not by template matching alone. Outputs that could not be parsed were recorded as formatting errors. Parsed outputs that did not belong to the legal action set were recorded as illegal moves.

A.4 Compute and API Details

The experiments used OpenRouter as the inference backend for Qwen3-8B and Qwen3-14B. Both model sizes were evaluated in the `instruct` and `reasoning` configurations described above. A fixed seed was included in the Qwen3 API calls to reduce variation due to decoding randomness, although exact reproducibility still depends on the provider and backend implementation.

Match execution, logging, and aggregation were run on a virtual machine with 32 GB RAM, 140 GB of disk space, and 16 CPU cores. For each model–game–opponent condition, matches were executed as 30 role-swapped pairs, yielding 60 analyzed matches with the LLM moving first in 30 and second in 30. Executions affected by API or backend failures were excluded from the analysis and rerun under the same condition and player order until the target number of completed matches was obtained.

B Additional Calibration Results: GGP vs. GGP Matches

This appendix provides a more detailed view of the GGP calibration results. Table 3 shows the total number of matches and moves, per pairings and games. Table 4 reports the full pairwise win rates for all games, and Table 5 gives the corresponding win/draw/loss counts by role and in aggregate. Figure 4 shows first-mover effects by game and agent, while Tables 6 and 7 summarize match length and execution time per pairings. Figure 5 examines intrinsic asymmetries through self-play, Figure 6 reports hierarchy validation among agents, and Figure 7 provides additional analysis of Centralis.

Table 3: Number of matches and total moves by pairing and game.

Emparejamientos	**Tictacoe**		**Suicide**		**Connect4**		**NotConnect4**		**Centralis**	
	Matches	Moves	Matches	Moves	Matches	Moves	Matches	Moves	Matches	Moves
greedy_vs_greedy	100	900	100	746	100	2484	100	3742	100	1100
greedy_vs_legal	100	534	100	664	100	1054	100	1181	100	2158
greedy_vs_mcs	100	814	100	842	100	2279	100	3382	100	1520
greedy_vs_random	100	574	100	781	100	1600	100	2820	100	1192
legal_vs_greedy	100	599	100	580	100	1254	100	977	100	2462
legal_vs_legal	100	700	100	700	100	1900	100	1900	100	4800
legal_vs_mcs	100	600	100	670	100	1063	100	900	100	2152
legal_vs_random	100	631	100	620	100	956	100	956	100	4056
mcs_vs_greedy	100	900	100	900	100	2530	100	3451	100	1560
mcs_vs_legal	100	500	100	644	100	784	100	950	100	2078
mcs_vs_mcs	100	666	100	900	100	2383	100	4141	100	1840
mcs_vs_random	100	544	100	779	100	974	100	2058	100	1940
random_vs_greedy	100	701	100	748	100	1888	100	2578	100	1335
random_vs_legal	100	741	100	732	100	1035	100	1035	100	4075
random_vs_mcs	100	654	100	772	100	1211	100	1864	100	1976
random_vs_random	100	778	100	778	100	2143	100	2143	100	3662

Table 4: GGP round-robin: pairwise win rate of the row agent against the column opponent on each game (200 matches per pairing).

		TicTacToe	Suicide	Connect4	NotConnect4	Centralis
Agent	vs.	WR	WR	WR	WR	WR
Legal	Random	0.57	0.37	0.85	0.14	0.00
	Greedy	0.01	0.05	0.00	0.02	0.00
	MCS	0.00	0.00	0.03	0.00	0.00
Random	Legal	0.39	0.60	0.14	0.85	0.76
	Greedy	0.00	0.07	0.06	0.10	0.01
	MCS	0.01	0.01	0.01	0.00	0.03
Greedy	Legal	0.99	0.94	1.00	0.97	1.00
	Random	0.88	0.78	0.94	0.90	0.99
	MCS	0.20	0.49	0.17	0.04	0.86
MCS	Legal	1.00	1.00	0.97	1.00	0.99
	Random	0.97	0.94	0.99	1.00	0.97
	Greedy	0.00	0.24	0.82	0.95	0.14

