SPIREBENCH: EVALUATING LONG-HORIZON STRATEGIC PLAY OF LLMS WITH SLAY THE SPIRE

Anonymous authors
Paper under double-blind review

ABSTRACT

Agentic large language models now tackle complex tasks, yet strategic reasoning over long horizons is under-studied: most agentic benchmarks are short and episodic, and performance on long tasks is hard to dissect into stored knowledge, reasoning and luck. We introduce SpireBench, in which models play the complete commercial deck-builder Slay the Spire (StS), where early decisions decide long-term success. Across 20 LLMs at the game's highest difficulty, frontier models beat heuristic baselines and the average human player but remain far behind experts, and the gap opens as the run lengthens: the strongest models win the first act as often as experts, then fall behind. What separates strong models from weak is how well they play each fight; all of them misjudge risk over a run, preserving health rather than spending it to grow stronger. A comprehensive, self-consistent renaming of every game term shows that part of their performance rests on prior knowledge of the game, and models do not improve from the memories they write across runs, which are often wrong and overly cautious. Our harness is the first to give a model the complete original game, and we compare it with existing harnesses. Overall, SpireBench offers an unsaturated long-horizon benchmark with tools to locate where and why agents fail.

1 INTRODUCTION

Large language model (LLM) progress has been rapid, with high scores on tests of a wide range of general-intelligence-aligned skills (Stanford Institute for Human-Centered Artificial Intelligence (HAI), 2026), and LLMs increasingly act as autonomous agents over extended periods (Brohi et al., 2025). However, single-step competence does not necessarily translate into reliable performance over many sequential decisions, a shortfall termed the *horizon gap* (Chen et al., 2026). Performance on long tasks is also hard to dissect: does a model rely on stored knowledge or on reasoning, and can it store and effectively use long-term memories? How is it impacted by stochasticity?

Existing evaluations leave these questions open. Agentic benchmarks (Yehudai et al., 2025) are mostly short-horizon, with little carried between episodes; game benchmarks (Paglieri et al., 2025) rarely hand a model every decision a player makes, or ask whether performance reflects prior knowledge, reasoning or luck; and reported memory gains (Shinn et al., 2023; Zhao et al., 2024) usually come with an external check on what was learned.

We introduce SPIREBENCH, built on the commercial game *Slay the Spire* (StS), which combines tactical combat, deck construction and route planning over 56 floors (Fig. 1). It is turn-based, fully describable in text, and its choices involve weighing short-term gains over long term success at multiple time horizons, making it well suited for a long-horizon benchmark (Appx. B). Unlike previous work on benchmarking StS (Bateni & Whitehead, 2024; Park et al., 2026), SpireBench interacts directly with all aspects of the game, and allows for separating knowledge from reasoning.

Our contributions are as follows:

- **A harness for the complete original game.** Agents play the unmodified commercial game through our harness, making every decision a human player makes (a first) and seeing the same information as a human player would. We allow the model to return a whole turn's actions as multiple tool calls in one response, and the harness reports the effect of each action back to the model. Against its other variants on the same models and seeds, it plays further and costs less per floor (Table 2).

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.

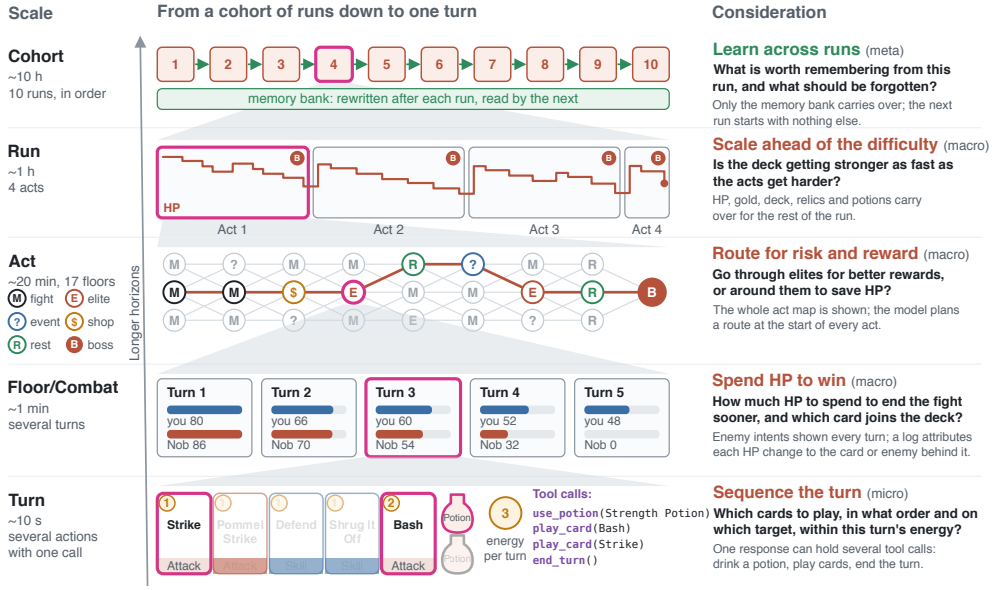


Figure 1: **Multiple time horizons at play in Slay the Spire. Cohort:** the model plays a series of runs on different seeds, carrying memories it writes between them. **Run:** four acts of rising difficulty, each ending in a boss; the player must grow stronger fast enough to keep up. **Act:** about 17 floors (Act 3 ends in two bosses at this difficulty; Act 4 has 4) through which the player plans a route, trading safer floors against riskier ones with larger rewards. **Floor/Combat:** most floors are fights of several turns, each followed by a deck choice that persists for the run; a card weak now may matter later, and a generally good one may be wrong for what lies ahead. **Turn:** the player draws part of the deck; the model plans the turn and plays it in one response of several tool calls (purple). Sometimes the best play takes more damage now to end the fight sooner. Magenta: the part zoomed into.

- **A benchmark at the game's highest difficulty, calibrated against human data.** We evaluate 20 LLMs at Ascension 20 on the same ten held-out seeds, with cost and reasoning spend. For a human performance reference we use Mega Crit's released telemetry (206 k A20 Ironclad runs) and the run histories of 4 expert streamers. Frontier models beat rule-based and symbolic baselines and the average human player, yet remain far from expert play. The benchmark tracks the Artificial Analysis Intelligence Index but is among the least predictable from other evaluations.
- **Knowledge ablation with reskin.** We reskin the entire game with a holistically consistent theme, separating prior knowledge of the game from reasoning about its rules. Hiding the real names costs performance, concentrated in a few models. Random and permuted control mappings show that the cost comes from the knowledge attached to familiar names, not from reading unfamiliar ones.
- **Memory mechanisms, and what the models write.** We allow models to write notes and memories within their run and between runs with various methods. No memory formation between runs is able to improve play. Reading the banks shows why: 17% of memories are false, advice to be cautious outnumbers advice to take risks seven to one, reinforcing existing tendencies, and nothing prunes errors.
- **Analysis of model understanding and failures.** We probe the models to predict future game states to evaluate their understanding of play, demonstrating that more performant models are better able to predict near-term game states, but not capable of predicting long-term success. Comparing model in-game choices against humans, the best models are too conservative, failing to amortize their risk profile across the run into earlier sections so they do not take the risks needed to outpace the game's difficulty curve.

2 Background

Slay the Spire is a seminal roguelike deck-builder. In combat the player draws a hand from a deck of cards and spends a limited pool of energy to attack, block, or build effects that scale over turns;

between fights they add cards, chosen from random offers, to a deck that is hard to thin, and pick a route through a map of fights, elites, shops, rest sites and events. Decisions are taken under uncertainty and compound: a card taken for the fight at hand stays in every later fight, and threats met acts later must be prepared for early. A core component is balancing risk, since riskier options now yield the growth that the rising difficulty curve demands (Fig. 1). These are properties shared with other long tasks agents face (Appx. B).

2.1 Related work

Slay the Spire. Analyses of the game measure its mix of luck and skill (Andersson et al., 2023; Nam & Rerkjirattikal, 2023) and the complexities of its map uncertainty (Bazzaz & Cooper, 2025). MiniStS (Bateni & Whitehead, 2024) tests LLMs on 8 cards and 5 isolated encounters, evaluating micro-play but no macro-level deck-building. The twelve-game benchmark ORAK (Park et al., 2026) drives the real game through CommunicationMod (ForgottenArbiter et al., 2019), but its LLM plays only combat and card rewards, at Ascension 0 on one seed, with a rule-based bot doing the rest. Bottled AI (Bottled AI contributors, 2022), the strongest open symbolic player, does not support every card or event. Concurrent work, AgenticSTS (Cheng et al., 2026), plays complete runs of the sequel, still in early access with fortnightly balance patches, at its lowest difficulty, mainly with one model. In contrast to previous work, we study the released Slay the Spire at its highest difficulty (A20) across 20 models, evaluated against extensive human telemetry, with a controlled reskin; to our knowledge, ours is the first harness giving a model the complete original game.