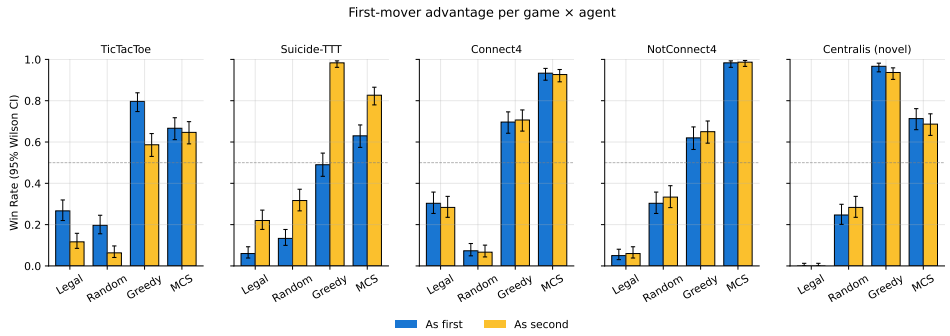

Figure 4: First-mover advantage by game and GGP agent. Bars show win rate with 95% Wilson confidence intervals when the agent plays first or second.

Table 5: GGP round-robin outcomes. Each row fixes a game and an opponent. For each agent, cells report wins/draws/losses (W/D/L) when that agent moves first (X), moves second (O), and in the aggregate total. Self-play cells are omitted.

		Legal			Random			Greedy			MCS		
Game	Opponent	X	O	total	X	O	total	X	O	total	X	O	total
TicTacToe	Legal	–	–	–	58/7/35	19/2/79	77/9/114	100/0/0	99/0/1	199/0/1	100/0/0	100/0/0	200/0/0
	Random	79/2/19	35/7/58	114/9/77	–	–	–	99/1/0	77/23/0	176/24/0	100/0/0	94/5/1	194/5/1
	Greedy	1/0/99	0/0/100	1/0/199	0/23/77	0/1/99	0/24/176	–	–	–	0/100/0	0/60/40	0/160/40
	MCS	0/0/100	0/0/100	0/0/200	1/5/94	0/0/100	1/5/194	40/60/0	0/100/0	40/160/0	–	–	–
Suicide	Legal	–	–	–	40/4/56	80/2/18	120/6/74	88/2/10	100/0/0	188/2/10	100/0/0	100/0/0	200/0/0
	Random	18/2/80	56/4/40	74/6/120	–	–	–	59/28/13	97/3/0	156/31/13	89/9/2	99/1/0	188/10/2
	Greedy	0/0/100	10/2/88	10/2/188	0/3/97	13/28/59	13/31/156	–	–	–	0/51/49	98/2/0	98/53/49
	MCS	0/0/100	0/0/100	0/0/200	0/1/99	2/9/89	2/10/188	0/51/49	98/2/0	98/53/49	–	–	–
Connect4	Legal	–	–	–	15/0/85	14/0/86	29/0/171	100/0/0	100/0/0	200/0/0	100/0/0	95/0/5	195/0/5
	Random	86/0/14	85/0/15	171/0/29	–	–	–	94/0/6	94/0/6	188/0/12	100/0/0	99/0/1	199/0/1
	Greedy	0/0/100	0/0/100	0/0/200	6/0/94	6/0/94	12/0/188	–	–	–	80/2/18	84/1/15	164/3/33
	MCS	5/0/95	0/0/100	5/0/195	1/0/99	0/0/100	1/0/199	15/1/84	18/2/80	33/3/164	–	–	–
NotConnect4	Legal	–	–	–	85/0/15	86/0/14	171/0/29	97/0/3	98/1/1	195/1/4	100/0/0	100/0/0	200/0/0
	Random	14/0/86	15/0/85	29/0/171	–	–	–	85/1/14	94/0/6	179/1/20	100/0/0	100/0/0	200/0/0
	Greedy	1/1/98	3/0/97	4/1/195	6/0/94	14/1/85	20/1/179	–	–	–	95/2/3	96/0/4	191/2/7
	MCS	0/0/100	0/0/100	0/0/200	0/0/100	0/0/100	0/0/200	4/0/96	3/2/95	7/2/191	–	–	–
Centralis	Legal	–	–	–	71/29/0	81/19/0	152/48/0	100/0/0	100/0/0	200/0/0	100/0/0	98/2/0	198/2/0
	Random	0/19/81	0/29/71	0/48/152	–	–	–	100/0/0	99/0/1	199/0/1	96/0/4	98/0/2	194/0/6
	Greedy	0/0/100	0/0/100	0/0/200	1/0/99	0/0/100	1/0/199	–	–	–	18/0/82	10/0/90	28/0/172
	MCS	0/2/98	0/0/100	0/2/198	2/0/98	4/0/96	6/0/194	90/0/10	82/0/18	172/0/28	–	–	–

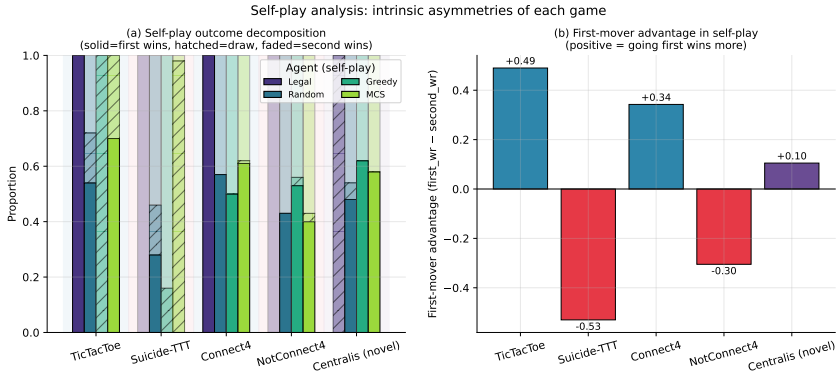

Figure 5: Self-play analysis of the five games. (a) Outcome decomposition in self-play, showing the proportions of wins by the first player, draws, and wins by the second player for GGP agents. (b) First-mover advantage in self-play, measured as the difference between the first-player and second-player win rates.

Table 6: GGP round-robin: moves per match per row agent, broken down by game and column opponent (100 matches per cell). Opponent abbreviations: L = Legal, R = Random, G = Greedy, M = MCS.

		TicTacToe				**Suicide**				**Connect4**				**NotConnect4**				**Centralis**			
Agent	Stat	L	R	G	M	L	R	G	M	L	R	G	M	L	R	G	M	L	R	G	M
Legal	Avg	7.0	6.3	6.0	6.0	7.0	6.2	5.8	6.7	19.0	9.6	12.5	10.6	19.0	9.6	9.8	9.0	48.0	40.6	24.6	21.5
	sd	0.0	1.7	0.1	0.0	0.0	1.7	1.6	1.9	0.0	4.7	2.5	1.5	0.0	4.7	5.5	4.0	0.0	8.0	7.7	6.6
	Md	7	5	6	6	7	5	5	5	19	7	12	10	19	7	7	7	48	43	26	22
	Min	7	5	5	6	7	5	5	5	19	7	10	7	19	7	2	7	48	16	10	2
	Max	7	9	6	6	7	9	9	9	19	26	22	14	19	26	29	27	48	48	42	34
	IQR	0.0	3.0	0.0	0.0	0.0	3.0	0.0	4.0	0.0	2.0	4.0	2.0	0.0	2.0	2.0	2.0	0.0	12.0	12.0	8.0
Random	Avg	7.4	7.8	7.0	6.5	7.3	7.8	7.5	7.7	10.3	21.4	18.9	12.1	10.3	21.4	25.8	18.6	40.8	36.6	13.3	19.8
	sd	1.5	1.3	1.3	1.0	1.4	1.3	1.2	1.3	4.4	7.8	6.9	4.5	4.4	7.8	9.3	6.3	7.9	8.9	4.3	5.5
	Md	7	8	6	6	7	8	7	7	8	21	18	12	8	21	27	19	43	38	14	20
	Min	5	5	6	5	5	5	5	5	7	7	8	8	7	7	7	7	19	16	8	8
	Max	9	9	9	9	9	9	9	9	26	37	38	32	26	37	42	35	48	48	38	32
	IQR	3.0	2.0	2.0	0.5	3.0	2.0	2.0	2.0	2.2	11.2	8.5	6.0	2.2	11.2	14.0	10.0	13.0	14.0	6.0	6.5
Greedy	Avg	5.3	5.7	9.0	8.1	6.6	7.8	7.5	8.4	10.5	16.0	24.8	22.8	11.8	28.2	37.4	33.8	21.6	11.9	11.0	15.2
	sd	0.8	1.2	0.0	1.1	1.0	1.1	1.5	1.0	2.9	7.8	7.8	8.0	6.2	9.3	5.4	3.6	8.3	3.8	2.3	3.9
	Md	5	5	9	9	6	8	7	9	11	15	23	20	8	28	37	33	23	11	11	15
	Min	5	5	9	5	5	5	5	5	7	7	13	9	8	10	1	27	7	7	7	7
	Max	7	9	9	9	9	9	9	9	17	37	44	48	30	48	48	48	39	27	16	34
	IQR	0.0	2.0	0.0	2.0	2.0	2.0	2.0	2.0	6.0	12.0	11.0	8.0	8.0	13.2	5.0	4.0	12.5	4.0	4.0	2.0
MCS	Avg	5.0	5.4	9.0	6.7	6.4	7.8	9.0	9.0	7.8	9.7	25.3	23.8	9.5	20.6	34.5	41.4	20.8	19.4	15.6	18.4
	sd	0.0	1.0	0.0	1.7	0.8	0.8	0.0	0.0	1.5	3.8	9.5	10.7	3.7	6.7	3.7	3.4	5.9	5.8	4.1	4.6
	Md	5	5	9	7	6	8	9	9	7	9	23	21	8	21	34	42	21	19	14	19
	Min	5	5	9	5	6	6	9	9	7	7	11	7	8	8	26	31	9	7	8	10
	Max	5	9	9	9	8	9	9	9	13	23	48	48	26	34	48	48	35	33	33	30
	IQR	0.0	0.0	0.0	4.0	0.0	0.0	0.0	0.0	2.0	4.0	13.0	17.0	0.0	10.0	4.0	4.0	8.0	8.0	2.0	7.0