Games as benchmarks. Deep reinforcement learning reached superhuman play in Atari, Go, chess and shogi (Mnih et al., 2015; Silver et al., 2016; 2018) and hard-exploration games (Ecoffet et al., 2021) by training on the game. LLM agents are evaluated untrained, in NetHack, Crafter and Minecraft (Küttler et al., 2020; Hafner, 2022; Wang et al., 2024a), benchmark suites (Paglieri et al., 2025; Wu et al., 2024a) and commercial games from frames (Zhang et al., 2025), mostly on simplified states or partial games and rarely with learning across runs. The nearest deck-builder benchmark, BalatroBench (Coder, 2026), has shorter horizons and only the basic difficulty; earlier deck-construction work searches whole decks before play (García-Sánchez et al., 2016).

Long-horizon benchmarks. Tool-use, web and agentic benchmarks (Jimenez et al., 2024; Mialon et al., 2024; Yao et al., 2024; Zhou et al., 2024; Koh et al., 2024; Liu et al., 2024b) are mostly tens of steps long; others lengthen the horizon by task duration (Kwa et al., 2026), execution length (Sinha et al., 2026) or a simulated business over tens of millions of tokens (Backlund & Petersson, 2025). Trading (Li et al., 2025a) is long but has a fixed action space; a Slay the Spire agent builds the action space it gets to act in later.

Memory across episodes. Agents with persistent memory (Park et al., 2023; Packer et al., 2023) have previously been found to improve across episodes (Shinn et al., 2023; Wang et al., 2024a; Majumder et al., 2023; Zhao et al., 2024; Wang et al., 2024b) given an external check on what is stored; without one, models struggle to self-correct (Huang et al., 2024) and propagate errors in stored experience (Xiong et al., 2025). We test memory on a long, stochastic task with no such check.

Reskinning. Renaming entities but not mechanics probes memorization against reasoning: nonsense predicates (Valmeekam et al., 2023), changed rules (Wu et al., 2024b), isomorphic worlds (Gu et al., 2026), masked names (Hu et al., 2026), renamed security tasks (Honarvar et al., 2026) and re-textured game objects (Dubey et al., 2018); other benchmarks generate environments a model cannot have seen (Costarelli et al., 2024; Li et al., 2025b). Our reskin covers the whole game with names evoking each mechanic, with random and permuted controls (Sec. 3), paired within seed.

3 Methods

Agent harness. Agents play the released game through `CommunicationModPlus`, our fork of CommunicationMod (ForgottenArbiter et al., 2019), which exchanges JSON states and text commands and adds snapshots around each enemy's turn. The harness holds one conversation per floor, shared by a combat, its rewards, and the post-floor reflection. See Fig. 5 and Appx. A in the appendix for an overview and example floor.

Table 1: **Benchmark results.** Ascension 20 with the Ironclad. Each LLM is run at its highest reasoning setting (grey), played under the themed reskin, with model note-taking across floors within a run, and between runs within a cohort. Runs: number of runs used. Act k win rate: share of runs that cleared Act k. Acts cleared: mean number of acts cleared per run. Floor: final floor reached. BottledAI: symbolic agent (Bottled AI contributors, 2022). Human experts: public run histories of Twitch streamers who play Slay the Spire professionally (see Appx. D for details). An author: non-professional player. All humans: Mega Crit released run telemetry (Oct–Nov 2020, final content build) (Appx. D). **Best** and second best over LLMs. Among the streamers, **bold** marks the best value on each column among the averages. †Self-hosted; priced at Artificial Analysis headline rates. ‡Peak-rate list price (upper bound).

			Act win rate (%)				Acts	Floor			Thinking	Cost ($)	
	Agent	Runs	1	2	3	4	cleared	mean	median	[min–max]	(k/floor)	/floor	/run
Baseline policies													
←	Pick first option	10	0	0	0	0	0.00	7.1	6.0	[1–16]	–	–	–
	Pick random option	10	0	0	0	0	0.00	7.5	7.0	[4–14]	–	–	–
Symbolic agent (Bottled AI)													
	BottledAI w/o Heart	10	20	0	0	0	0.20	11.3	9.0	[5–25]	–	–	–
	BottledAI w/ Heart	10	20	0	0	0	0.20	11.3	9.0	[5–25]	–	–	–
Human players: averages over a period (real game, own seeds)													
	Baalorlord (2020–2026)	687	89	77	70	63	2.98	47.7	**57.0**	[1–57]	–	–	–
	panacea108 (Sep–Nov 2022)	51	**92**	**78**	**71**	**65**	**3.06**	**48.3**	**57.0**	[7–57]	–	–	–
	vmService (Sep–Dec 2022)	50	88	74	64	58	2.84	46.8	**57.0**	[7–57]	–	–	–
	cheg0 (Feb 2024)	46	85	65	56	46	2.52	43.1	56.0	[7–57]	–	–	–
	An author (all runs, 2020–2024)	50	54	24	18	14	1.10	26.5	21.5	[6–57]	–	–	–
	An author (first 10 runs)	10	50	20	20	10	1.00	24.4	20.0	[6–57]	–	–	–
All human players (Mega Crit telemetry, 2020; real game)													
	All human players	206,082	47	15	7	4	0.73	22.1	16.0	[1–57]	–	–	–
	1000+ hours played	8,287	54	20	11	7	0.92	24.7	21.0	[1–57]	–	–	–
	After an earlier Heart kill	3,641	58	29	21	16	1.24	27.9	23.0	[1–57]	–	–	–
LLM agents (thematic reskin)													
	Qwen3.8 27B (xhigh)	10	20	0	0	0	0.20	14.9	16.0	[5–29]	18.1	0.091†	1.45†
	Qwen3.6 35B-A3B (xhigh)	10	0	0	0	0	0.00	10.2	8.0	[6–16]	4.2	0.078†	0.87†
	Qwen3.5 122B-A10B (xhigh)	10	0	0	0	0	0.00	9.8	9.0	[7–16]	3.7	0.072†	0.77†
	Claude Fable 5.1 (max)	10	**90**	**50**	**20**	0	**1.60**	36.2	39.5	[7–52]	17.7	1.664	61.91
	Claude Opus 5 (max)	10	**90**	40	0	0	1.30	32.5	33.0	[7–51]	21.3	1.743	60.62
	Claude Sonnet 5 (max)	10	40	10	0	0	0.50	19.3	16.0	[7–36]	23.2	0.509	10.70
	Claude Haiku 4.5 (12k)	10	10	0	0	0	0.10	11.9	11.5	[4–22]	7.6	0.094	1.23
	DeepSeek V4.1 Flash (max)	10	30	0	0	0	0.30	16.9	16.0	[8–22]	11.7	0.019‡	0.35‡
	Gemini 3.8 Flash (high)	10	50	10	0	0	0.60	22.9	18.0	[**16**–47]	13.7	0.117	2.89
	Gemma 4 E4B	10	0	0	0	0	0.00	7.7	6.5	[1–16]	5.3	**0.004**†	**0.03**†
	Muse Spark 1.3 (xhigh)	10	60	10	0	0	0.70	21.9	22.5	[5–44]	12.9	0.132	2.93
	Nemotron 3 Nano Omni	10	0	0	0	0	0.00	6.8	6.0	[2–12]	34.0	0.072†	0.54†
	GPT-6 Astra (max)	10	**90**	**50**	10	0	1.50	**37.0**	**40.0**	[5–55]	**3.3**	0.834	33.09
	GPT-5.6 Sol (max)	10	70	20	10	**10**	1.10	27.8	26.0	[7–**57**]	4.0	0.272	8.39
	GPT-5.6 Terra (max)	10	50	0	0	0	0.50	18.2	18.0	[10–25]	5.0	0.149	2.91
	GPT-5.6 Luna (max)	10	10	0	0	0	0.10	14.2	16.0	[7–24]	5.0	0.015	0.24
	gpt-oss-120b (high)	10	10	0	0	0	0.10	10.4	8.0	[5–21]	17.0	0.085†	0.91†
	Grok 4.6 (xhigh)	10	30	0	0	0	0.30	15.4	16.0	[5–28]	11.7	0.346	5.68
	GLM-5.3 (max)	10	60	0	0	0	0.60	22.1	22.0	[5–33]	13.4	0.167	3.92
	GLM-5.3 Flash (max)	10	40	0	0	0	0.40	15.6	16.0	[5–27]	14.4	0.024	0.40

A cached system prompt holds the task instructions, a compendium of the game's rules and items, and the memory bank (Sec. 3); each floor adds the act map, the run's notes and the player's state. The model acts only through tool calls, interleaving reasoning with action (Yao et al., 2023), and may return a whole turn's actions in one response (*multi-action*); they run in order up to the first that fails, and each state change is returned as the result of the call that caused it.