Table 7: GGP round-robin: match time (s) per game × row agent × column opponent (100 matches per cell). Avg, sd, Md report mean, standard deviation and median; Min and Max are the per-match extremes; IQR is the interquartile range.

		Legal						**Random**						**Greedy**						**MCS**					
Game	Agent	Avg	sd	Md	Min	Max	IQR	Avg	sd	Md	Min	Max	IQR	Avg	sd	Md	Min	Max	IQR	Avg	sd	Md	Min	Max	IQR
TicTacToe	Legal	0.4	0.2	0.4	0.1	1.0	0.4	0.3	0.4	0.1	0.1	1.6	0.5	3.9	1.8	3.5	1.9	16.2	1.1	24.2	0.2	24.1	24.0	24.7	0.1
	Random	0.3	0.4	0.1	0.1	1.9	0.5	0.1	0.1	0.1	0.1	0.7	0.1	5.2	2.0	4.5	1.9	9.2	3.5	26.1	3.6	24.2	16.1	32.7	2.7
	Greedy	8.7	0.6	8.5	8.1	11.1	0.7	8.7	0.6	8.5	8.1	10.6	0.7	14.6	2.7	14.0	10.2	21.9	4.1	37.5	4.5	40.5	24.3	43.2	8.1
	MCS	24.2	0.1	24.1	24.0	24.8	0.1	25.5	3.0	24.1	24.1	32.7	0.3	35.5	1.3	35.3	33.5	40.6	1.8	52.4	14.1	56.1	40.0	141.8	24.0
Suicide	Legal	0.8	1.5	0.5	0.1	14.4	0.6	0.5	0.3	0.5	0.1	2.5	0.4	10.5	1.7	10.2	8.3	15.2	2.7	23.2	7.6	16.8	16.1	34.1	16.1
	Random	0.6	0.4	0.5	0.1	1.7	0.4	0.5	0.5	0.3	0.1	3.8	0.4	7.8	3.6	5.7	4.6	17.4	4.6	27.2	5.4	25.0	16.1	33.6	8.0
	Greedy	16.1	4.2	15.8	9.3	32.3	5.7	11.8	5.0	8.7	8.4	23.3	8.0	29.4	8.2	28.7	22.5	106.0	3.7	38.3	4.2	40.6	24.5	41.3	8.1
	MCS	26.1	3.4	24.3	24.1	33.3	0.8	31.0	3.0	32.1	24.1	33.3	0.1	37.5	0.6	37.4	36.7	40.6	0.6	64.0	0.0	64.0	64.0	64.2	0.0
Connect4	Legal	0.2	0.1	0.2	0.2	0.6	0.1	0.2	0.1	0.2	0.1	1.0	0.1	37.4	12.0	33.0	24.1	80.3	16.1	42.5	6.5	40.2	24.1	56.3	8.0
	Random	0.2	0.1	0.2	0.1	1.1	0.1	0.3	0.2	0.3	0.1	0.9	0.2	64.6	27.0	64.2	16.7	138.3	32.2	48.7	17.9	48.2	32.1	128.5	24.0
	Greedy	33.6	12.3	32.2	16.5	64.2	16.8	56.5	29.4	56.2	16.6	144.5	41.9	178.7	61.0	165.1	80.5	315.9	83.0	173.5	61.1	154.3	56.9	349.3	64.9
	MCS	35.5	6.2	32.1	32.1	56.1	8.0	43.2	15.3	40.1	32.1	96.5	16.1	191.4	70.1	176.6	80.2	339.0	101.2	189.8	83.4	168.5	56.1	353.0	136.2
NotConnect4	Legal	1.2	0.6	1.0	0.7	4.0	0.3	0.2	0.1	0.1	0.1	0.7	0.1	36.1	22.2	24.1	5.8	112.4	8.0	33.2	17.7	24.2	24.1	123.2	8.1
	Random	0.2	0.2	0.1	0.1	0.7	0.2	0.3	0.1	0.3	0.1	0.8	0.2	100.5	37.1	105.0	24.1	184.9	56.1	70.8	25.1	72.2	24.1	136.4	40.1
	Greedy	48.3	25.0	32.1	32.1	120.4	33.9	112.9	35.1	113.1	40.1	204.4	50.9	283.1	42.4	281.3	3.4	437.4	25.1	260.8	23.1	259.2	211.3	345.8	29.0
	MCS	39.1	15.4	32.3	32.1	119.9	4.9	82.6	26.7	84.3	32.1	136.4	40.1	266.3	23.5	266.4	202.0	337.2	26.2	330.3	24.5	337.3	249.4	370.1	31.8
Centralis	Legal	0.6	0.3	0.5	0.3	2.8	0.4	0.8	0.4	0.7	0.2	2.6	0.3	19.4	4.3	18.0	16.6	35.1	1.3	88.6	29.9	88.4	1.7	160.7	32.2
	Random	0.8	0.3	0.7	0.4	2.4	0.3	0.6	0.2	0.6	0.3	1.0	0.2	40.4	11.9	39.3	21.8	75.2	15.6	79.4	22.2	80.4	32.2	128.6	26.2
	Greedy	450.1	4218.9	25.8	21.6	42216.7	2.2	39.8	11.9	37.4	22.1	67.8	15.2	70.6	18.3	70.3	38.1	111.0	30.4	113.3	32.6	110.8	45.9	288.1	15.9
	MCS	89.9	27.4	88.5	40.2	164.0	38.0	81.8	23.2	80.4	32.2	136.6	32.1	116.0	34.0	104.6	56.4	277.9	17.5	178.6	310.4	152.5	80.3	3230.3	56.1

C Inference Cost Results

.

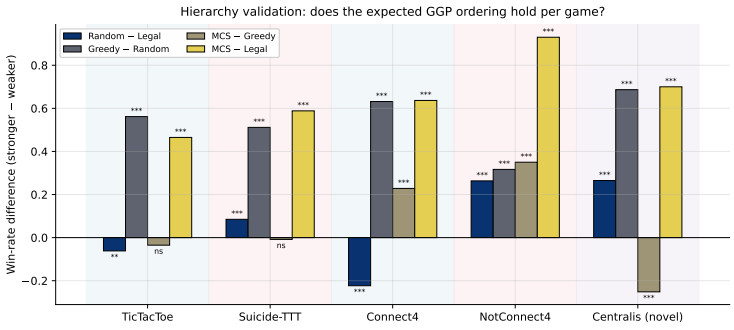

Figure 6: Pairwise GGP comparisons

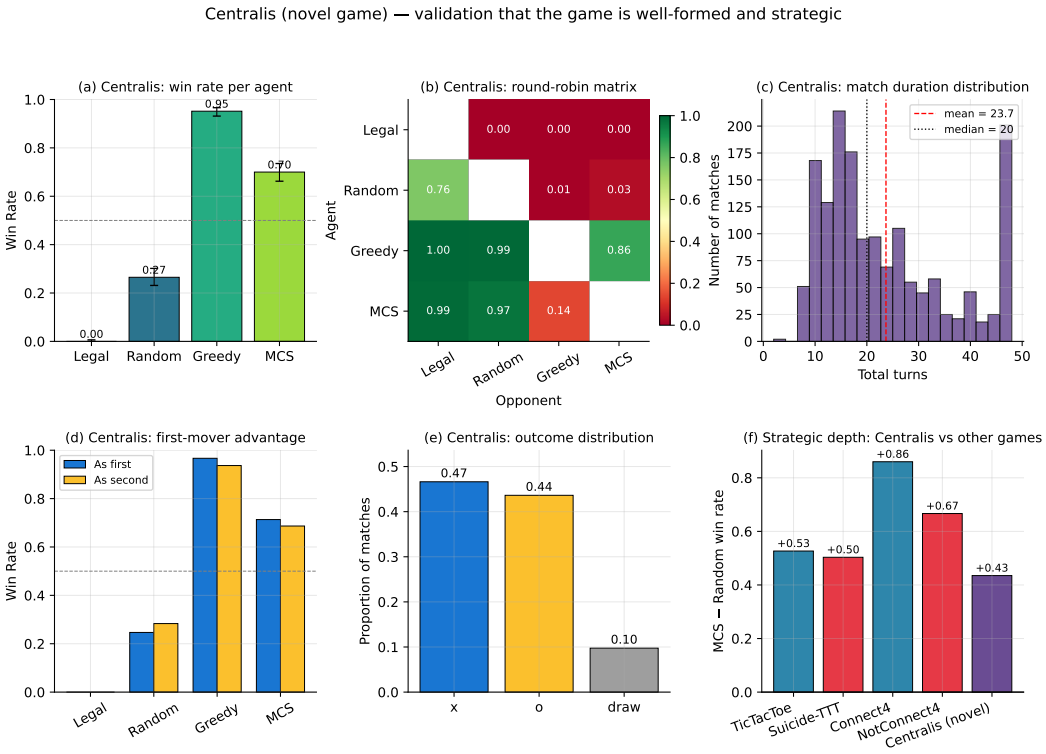

Figure 7: Centralis validation via multiple aggregate views. (a) Overall win rate for each GGP agent. (b) Round-robin win-rate matrix, where each cell shows the win rate of the row agent against the column agent. (c) Distribution of match duration in total turns, including mean and median. (d) Win rate when playing first and when playing second, showing first-mover effects. (e) Outcome distribution among wins for `x`, wins for `o`, and draws. (f) MCS–Random win-rate difference, measured as the win-rate gap between `MCS` and `Random`, placing Centralis alongside the other games.

Table 8: Compute and time metrics per game × GGP opponent for Qwen3-8B and Qwen3-14B, in both Instruct and Reasoning variants. Tokens: completion tokens emitted by the LLM per match. Match time: end-to-end wall-clock per match. GGP move: mean per-move latency (ms) of the opponent, averaged over the matches in the cell. Avg, sd, Md report mean, standard deviation, and median; Min and Max are the per-match extremes.