Our multi-action implementation asks the model to plan the turn before acting and needs fewer calls: a turn costs one response and one pass over the context, where one action per call pays for both again at every card. This is non-trivial to work with since the action space can shift during a turn; we ask models to internalize these changes to make correct tool calls, and interrupt the action queue if an unpredictable event occurs. MiniStS makes one call per action (Bateni & Whitehead, 2024); Orak's agent may return several actions but spends three calls on each decision (reflection, planning, action), shows only the state they leave, re-plans only once they are spent, and leaves every mid-fight card interaction selection to its rule bot (Park et al., 2026); AgenticSTS submits combat as a plan and rebuilds the prompt for every decision by retrieval (Cheng et al., 2026). Table 2 compares the designs and measures ours against its variants and Orak on the same models and seeds. Ours sees what a

Table 2: **Comparison of Slay the Spire harness designs.** *Actions/call*: how many game actions one model response may issue; *Effects*: whether each action's result is returned to the model; *Context*: what the model sees of earlier decisions; *Choices*: what resolves card choices opened mid-fight (e.g. select a card to put in your hand, on top of your draw pile, or upgrade). Measured rows are paired with our harness on the same models and seeds without the reskin or a cross-run memory bank for consistency with Orak which does not support either. Not run here: MiniStS (Bateni & Whitehead, 2024) (a reimplementation; one action per call, one fight) and AgenticSTS (Cheng et al., 2026) (the sequel; combat as a plan, a prompt rebuilt by retrieval, routine choices to a fast model tier). Calls are per floor on floors both arms played; cost is the run's API cost per floor it reached, priced as in Table 1 (cached input at its own rate), so the Orak row is comparable with the rest. †Orak keeps no per-floor record: calls and cost per floor reached over whole runs, not floor-matched, and over the 40 of its 51 runs with a usage record, since a run Orak's harness ended records none; a run it ended counts as a loss at the floor reached; its rule bot plays non-combat floors with no model call. Full table: Table 26.

						Per floor				Tool errors
Harness	Model plays	Actions/call	Effects	Context	Choices	Calls	Cost ($)	Acts	Floor	/100 calls
Ours, multiaction	whole game	a turn	per action	floor	model	8.9	0.057	0.20	13.1	0.8
single action	whole game	one	per action	floor	model	+3.0	+23%	−0.07	−1.5	+0.4
single action, by index	whole game	one	per action	floor	model	+2.9	+24%	+0.00	−0.2	+0.1
continuous	whole game	a turn	per action	run	model	−0.3	+22%	−0.05	−1.0	+1.2
Orak (Park et al., 2026)	combat, card rewards	a turn, 3 calls	end state	reflection	rule bot	+3.2†	−33%†	−0.16	−4.7	–

human player sees, without strategic steering. It plans a route each act, forecasts at the end of each floor its chances of surviving the act and winning the run, and may predict its HP when it ends a turn. Reasoning is kept across turns wherever the provider allows, and no response reached the output ceiling (Appx. C.2).

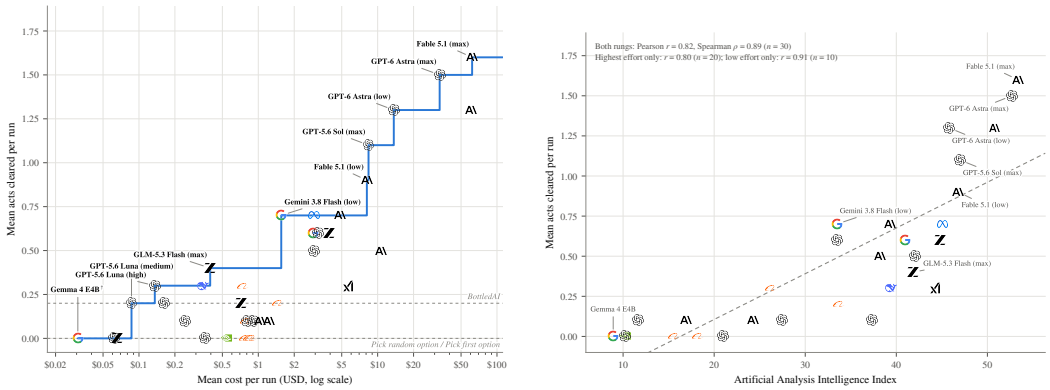
Notes and memory. After each floor the model may write, overwrite or delete free-form *run notes* under keys of its choosing. After each run it consolidates them into its persistent *memory bank* one operation at a time (save, delete or finish); the bank enters every later run's system prompt, so a read-write cohort is a chain played in seed order. We conduct memory ablation experiments to investigate the utility of this schema (Appx. C.3).

Knowledge reskin. Unless stated otherwise every run is reskinned: each game-specific name the model can see is replaced by a name from a hand-written internally-consistent mapping themed around tabletop RPGs. We do this so we can evaluate the model's ability to reason about the game, ablating as much prior information about rules of thumb for how to play StS which it may have seen in pretraining as possible. The model's own text is not renamed, so a real game name it writes or thinks is evidence of recognition of the base game. We conduct ablation experiments to evaluate the effect of this thematic renaming compared to the original or a randomized rename vocabulary (Appx. C.4).

Models and reasoning settings. We evaluate hosted models from Anthropic, DeepSeek, Google, Meta, OpenAI, xAI and Z.ai, and open-weight Qwen, Nemotron and gpt-oss models on our own hardware. Table 1 runs each at its highest reasoning setting (grey, beside its name), resolved to the nearest level the provider supports at or above the request (Appx. C.5).

Evaluation protocol. Every main result is on the Ironclad at Ascension 20, the hardest difficulty, on the same ten seeds—freshly minted both to prevent hyperparameter optimization against model/seed pairs used during preliminary experiments, and so that model providers cannot have trained on our earlier results on these seeds (Sainz et al., 2023). A *cohort* is one model under one configuration on these seeds; each table cell is one cohort, validated by the analysis code (Appx. C.6). We score a run by the acts it clears (Act 4 ends with the Heart) and also report the floor it reaches. A run clears Acts 1 and 2 by beating their (randomly selected) bosses; Act 3 by beating two random bosses; and Act 4 by killing the Corrupt Heart. The *act k win rate* is the share of runs that cleared Act k; mean *acts cleared* (0–4), is our headline summary. *Floor* is the final floor reached. A frontier cohort can cost hundreds of dollars (Table 31), so ten seeds is a budget decision, too few to reliably order neighbouring models but sufficient for broader trends.

Baselines and cost. Three non-LLM policies play the same seeds without seeing names: *pick first option* and *pick random option* bracket the models from below; *BottledAI* (Bottled AI contributors, 2022) is a hand-written symbolic player (Appx. C.9). Human references at the same setting, scored



(a) Acts cleared against cost per run

(b) Acts cleared against the Intelligence Index

Figure 2: **Mean acts cleared per run** for the LLM agents of Table 1. (a) Against mean dollar cost per run (log scale), across reasoning-effort levels. Only points on the Pareto front (blue step line) are named; see Fig. 19 for other names. Dashed lines: the non-LLM baselines (no cost). (b) Against the Artificial Analysis Intelligence Index, at their highest effort and at low effort (*low*). Dashed line: least-squares fit. Only models on the cost Pareto front of (a) are named. For correlations with each Artificial Analysis evaluation, see Table 14.

alike, are the run histories of seven expert streamers (four in Table 1, three record sets in Table 5) and 206,082 finished runs of the game's telemetry, released by Mega Crit. Human games are not matched cohorts against the LLM seeds (Appendices C.10 and D).

4 Results

Performance and cost. Our benchmark results are shown in Table 1. The strongest models are Claude Fable 5.1, GPT-6 Astra, Claude Opus 5 and GPT-5.6 Sol, and only Sol kills the Heart, once; they are also among the most expensive, per run and per floor (Fig. 2a). Mid-tier models beat Bottled AI but not the average human, and four models never clear an act, no better than the random-choice policy. Performance tracks the Artificial Analysis Intelligence Index (Fig. 2b), but reasoning spent does not buy play: GPT-6 Astra thinks least per floor (3.3k tokens) and clears 1.50 acts, while Nemotron 3 Nano Omni thinks most (34.0k) and clears none.

Comparison with human players. The strongest models clear more acts than the average Ascension 20 Ironclad run in Mega Crit's telemetry (0.73) and than players with over 1000 hours (0.92), but far fewer than the four streamers (2.5–3.1 acts). The gap opens as the run lengthens (Table 1 and Fig. 4): the three strongest models win 90% of their Act 1 runs, as the streamers do (85–92%), but 40–50% of Act 2 against 65–78%, and 0–20% of Act 3 against 56–71%. Consequently, the models have a drastically lower win rate for the full game (Act 4) at 0–10% against 46–65%. These results are a demonstration of the horizon gap of Chen et al. (2026), exhibited within a single task.