			Instruct													**Reasoning**											
			tokens / match					Match time (s)					GGP move (ms)			tokens / match					Match time (s)					GGP move (ms)		
Model	Game	vs.	Avg	sd	Md	Min	Max	Avg	sd	Md	Min	Max	Avg	sd	Md	Avg	sd	Md	Min	Max	Avg	sd	Md	Min	Max	Avg	sd	Md
Qwen3-8B	**TTT**	Legal (L)	18	5	18	14	28	4.4	8.3	3.0	1.3	66.7	1	1	1	5 534	1 814	5 306	3 020	11 069	90.3	37.4	82.4	39.6	190.4	1	1	1
		Random (R)	30	10	28	14	49	5.5	5.8	4.5	2.0	48.1	1	0	1	6 874	2 963	6 548	2 660	14 987	112.5	55.1	101.6	41.6	282.8	1	0	1
		Greedy (G)	26	10	21	14	63	12.5	12.7	10.2	4.0	73.1	1 343	1 008	1 059	12 035	3 212	11 697	4 635	20 916	203.4	62.0	191.0	75.2	360.4	1 038	745	998
		MCS (M)	28	11	21	21	77	35.6	20.9	28.9	18.9	126.5	7 974	207	8 001	8 571	3 283	8 044	3 252	18 841	179.3	66.4	170.9	58.3	365.8	7 415	778	8 001
	Suicide	Legal (L)	24	4	24	21	28	213.2	1614.4	4.3	3.1	12509.6	1	0	1	13 223	5 086	13 082	5 330	28 412	219.5	94.7	198.0	82.1	494.0	1	1	1
		Random (R)	43	15	42	21	105	10.5	13.8	6.8	2.5	66.8	1	0	1	17 249	4 194	16 746	8 538	25 207	267.1	81.7	260.6	128.4	445.4	1	0	1
		Greedy (G)	38	11	35	21	70	18.3	12.1	15.2	10.7	74.8	2 537	587	2 236	17 020	3 231	16 866	11 188	29 599	290.1	76.3	276.7	167.9	581.0	1 980	322	1 987
		MCS (M)	39	8	35	28	63	44.1	19.1	38.2	28.8	97.5	8 001	1	8 001	17 741	2 961	17 379	10 747	25 798	325.1	64.4	308.4	207.8	522.1	8 001	2	8 001
	Connect4	Legal (L)	18	3	18	15	20	4.1	1.5	3.8	2.4	13.7	1	1	1	18 441	8 585	14 947	7 634	39 424	311.3	149.1	257.1	121.1	690.5	2	0	2
		Random (R)	38	20	30	20	105	9.4	8.4	7.0	3.7	57.1	1	0	1	29 178	23 155	19 349	8 455	112 902	567.8	786.6	307.8	111.8	5685.1	2	3	2
		Greedy (G)	54	24	50	15	120	84.2	36.2	77.5	28.8	213.0	6 772	549	7 006	49 376	25 219	47 843	4 006	118 255	911.1	454.8	850.3	72.9	1980.9	6 835	707	7 010
		MCS (M)	32	16	25	15	80	63.4	27.8	54.8	29.4	135.9	8 015	5	8 015	41 287	19 453	40 535	10 762	83 757	1340.0	4134.0	736.1	190.7	32704.6	8 015	5	8 015
	NotConnect4	Legal (L)	18	3	18	15	20	3.7	1.0	3.6	2.1	7.2	2	1	1	30 584	23 509	21 075	3 332	90 959	507.5	401.5	336.4	52.5	1578.7	1	0	1
		Random (R)	47	29	30	20	145	14.7	17.2	7.8	4.0	96.0	1	0	1	60 017	33 581	56 916	6 572	163 155	1002.5	596.6	936.3	96.0	2969.9	2	1	2
		Greedy (G)	49	38	30	20	180	73.0	36.4	57.5	28.5	167.6	7 987	95	8 008	63 402	40 315	55 676	9 210	172 773	1149.8	745.9	1109.7	174.4	3322.4	7 942	272	8 011
		MCS (M)	35	16	30	20	120	57.3	17.7	55.8	29.1	137.7	8 013	5	8 012	46 075	26 815	47 423	7 998	119 020	1401.6	4246.8	795.7	132.4	33512.7	8 014	4	8 014
	Centralis	Legal (L)	119	0	119	119	119	19.3	9.5	17.3	13.9	86.2	2	7	1	194 482	25 035	196 288	85 558	242 059	3290.9	497.7	3344.1	1384.7	4076.8	2	0	1
		Random (R)	177	119	122	98	675	28.3	19.0	19.9	11.8	91.6	4	1	4	148 115	45 570	149 315	10 229	222 357	2467.6	827.0	2535.6	189.1	4250.5	4	2	4
		Greedy (G)	138	61	112	49	311	48.8	18.5	42.4	31.7	105.1	5 617	1 292	5 421	80 133	43 491	80 484	11 942	175 241	1361.9	769.3	1187.8	197.0	2912.6	2 684	1 683	1 917
		MCS (M)	168	94	120	104	514	71.3	40.3	51.8	40.4	195.0	8 018	6	8 017	61 604	28 251	58 230	15 468	151 726	1139.0	577.0	1012.3	270.3	3041.9	8 017	4	8 017
Qwen3-14B	**TTT**	Legal (L)	1 104	712	1 294	21	3 130	85.3	66.1	69.8	2.1	292.6	1	1	1	4 662	1 812	3 948	2 184	8 532	118.3	53.1	105.7	39.8	239.0	1	1	1
		Random (R)	1 628	472	1 568	28	2 716	63.1	34.2	51.7	4.2	188.1	2	1	1	5 155	1 762	4 868	2 504	12 106	111.8	57.0	100.9	49.4	358.3	2	2	1
		Greedy (G)	1 548	494	1 432	279	2 446	65.0	31.1	54.8	27.0	178.0	2 591	2 005	2 690	9 266	2 729	8 812	2 888	14 455	213.3	77.0	204.3	85.9	380.1	1 007	741	1 008
		MCS (M)	1 350	489	1 298	14	2 483	60.4	39.6	46.8	25.9	268.6	8 003	3	8 002	6 983	2 170	6 488	2 928	12 236	175.8	56.5	158.0	74.4	406.5	7 281	803	8 000
	Suicide	Legal (L)	1 022	771	914	21	3 349	78.6	56.9	79.2	2.6	204.5	2	3	1	9 331	2 443	9 054	5 478	16 530	206.5	65.8	201.6	107.1	415.2	1	0	1
		Random (R)	1 591	669	1 546	28	3 228	58.1	34.5	47.1	4.0	198.7	1	1	1	11 160	3 345	10 364	5 386	24 613	231.2	88.8	216.3	119.2	544.9	1	1	1
		Greedy (G)	1 796	654	1 828	284	3 093	73.1	42.0	58.1	21.2	224.7	3 500	1 753	2 921	12 585	3 647	12 048	4 888	28 502	289.4	118.9	267.1	114.3	868.5	1 812	384	1 832