How the models spend a run. The models play Act 1 cautiously. They fight 0.9–1.4 elites, against 2.0 for the telemetry population and 2.6 for the streamers, rest at 57–75% of rest sites against 23% and 9%, and rest at 31% of the rest sites they enter with at least 70% HP, where the population does so at 2% and the streamers never, upgrading a card instead, a permanent boon for the rest of the game. At the Act 1 boss the top four models hold a median 90% of their maximum HP, the streamers 53% (Fig. 3; Appx. E). Since the transition to the next act restores 75% of missing HP, HP carried to the end of an act keeps only a quarter of its value; the scarce resource is *floors* in which to outgrow enemies that scale as the game progresses. Human runs that play Act 1 as the models do clear Act 2 in 15% of cases, against 38% for runs that fight more elites and rest less (Table 10), and in the models' own runs upgrading rather than resting predicts clearing Act 2 ($r = 0.56$, 49 runs). Experience does not change this: over a chain, rest and elite rates move no differently from no-memory runs; the banks advise caution seven times as often as risk; and the four strongest models record the end-of-act

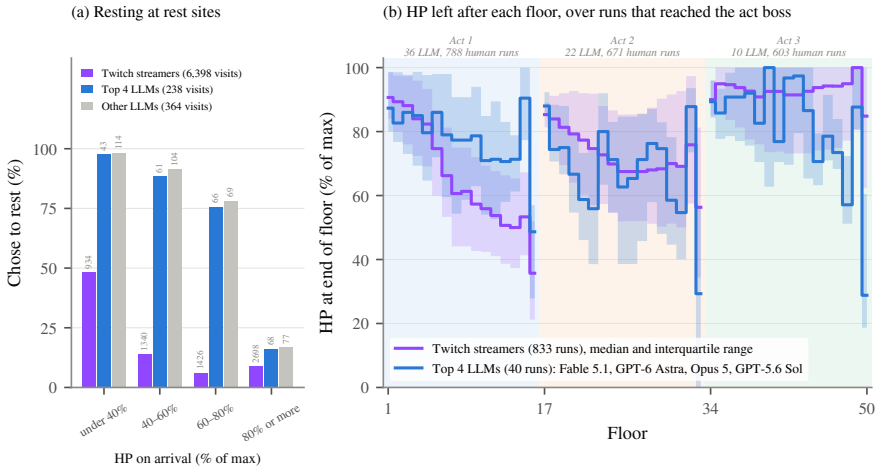


Figure 3: **Do models use HP as a resource?** (a) Rest sites can be used to heal or upgrade a card; both top (blue) and other (grey) tier LLMs nearly always rest if they are missing any HP, whereas Twitch streamers (magenta) favour upgrading much more heavily. Bars are labelled with their number of visits. (b) Relative HP left at the end of each floor. Each act is drawn over the runs that reached the act boss (runs that die to the boss count as 0 HP for the boss floor); run counts are above each act. Lines: medians; bands: interquartile range. Streamers spend their HP in Act 1 to get stronger, reducing the amount of damage they take in future floors as a result.

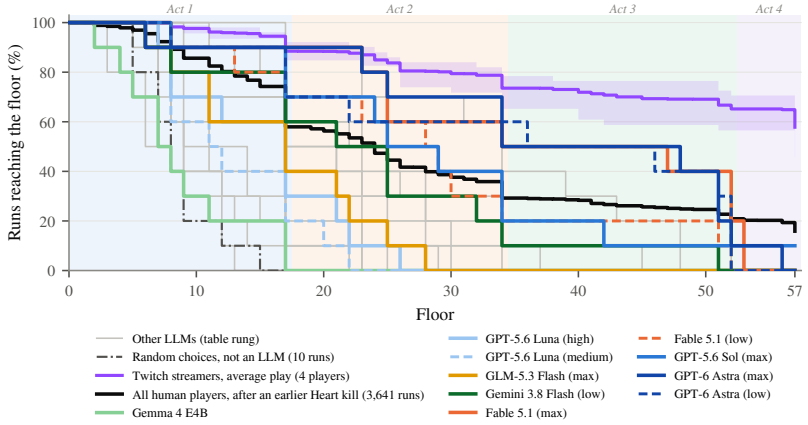


Figure 4: **Survival over a run.** The share of each agent's runs that reached each floor. Every LLM agent in Table 1 with all 10 seeds is shown; models on the cost Pareto front (Fig. 2a) are coloured in shades of one hue per developer, darker for the model that clears more acts (dashed for a model's lower effort level), the rest grey. Floor 57 is past the Heart; a win at the Act 3 boss without the keys stops at floor 52. Dark grey dash-dot: the random-choice policy. Purple: the mean of four Twitch streamers (band spans the four, weighted equally). Black: the 3,641 A20 Ironclad telemetry runs that follow an earlier A20 Heart kill in the same session chain (Appx. D). Shading marks the acts.

heal correctly yet still rest in 21 of the 32 late-act decisions that weigh it. The route explains only about a fifth of the gap to the streamers; the rest is how the fights are played.

Strong models fight better. What separates the models is neither the route (Appx. E) nor, much, the deck. Agreement with human card picks barely rises with model strength (Spearman $\rho = 0.16$), no model builds its deck for the act boss the map shows, and the one deck-building signal is taking area-damage cards (46% of Act 1 offers for the top four models, 20% for the bottom tier; $\rho = 0.76$). The fights separate them. The top four take 8.2 damage per Act 1 hallway fight, the bottom tier 14.0

Table 3: Real game names against our themed reskin, for models with all 10 seeds under both; Δ is themed minus real. Control mappings are in Table 23.

	Mean acts cleared			Mean floor reached		
Agent	Real	Reskin	Δ	Real	Reskin	Δ
Qwen3.8 27B (xhigh)	0.20	0.20	0.00	14.0	14.9	+0.9
Qwen3.6 35B-A3B (xhigh)	0.00	0.00	0.00	9.7	10.2	+0.5
Qwen3.5 122B-A10B (xhigh)	0.10	0.00	−0.10	10.9	9.8	−1.1
Claude Haiku 4.5 (12k)	0.10	0.10	0.00	11.4	11.9	+0.5
DeepSeek V4.1 Flash (max)	0.80	0.30	−0.50	22.0	16.9	−5.1
Gemini 3.8 Flash (low)	0.50	0.70	+0.20	20.2	22.6	+2.4
Nemotron 3 Nano Omni	0.00	0.00	0.00	5.3	6.8	+1.5
GPT-5.6 Luna (max)	0.60	0.10	−0.50	18.5	14.2	−4.3
gpt-oss-120b (high)	0.00	0.10	+0.10	10.4	10.4	0.0
GLM-5.3 (max)	0.70	0.60	−0.10	24.4	22.1	−2.3
GLM-5.3 Flash (max)	0.70	0.40	−0.30	23.1	15.6	−7.5
Mean (11 models)	0.34	0.23	−0.11	15.4	14.1	−1.3

($\rho = -0.87$, 19 models; Table 11). Every tier spends the same energy and blocks the same share of incoming damage; the strong models turn the energy into damage (7.5 against 5.8 per energy spent on attacks, $\rho = 0.91$), rarely block on a turn when no enemy attacks (6% against 31%) and never miss a lethal play in hand (0% against 8.7%), so they kill a turn sooner and are hit less. The difference holds on the same encounter of the same seed, including the first six floors, before the decks diverge, and about three quarters of the top–bottom gap in Act 1 boss wins (94% against 19%) is not explained by the state in which the models arrive.

Relation to public LLM benchmarks. Across 30 model and effort points, mean acts cleared tracks the Artificial Analysis Intelligence Index closely (Pearson $r = 0.82$, Spearman $\rho = 0.89$; Fig. 2b), but the public evaluations behind it, among them GPQA Diamond (Rein et al., 2024) and SciCode (Tian et al., 2024), are strongly correlated with each other too (Burnell et al., 2023; Ruan et al., 2024). Among them SpireBench is one of the least substitutable (Table 14 in Appx. I): the other evaluations predict 50% of its variance for a held-out model, against 85% for Humanity's Last Exam (Phan et al., 2025), consistent with lmgame-Bench, where a game probes a blend of capabilities that other benchmarks test in isolation (Hu et al., 2026). Of the agentic evaluations, only Terminal-Bench 4.0 adds information beyond knowledge and reasoning scores (Table 17).

The harness moves scores as much as neighbouring models differ. On the same models and seeds (Table 2), Orak's harness, whose rule bot plays every non-combat floor, reaches 4.7 fewer floors than ours and clears 0.16 fewer acts, a gap as large as that between many neighbouring models, and it ended 18 of its 142 launches itself (Table 27). The comparison uses six mostly weaker models, which clear 0.20 acts on those seeds with our harness and 0.04 with Orak's; its effect on frontier models is untested. Orak spends 86% fewer tokens per floor but costs only 33% less: most of our input is cached. One action per response needs 3.0 more calls per floor, costs 23% more and loses 0.07 acts, a loss that disappears when choices are also made by number (+0.00 acts), so cost is the robust advantage of multi-action. One conversation for the whole run costs 22% more per floor, loses 0.05 acts and raises tool-call errors from 0.8 to 2.0 per 100 calls (Liu et al., 2024a; Sinha et al., 2026).

Memory does not compound. Agents that accumulate experience across episodes are reported to improve with it (Shinn et al., 2023; Wang et al., 2024a; Majumder et al., 2023; Zhao et al., 2024; Wang et al., 2024b; Suzgun et al., 2026); ours do not, nor did the episodic store of Cheng et al. (2026) in StS2 (6/10 wins with and without it). Consolidating a run's notes into a memory bank or writing a compact summary has no impact on the number of acts won (see Fig. 16 and Table 24). Carrying the notes from during the run forward verbatim without a memory consolidation or compaction phase does slightly better (+0.06 acts): the rewriting adds nothing, much as iterative rewriting of an agent's context erodes its detail (Zhang et al., 2026). Without an external check on what is stored, models do not reliably correct their own reasoning (Huang et al., 2024), and agents propagate the errors they store (Xiong et al., 2025). A run gives one delayed, noisy outcome for hundreds of decisions, so nothing verifies a consolidated memory, and the bank grows run on run (Fig. 17). Moreover, in the final banks 17% of memories are false, mostly generalizations from a single run, and models whose banks hold more errors of their own clear fewer acts (Spearman $\rho = -0.67$, permutation $p = 0.002$, $n = 20$; Table 19). Nothing prunes them: over the 200 runs the models opt to delete only 30 memories.