		MCS (M)	1 564	540	1 501	533	2 986	95.9	47.8	80.5	31.8	252.6	7 948	292	8 002	12 495	2 846	12 418	6 116	19 329	291.3	92.4	277.3	133.4	625.7	8 001	1	8 001
	Connect4	Legal (L)	1 507	765	1 410	148	3 594	62.5	56.8	45.8	10.4	314.1	3	4	2	10 073	5 104	7 707	4 568	23 278	238.3	123.2	228.0	88.7	674.3	2	1	2
		Random (R)	2 054	1 158	2 068	30	5 165	121.5	84.8	100.2	6.3	392.7	3	3	2	19 754	15 827	14 414	4 976	84 370	413.1	364.2	261.3	80.6	1833.6	2	1	2
		Greedy (G)	2 387	1 051	2 252	409	5 834	125.0	62.4	116.6	35.0	346.0	7 558	777	8 007	36 519	18 451	34 496	4 855	119 129	1040.0	543.5	908.0	123.7	3401.4	7 010	544	7 013
		MCS (M)	2 036	858	1 940	15	4 249	105.2	48.2	100.1	32.8	279.2	8 016	13	8 013	26 616	13 427	25 136	6 192	60 087	764.6	455.2	639.4	171.2	2371.1	8 015	5	8 014
	NotConnect4	Legal (L)	1 374	704	1 408	143	3 599	59.3	46.2	47.2	4.8	289.3	2	1	2	35 392	15 389	35 106	5 554	64 195	1283.6	851.7	1126.8	119.0	3955.8	1	0	1
		Random (R)	1 987	1 060	1 975	40	4 798	100.3	65.9	85.6	6.2	280.3	3	3	2	59 440	28 445	56 458	8 002	132 018	1407.4	689.6	1351.9	124.5	3090.4	2	0	1
		Greedy (G)	2 154	840	1 940	665	4 387	117.2	78.1	88.8	41.5	373.6	8 009	5	8 008	65 159	29 582	64 800	9 029	122 960	2354.6	1352.2	1999.6	312.9	6537.2	7 813	439	8 010
		MCS (M)	2 052	712	1 940	281	3 870	103.0	54.5	92.7	27.4	304.4	8 016	7	8 015	44 193	19 204	41 796	11 586	86 109	1000.0	428.8	966.0	227.2	1946.4	8 011	4	8 010
	Centralis	Legal (L)	1 697	445	1 671	187	3 023	70.4	48.9	55.0	20.4	264.7	3	3	2	74 516	73 081	49 008	6 302	317 976	2309.0	2335.6	1359.5	146.5	8978.6	1	1	1
		Random (R)	1 709	502	1 558	524	3 487	91.2	53.7	73.4	26.7	250.7	8	6	6	47 230	44 107	32 918	7 993	217 494	1085.1	1162.5	673.9	148.9	6177.8	4	2	4
		Greedy (G)	1 769	409	1 678	959	2 867	96.4	51.8	77.6	31.5	277.5	7 910	420	8 003	51 514	54 102	32 776	6 311	242 035	1831.5	2400.9	881.7	144.0	12557.8	3 844	1 632	3 520
		MCS (M)	1 781	429	1 685	1 059	2 979	77.9	43.8	64.5	31.6	268.4	8 022	9	8 021	30 633	24 598	23 069	5 223	126 251	710.6	544.6	542.3	135.6	2943.7	8 015	5	8 014