Prior knowledge is part of the score. Renaming or altering a familiar task has exposed reliance on memorized content (Valmeekam et al., 2023; Wu et al., 2024b; McCoy et al., 2024), while capture-

the-flag agents are largely robust to renaming (Honarvar et al., 2026). Our result lies between. Hiding the real names costs 0.14 acts and 11 points of Act 1 win rate, pooled over 12 model configurations (11 models) and every paired seed (Table 23; per model with all 10 seeds, Table 3), concentrated in a few: DeepSeek V4.1 Flash falls from 0.80 acts to 0.30 and GPT-5.6 Luna from 0.60 to 0.10 (Table 29). Nonsense names cost least (0.01 acts), so unfamiliar words are not the difficulty; the two permuted mappings, which give each entity the name (real or themed) of another, cost about as much as the themed one (0.10 and 0.12). A familiar name pulls a model towards what it usually means although the prompt states the rules (McCoy et al., 2024): the signature of knowledge acquired in training, which contamination studies seek in static benchmarks (Golchin & Surdeanu, 2024) and we show by intervention. The reskin hides names, not the game: in 125 of the 190 themed runs of Table 1 with full logs (Gemma 4 E4B has none), from 18 of 19 models, the model recognizes Slay the Spire or a real entity the reskin replaced, yet recognition does not buy back the cost (Spearman $\rho = 0.18$, permutation $p = 0.62$; Table 20).

Models forecast the next turn, not the consequences for the rest of the run. Larger models are reported to be well calibrated on questions they can answer (Kadavath et al., 2022), while confidence stated in words tends to be overconfident (Xiong et al., 2024); here the horizon decides which holds. Models that forecast the next turn's HP well clear more acts (Spearman $\rho = -0.84$ between error and acts; Table 13). This forecast measures the fight-reading skill above: on turns where the intents shown determine the next HP exactly, the bottom tier still mispredicts it 31% of the time and the top four 0.5%. Forecasts of surviving the act are poorly calibrated: the four models that never clear an act give themselves a 55–76% chance, about as much as the four strongest (59–74%), whose forecasts are followed by surviving the act 63–76% of the time (Fig. 13 in Appx. H), and they barely sharpen as the act nears its end (Fig. 14). Card-reward policy, failures in the reasoning traces and tool-call errors are analysed in Appx. I (Tables 15, 18 and 21).

5 Discussion

Our findings share one shape—the models fail to calibrate risk correctly over the long-term. Where the outcome is a turn or a fight away, the strongest models are competent: they win the first act as often as streamers do, win almost every ordinary fight and forecast the next turn well, though they still lose more HP in each fight than humans do. At the time horizon of an act or a run away, they fall behind: the gap to the streamers opens from Act 2, boss fights decide where each model stops, forecasts of surviving an act barely sharpen; moreover, what a model writes down for later runs does not compound. Our findings are indicative of the horizon gap of Chen et al. (2026), but within one task whose time scales share a single pool of resources. Measured as a time horizon (Kwa et al., 2026), the fall is a step: every model succeeds on at least 87% of its turns and falls below 50% between a fight and a run, for most models already at the act (Table 16). The difficulty is not how long a model can keep going: it is fighting well enough to afford risk, and spending early floors on growth whose payoff comes an act later, which no model learns from its own runs. Any agent planning over a long, stochastic task with one delayed outcome needs memories checked against outcomes and forgotten when wrong, forecasts that sharpen as the stakes approach, and a harness that plays the whole task (Sec. 5.1).

5.1 Limitations

What plausibly transfers. Four findings rest on mechanisms that prior work finds outside games. *The harness is part of the score*, as for other agents (Kapoor et al., 2025; Yang et al., 2024). *Unverified memory does not compound* wherever the outcome is delayed and noisy over many decisions, as in most open-ended deployments; where a verifier exists, such as tests for code, the reported gains should hold. *Long contexts cost performance* (Liu et al., 2024a; Sinha et al., 2026). *Calibration degrades with the horizon* in any task whose outcomes resolve at several distances.

What does not, or may transfer. *The ordering of models* is measured on one character in a turn-based game played from parsed text; other characters, genres, real-time control or perception would weight the decisions differently. *The size of the reskin cost* depends on how much of this game each model saw in training, and the reskin hides names but not familiarity with deck-builders, so it is a lower bound. *The human comparison* is not seed-matched. Ten seeds per cohort cannot order neighbouring models, and providers can change the model behind an identifier without notice.

AI USE STATEMENT

In this work, we used generative AI tools to assist with data analysis. We have not used generative AI tools for designing the research methodology. Additionally, we used generative AI tools for implementing the harness, creating figures, and drafting sections of the text. We have reviewed all AI-assisted work. The harness was developed iteratively with a human-in-the-loop to identify errors and find new edge cases to probe in a process that took numerous rounds of refinement. Data analysis, figure creation, and initial drafts of the text were directed with detailed prompts and iterated on with AI before manual refinement and final verification by the authors. We take responsibility for the final content of this work, including text, claims or artefacts produced with the aid of generative AI.

ETHICS STATEMENT

We used identifiable human data as part of our comparisons in this work. Data from Twitch streamers is publicly available. Privately held data was obtained and shared with permission.

REPRODUCIBILITY STATEMENT

Upon acceptance, we will make our harness source code and run logs publicly available.

REFERENCES

Johan Andersson, Mikolaj Trojanowski, and Alberto Alvarez. Are you lucky or skilled? An analysis of elements of randomness in Slay the Spire. In *IEEE Conference on Games (CoG)*, 2023. doi:10.1109/CoG57401.2023.10333231.

Axel Backlund and Lukas Petersson. Vending-Bench: A benchmark for long-term coherence of autonomous agents. *arXiv preprint arXiv:2502.15840*, 2025.

Bahar Bateni and Jim Whitehead. Language-driven play: Large language models as game-playing agents in Slay the Spire. In *Proceedings of the 19th International Conference on the Foundations of Digital Games (FDG)*, pp. 1–10. ACM, 2024. doi:10.1145/3649921.3650013.

Mahsa Bazzaz and Seth Cooper. Analysis of uncertainty in procedural maps in Slay the Spire. In *Proceedings of the 20th International Conference on the Foundations of Digital Games (FDG)*, pp. 1–10. ACM, 2025. doi:10.1145/3723498.3723846. arXiv:2504.03918.

Bottled AI contributors. Bottled AI: A customizable bot for Slay the Spire. `https://github.com/xaved88/bottled_ai`, 2022.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi:10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.

Sarfraz Brohi, Qurat-ul-ain Mastoi, N. Z. Jhanjhi, and Thulasyammal Ramiah Pillai. A research landscape of agentic AI and large language models: Applications, challenges and future directions. *Algorithms*, 18(8):499, 2025. doi:10.3390/a18080499.

Ryan Burnell, Han Hao, Andrew R. A. Conway, and José Hernández-Orallo. Revealing the structure of language model capabilities, 2023. URL `https://arxiv.org/abs/2306.10062`.

Mingguang Chen, Licheng Wang, and Bo Qu. The horizon gap: Planning, memory, execution, training, and evaluation for long-horizon LLM agents. *arXiv preprint arXiv:2608.06663*, 2026.

Xiangchen Cheng, Yunwei Jiang, Jianwen Sun, Zizhen Li, Chuanhao Li, Xiangcheng Cao, Yihao Liu, Fanrui Zhang, Li Jin, and Kaipeng Zhang. AgenticSTS: A bounded-memory testbed for long-horizon LLM agents, 2026. URL `https://arxiv.org/abs/2607.02255`.

Coder. BalatroBench: Benchmark LLMs' strategic performance in Balatro. `https://github.com/coder/balatrobench`, leaderboard at `https://balatrobench.com`, 2026.

Anthony Costarelli, Mat Allen, Roman Hauksson, Grace Sodunke, Suhas Hariharan, Carlson Cheng, Wenjie Li, Joshua Clymer, and Arjun Yadav. GameBench: Evaluating strategic reasoning abilities of LLM agents, 2024. URL `https://arxiv.org/abs/2406.06613`.

Rachit Dubey, Pulkit Agrawal, Deepak Pathak, Thomas L. Griffiths, and Alexei A. Efros. Investigating human priors for playing video games. In *International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1349–1357. PMLR, 2018.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. First return, then explore. *Nature*, 590(7847):580–586, 2021. doi:10.1038/s41586-020-03157-9.

Ronald A. Fisher. *The Design of Experiments*. Oliver and Boyd, Edinburgh, 1935.

ForgottenArbiter et al. CommunicationMod: A mod that adds external-process communication to Slay the Spire. `https://github.com/ForgottenArbiter/CommunicationMod`, 2019.

Pablo García-Sánchez, Alberto Tonda, Giovanni Squillero, Antonio Mora, and Juan J. Merelo. Evolutionary deckbuilding in HearthStone. In *IEEE Conference on Computational Intelligence and Games (CIG)*, pp. 1–8, 2016. doi:10.1109/CIG.2016.7860426.

Shahriar Golchin and Mihai Surdeanu. Time travel in LLMs: Tracing data contamination in large language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Ken Gu, Advait Bhat, Mike A. Merrill, Robert West, Xin Liu, Daniel McDuff, and Tim Althoff. SynthWorlds: Controlled parallel worlds for disentangling reasoning and knowledge in language models. In *International Conference on Learning Representations (ICLR)*, 2026. URL `https://openreview.net/forum?id=46AQ4qaWqQ`. arXiv:2510.24427.

Danijar Hafner. Benchmarking the spectrum of agent capabilities. In *International Conference on Learning Representations (ICLR)*, 2022. URL `https://openreview.net/forum?id=1W0z96MFEoH`.

Shahin Honarvar, Amber Gorzynski, James Lee-Jones, Harry Coppock, Marek Rei, Joseph Ryan, and Alastair F. Donaldson. Capture the flags: Family-based evaluation of agentic LLMs via semantics-preserving transformations, 2026. URL `https://arxiv.org/abs/2602.05523`.

Lanxiang Hu, Mingjia Huo, Yuxuan Zhang, Haoyang Yu, Eric P. Xing, Ion Stoica, Tajana Rosing, Haojian Jin, and Hao Zhang. lmgame-bench: How good are LLMs at playing games? In *International Conference on Learning Representations (ICLR)*, 2026. URL `https://openreview.net/forum?id=qeziG97WUZ`. arXiv:2505.15146.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *International Conference on Learning Representations (ICLR)*, 2024.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations (ICLR)*, 2024.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Sayash Kapoor, Benedikt Stroebl, Zachary S. Siegel, Nitya Nadgir, and Arvind Narayanan. AI agents that matter. *Transactions on Machine Learning Research*, 2025.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. VisualWebArena: Evaluating multimodal agents on realistic visual web tasks. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Heinrich Küttler, Nantas Nardelli, Alexander H. Miller, Roberta Raileanu, Marco Selvatici, Edward Grefenstette, and Tim Rocktäschel. The NetHack learning environment. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Thomas Kwa, Ben West, Joel Becker, Amy Deng, Katharyn Garcia, Max Hasin, Sami Jawhar, Megan Kinniment, Nate Rush, Sydney Von Arx, Ryan Bloom, Thomas Broadley, Haoxing Du, Brian Goodrich, Nikola Jurkovic, Luke Harold Miles, Seraphina Nix, Tao Lin, Chris Painter, Neev Parikh, David Rein, Lucas Jun Koba Sato, Hjalmar Wijk, Daniel M. Ziegler, Elizabeth Barnes, and Lawrence Chan. Measuring AI ability to complete long software tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2026.

Haohang Li, Yupeng Cao, Yangyang Yu, Shashidhar Reddy Javaji, Zhiyang Deng, Yueru He, Yuechen Jiang, Zining Zhu, K.P. Subbalakshmi, Jimin Huang, Lingfei Qian, Xueqing Peng, Jordan W. Suchow, and Qianqian Xie. INVESTORBENCH: A benchmark for financial decision-making tasks with LLM-based agent. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL), Volume 1: Long Papers*, pp. 2509–2525, 2025a. doi:10.18653/v1/2025.acl-long.126.

Yuchen Li, Cong Lin, Muhammad Umair Nasir, Philip Bontrager, Jialin Liu, and Julian Togelius. GVGAI-LLM: Evaluating large language model agents with infinite games, 2025b. URL https://arxiv.org/abs/2508.08501.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024a. doi:10.1162/tacl_a_00638.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. AgentBench: Evaluating LLMs as agents. In *International Conference on Learning Representations (ICLR)*, 2024b.

Bodhisattwa Prasad Majumder, Bhavana Dalvi Mishra, Peter Jansen, Oyvind Tafjord, Niket Tandon, Li Zhang, Chris Callison-Burch, and Peter Clark. CLIN: A continually learning language agent for rapid task adaptation and generalization, 2023. URL https://arxiv.org/abs/2310.10134.

R. Thomas McCoy, Shunyu Yao, Dan Friedman, Matthew D. Hardy, and Thomas L. Griffiths. Embers of autoregression show how large language models are shaped by the problem they are trained to solve. *Proceedings of the National Academy of Sciences*, 121(41):e2322420121, 2024. doi:10.1073/pnas.2322420121.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: A benchmark for general AI assistants. In *International Conference on Learning Representations (ICLR)*, 2024.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015. doi:10.1038/nature14236.

Allan H. Murphy. A new vector partition of the probability score. *Journal of Applied Meteorology*, 12(4):595–600, 1973. doi:10.1175/1520-0450(1973)012<0595:ANVPOT>2.0.CO;2.

SangGyu Nam and Pavinee Rerkjirattikal. Exploring the impact of randomness in roguelike deckbuilding games: A case study of Slay the Spire. In *Integrated Uncertainty in Knowledge Modelling and Decision Making (IUKM)*, Lecture Notes in Computer Science. Springer, 2023. doi:10.1007/978-3-031-46775-2_9.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Davide Paglieri, Bartłomiej Cupiał, Sam Coward, Ulyana Piterbarg, Maciej Wołczyk, Akbir Khan, Eduardo Pignatelli, Łukasz Kuciński, Lerrel Pinto, Rob Fergus, Jakob Foerster, Jack Parker-Holder, and Tim Rocktäschel. BALROG: Benchmarking agentic LLM and VLM reasoning on games. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2411.13543.

Dongmin Park, Minkyu Kim, Beongjun Choi, Junhyuck Kim, Keon Lee, Jonghyun Lee, Inkyu Park, Byeong-Uk Lee, Jaeyoung Hwang, Jaewoo Ahn, Ameya S. Mahabaleshwarkar, Bilal Kartal, Pritam Biswas, Yoshi Suhara, Kangwook Lee, and Jaewoong Cho. Orak: A foundational benchmark for training and evaluating LLM agents on diverse video games. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2506.03610.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *ACM Symposium on User Interface Software and Technology (UIST)*, 2023.

Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The Berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48371–48392, 2025.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof Q&A benchmark. In *Conference on Language Modeling (COLM)*, 2024. arXiv:2311.12022.

Yangjun Ruan, Chris J. Maddison, and Tatsunori Hashimoto. Observational scaling laws and the predictability of language model performance. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10776–10787, 2023. doi:10.18653/v1/2023.findings-emnlp.722.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. doi:10.1038/nature16961.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018. doi:10.1126/science.aar6404.

Akshit Sinha, Arvindh Arun, Shashwat Goel, Steffen Staab, and Jonas Geiping. The illusion of diminishing returns: Measuring long horizon execution in LLMs. In *International Conference on Learning Representations (ICLR)*, 2026.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *International Conference on Learning Representations (ICLR)*, 2025.

Stanford Institute for Human-Centered Artificial Intelligence (HAI). The 2026 AI index report. `https://hai.stanford.edu/ai-index/2026-ai-index-report`, 2026.

Mirac Suzgun, Mert Yuksekgonul, Federico Bianchi, Dan Jurafsky, and James Zou. Dynamic cheatsheet: Test-time learning with adaptive memory. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7080–7106, 2026. doi:10.18653/v1/2026.eacl-long.333.

Minyang Tian, Luyu Gao, Shizhuo Dylan Zhang, Xinan Chen, Cunwei Fan, Xuefei Guo, Roland Haas, Pan Ji, Kittithat Krongchon, Yao Li, et al. SciCode: A research coding benchmark curated by scientists. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Mikolaj Trojanowski and Johan Andersson. Are you lucky or skilled in Slay The Spire?: An analysis of randomness. Bachelor's thesis, Malmö University, 2021. URL https://urn.kb.se/resolve?urn=urn:nbn:se:mau:diva-43090.

Karthik Valmeekam, Matthew Marquez, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. PlanBench: An extensible benchmark for evaluating large language models on planning and reasoning about change. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, volume 36, pp. 38975–38987, 2023. doi:10.52202/075280-1693.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research (TMLR)*, 2024a. URL https://openreview.net/forum?id=ehfRiF0R3a.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory, 2024b. URL https://arxiv.org/abs/2409.07429.

Yue Wu, Xuan Tang, Tom M. Mitchell, and Yuanzhi Li. SmartPlay: A benchmark for LLMs as intelligent agents. In *International Conference on Learning Representations (ICLR)*, 2024a.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL), Volume 1: Long Papers*, pp. 1819–1862, 2024b. doi:10.18653/v1/2024.naacl-long.102.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can LLMs express their uncertainty? An empirical evaluation of confidence elicitation in LLMs. In *International Conference on Learning Representations (ICLR)*, 2024.

Zidi Xiong, Yuping Lin, Wenya Xie, Pengfei He, Zirui Liu, Jiliang Tang, Himabindu Lakkaraju, and Zhen Xiang. How memory management impacts LLM agents: An empirical study of experience-following behavior. *arXiv preprint arXiv:2505.16067*, 2025.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-agent: Agent-computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024.

Asaf Yehudai, Lilach Eden, Alan Li, Guy Uziel, Yilun Zhao, Roy Bar-Haim, Arman Cohan, and Michal Shmueli-Scheuer. Survey on evaluation of LLM-based agents. *arXiv preprint arXiv:2503.16416*, 2025.

Alex L. Zhang, Thomas L. Griffiths, Karthik R. Narasimhan, and Ofir Press. VideoGameBench: Can vision-language models complete popular video games?, 2025. URL https://arxiv.org/abs/2505.18134.

Qizheng Zhang, Changran Hu, Shubhangi Upasani, Boyuan Ma, Fenglu Hong, Vamsidhar Kamanuru, Jay Rainton, Chen Wu, Mengmeng Ji, Hanchen Li, Urmish Thakker, James Y. Zou, and Kunle Olukotun. Agentic context engineering: Evolving contexts for self-improving language models. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2510.04618.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: LLM agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pp. 19632–19642, 2024. doi:10.1609/aaai.v38i17.29936.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2024.

APPENDICES

A The harness, step by step **17**

B Why Slay the Spire **18**

C Experimental details **19**

C.1 Game interface 19

C.2 Agent harness 19

C.3 Notes and memory 20

C.4 Knowledge reskin 20

C.5 Models and reasoning settings 20

C.6 Seeds and cohort selection 21

C.7 Metrics and statistics 21

C.8 Cost 21

C.9 Non-LLM baselines 22

C.10 Human baselines 22

C.11 Ablations and analyses 22

D Human baselines **24**

E How agents and humans differ where agents die **27**

F Strategy and play quality **32**

G Difficulty and where runs end **34**

H Calibration **36**

I Further analyses **41**

J Ablation results in full **50**

K Cost and reasoning frontiers in full **58**

L Time and cost of a run **61**

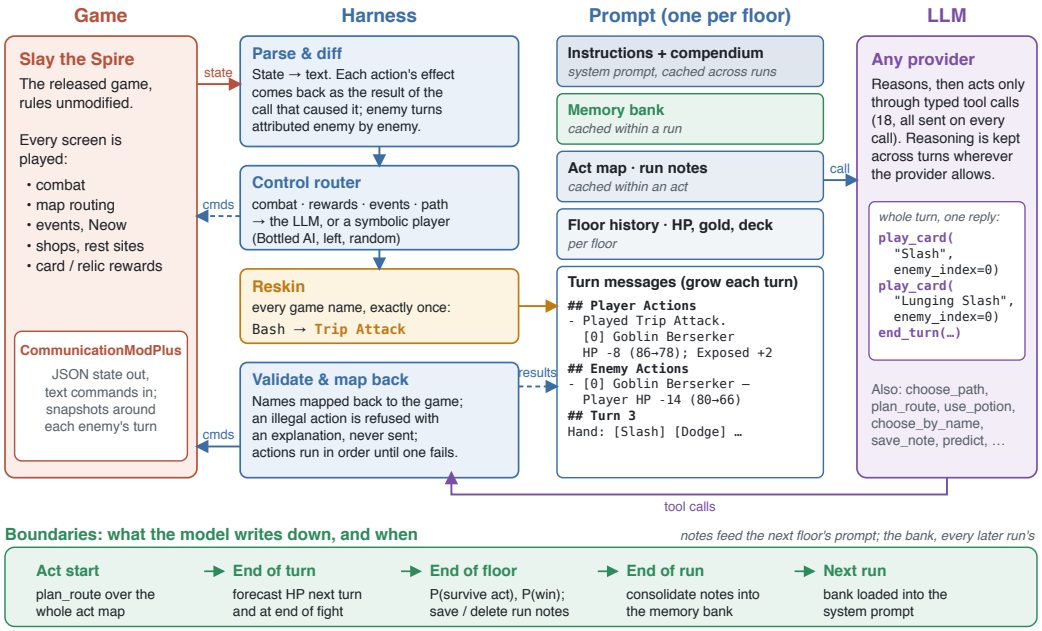


Figure 5: **Our harness.** Game states become text, each action's effect attributed to its tool call; a router sends each part of the game to the model or a symbolic player; in-game names are (optionally) reskinned; tool calls are mapped back and validated. The prompt runs from cached layers to turn messages. Bottom: when the model plans, forecasts and writes notes.

A THE HARNESS, STEP BY STEP

Figure 6 follows the messages exchanged between the game, the harness and the model through the four points at which the harness calls the model: the start of an act, each combat turn, the end of a floor and the end of a run (Sec. 3). Figure 5 gives an overview of its parts.

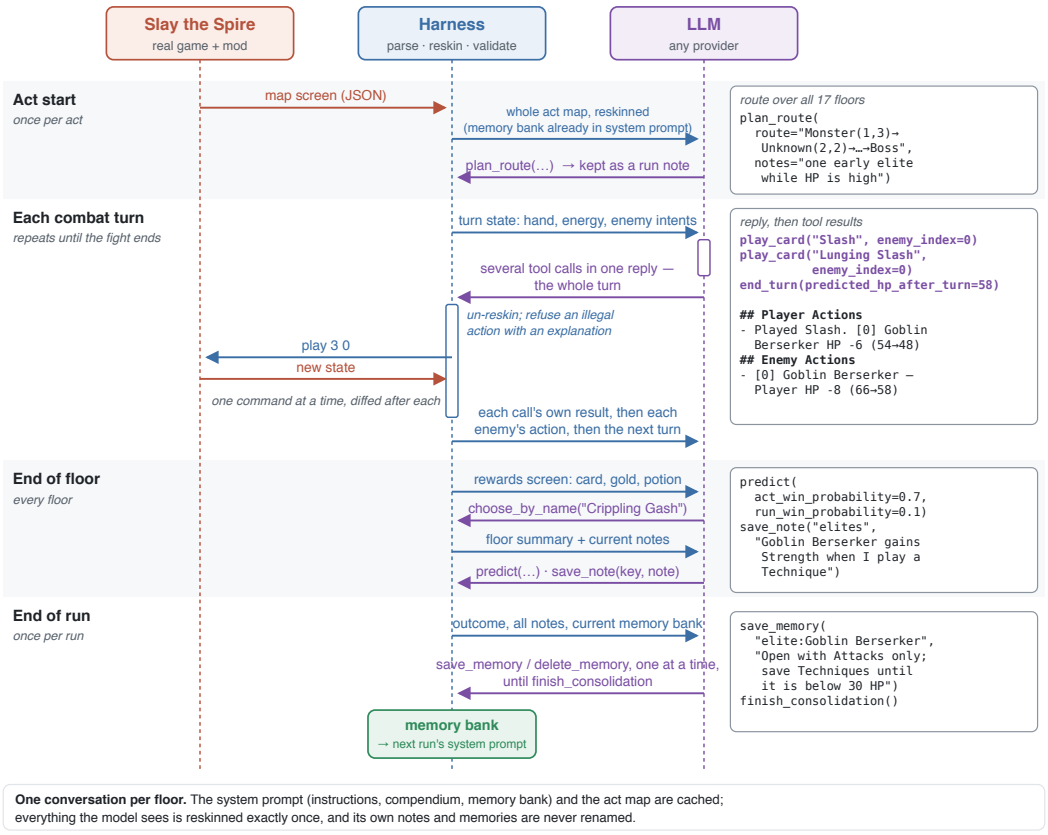


Figure 6: The harness as a sequence of messages, with time running downwards. In combat the model returns a whole turn as several tool calls in one reply; the harness sends them to the game one command at a time, diffs the state after each, and returns each call's own effect as that call's result, followed by each enemy's action and the next turn's state. At the end of each floor the model forecasts its chances and may update its run notes; at the end of the run it consolidates them into the memory bank that the next run's system prompt carries. Code on the right is the shape of what the model sends and receives, with names as it sees them under the reskin.

B WHY SLAY THE SPIRE

Slay the Spire is the game that established the roguelike deck-builder genre, and the properties that make it hard are those that make any long task hard for an agent. **Every choice recurs.** A card added to the deck on the second floor is drawn again in every later fight, so it has to be useful not only against the next enemy but against the act bosses an hour later, and a bad addition cannot simply be undone: removing one usually costs gold, a visit to a shop and a floor that could have gone to something else. The consequences of a decision are therefore spread thinly over hundreds of later ones, which is the credit-assignment problem of long-horizon execution, where small per-step errors compound over the length of a task (Sinha et al., 2026). **An archetype is not a deck.** Players learn named strategies, such as building around Strength or around exhausting cards, but a run offers only the cards and relics that its seed happens to deal, so a plan must be revised as the run unfolds rather than executed as written, and the agent has to judge when to commit to a plan and when to abandon it.

Risk is amortized over the run. HP, gold, potions and the deck form one irreversible pool that carries from floor to floor. Taking an elite fight costs HP now for a relic that pays out over every later fight; resting at a campfire recovers HP at the price of an upgrade. The right choice depends on what the rest of the run will demand, so the agent must price present resources against a future it can only estimate, and how well it does so shows in how well calibrated those estimates are. **The challenges ahead are known in kind but not in detail.** Each act draws its fights from a fixed pool, and the map shows the types of room ahead, so a player who remembers what the act's elites and bosses do can

build the deck to meet them; one who does not meets each as a surprise. Doing well therefore needs memory across runs as well as within one, which is the setting in which memory mechanisms for LLM agents are proposed (Park et al., 2023; Packer et al., 2023; Shinn et al., 2023) but seldom tested over long, stochastic episodes.

These four properties are the parts of the horizon gap (Chen et al., 2026) that short benchmarks cannot reach, and a single run of the game, about an hour of human play, sits at the task lengths on which agent progress is now charted (Kwa et al., 2026). The game also has three practical properties that make it a good testbed. It is turn-based and fully described in text, so what is tested is decision-making rather than perception or reaction time. Its outcomes mix skill with luck in a measured proportion (Trojanowski & Andersson, 2021), and its maps pose genuine route decisions under uncertainty (Bazzaz & Cooper, 2025), so a fixed set of seeds gives a paired, repeatable comparison. And it has a large, well-documented population of human players, from telemetry to expert streamers, against which models can be placed.

C Experimental details

This appendix gives the details of the setup summarized in Sec. 3.

C.1 Game interface

Our fork of CommunicationMod (ForgottenArbiter et al., 2019), `CommunicationModPlus`, adds a file-based channel that survives restarts of the game's Java process, tracking of the three keys that unlock Act 4, and intermediate state snapshots around each enemy's turn, from which the harness attributes damage, block and status changes to the enemy that caused them. Every run loads the same mod list: `BaseMod`, `StSLib`, `CommunicationModPlus`, `rngFix`, `permNeow` (a Neow blessing on every run), `SuperFastMode` (animation speed only), `RelicStats` and `fast_multicast`. None changes the rules a human plays by. Each run executes in an isolated game profile, and every act boss is marked as seen, so each run's seed, and not the profile's history, picks the act bosses.

C.2 Agent harness

Each call carries a cached system prompt in two parts. The first is stable across runs: task instructions and a compendium of the game's rules, keywords, status effects, cards, relics and potions with their effects. The second is stable within a run: the memory bank (Sec. 3). The first user message of a floor adds the act map, the run's notes, a history of the floors played so far and the player's current state (HP, gold, deck, relics, potions and keys). In combat, each turn's message reports what each of the model's actions did, what each enemy did, and the full state at the start of the new turn. Outside combat the message describes the current screen and its legal choices.

The tools are: playing a card or a sequence of cards, ending the turn, using or discarding a potion, choosing an option by index or by name, choosing a map node, planning a route, taking notes, predicting the floor's outcome, and the end-of-run memory operations. The full tool set is sent on every call, so that the cached prefix never changes, and an action illegal on the current screen is refused with an explanation. By default the model may return several actions in one response (*multi-action*), and the harness executes them in order, stopping at the first that fails. At the start of each act the model is asked for a route plan over the whole act map. At the end of each floor it is asked for its chance of surviving the current act and of winning the run, and may reflect and update its notes; when it ends a turn it may also predict its HP at the start of the next turn and at the end of the fight.

Two principles shaped the harness. The model is shown what a human player sees and nothing more: the prompts describe the task on the current screen without steering towards a strategy (they never say, for instance, to check first whether an enemy can be killed this turn), and the system prompt holds only the documentation a player has in the game itself, namely its tutorial, tips and compendium. And the model is asked to do as much as possible per call, to keep cost down. Many combat actions have a deterministic effect, so the model plans the whole turn and returns every action it wants in one response, up to the first whose outcome is random. MiniStS makes one call per action and re-prompts

with the new state (Bateni & Whitehead, 2024). Orak's agent may also return several actions at once, but spends three calls on each decision (reflection, planning, then the actions) and shows the model only the state its actions leave (Park et al., 2026). Multi-action play asks the model to hold the intervening states in mind, which is harder. So that it can still see what each action did, the harness diffs the game state between actions and returns each change as the result of the tool call that caused it, with enemies' turns attributed enemy by enemy. A call that names a card imprecisely is resolved where the intended card is unambiguous, and otherwise answered with a "did you mean" suggestion rather than executed.

Reasoning is kept wherever the provider allows. Claude's signed thinking blocks are returned unchanged; OpenAI's reasoning models and Mistral's reasoning models keep their reasoning server side (through `previous_response_id` and the Conversations API respectively); Gemini's thought signatures are round-tripped; and providers that return reasoning as text (DeepSeek, Z.ai GLM, Qwen, gpt-oss, Nemotron) have it sent back in later turns, stripped only where a provider rejects it. Temperature is left at each provider's default, and a response may use up to 65,536 tokens, reduced to the model's own output limit or a quarter of its context window where either is smaller. No response reached this ceiling: none of the 274,306 model calls in the 1,908 finished runs behind this paper was truncated.

C.3 Notes and memory

Run notes are rendered back to the model as `<note key="...">` tags. The memory ablation (Table 24) varies the mechanism while holding everything else fixed. *Consolidate (xml)* is the default of Sec. 3. *Keep notes* copies each run's notes into the bank verbatim, with no consolidation call, so it keeps persistence but removes the rewriting. *None* discards the notes at the end of each run. In the *compact* arms the model instead writes a single prose summary through a one-argument `compact` tool, overwriting it at the end of every floor and, when consolidating, at the end of every run. Every prompt that asks the model to write something down includes the same *generic* guidance: distil insights rather than log events, and record what went well, what went wrong and what the model wishes it had known. It names nothing specific to the game.

A cohort with a read-write bank is a chain. Seed k starts from the bank that seed $k-1$ left behind, so the seeds are played in order and a cohort's bank is never shared with any other configuration: each bank is partitioned by model, character, the full configuration tag and the seed list.

C.4 Knowledge reskin

The names replaced are those of cards, relics, potions, enemies, events, powers and keywords, the character, and the map's node labels. Names are replaced through placeholders, so that one replacement can never trigger another, and each replacement evokes the original's effect rather than its name. The harness's own text (the system prompt, state descriptions and tool results) is renamed; the model's own text is not, so a real game name in a note is evidence of recognized prior knowledge rather than something to be laundered.

The five mappings of the reskin ablation (Table 23) are all generated in the same format, so that only the mapping file changes between arms. *None* shows the real names. *Themed* is the default. *Random* replaces every word with a pronounceable nonsense word, consistently, so that multiword names keep their shared structure. *Permuted* gives each entity the real name of a different entity of the same kind (cards within their colour and type), which is coherent but actively misleading. *Permuted and themed* applies the same permutation in the themed vocabulary. The permuted mappings keep every rule word that the themed mapping leaves unrenamed, so the arms differ only in the names.

C.5 Models and reasoning settings

Reasoning intensity is set on one scale (`low`, `medium`, `high`, `xhigh`, `max`) and resolved for each provider to the nearest level it supports at or above the one asked for, so a model is never run below its label. Claude Haiku 4.5 takes a thinking budget instead of a level and is run at 12,288 tokens; Nemotron offers no depth control and runs at its default. Test-time compute can matter as much as model size (Snell et al., 2025), so the effort ablation (Table 22) reruns the same configuration at every

other level each model was run at. A model name that the provider points at successive versions is always recorded as the version that answered.

C.6 Seeds and cohort selection

We replaced the seed list the harness had used during preliminary development since the opening states of a seed (the Neow choice, the first map and the first card reward) are identical in every run of it, and some providers' terms permit training on API traffic, which is the route by which static benchmarks become contaminated (Sainz et al., 2023; Golchin & Surdeanu, 2024). New seeds close that route to memorization, though not the possibility that a model has seen the harness's prompt format and learned about the domain.

Runs are recorded in Weights & Biases, and each table cell is built from exactly one cohort, selected and then checked by the analysis code: (i) every run shares one run-name stem, which encodes the model, character, full configuration tag and seed list, so that runs from other configurations or seed lists cannot pool; (ii) only runs that finished and recorded a final floor are used; (iii) a seed played more than once contributes its latest run; (iv) every recorded configuration value other than the seed is identical across the cohort; and (v) for a read-write memory cohort, the runs form one unbroken chain: the first seed starts from an empty bank and each seed starts from the bank the previous one left. When a restart left a stray run newer than the chain it would be spliced into, the chain is rebuilt backwards from the last seed by matching each run's starting memory count to the previous run's final one; this changed one cohort. A failed check stops the analysis rather than being worked around, with one named exception: Gemma 4 E4B's first attempt at its first seed crashed after saving two memories, and the replay of that seed started from them, so its first seed saw two memories written about the same game; it is the only such case.

C.7 Metrics and statistics

Act 1 ends at floor 16, Act 2 at floor 33, and Ascension 20's two Act 3 bosses are at floors 50 and 51. Mean acts cleared rewards reaching the next act and is not moved by how far into an act a run died. Floor is reported as mean, median and range. Per-floor figures divide by the final floor plus one, since Neow counts as a floor. *Thinking tokens* are the reasoning tokens a provider reports; where a self-hosted server does not separate them, the count is apportioned by the reasoning text's share of the response's characters. Paired comparisons across models, such as an ablation against its baseline, use exact sign-flip permutation tests (Fisher, 1935); correlations across models are tested by permutation, and within-run contrasts by exact sign tests.

C.8 Cost

Hosted models are priced at each provider's list price for input, cached input, cache writes and output, with reasoning billed as output. Two conventions apply. OpenAI reports cache reads but not cache writes, so the written share is inferred from the growth of the cached prefix between calls, and can only be underestimated. DeepSeek's prices vary with the time of day; we charge the peak rate, so its figures are upper bounds. Z.ai's GLM models were run on a flat-rate subscription and are priced at the list price of the tokens they used.

Self-hosted models have no bill, so we price their recorded tokens at the headline prices published by Artificial Analysis (retrieved 22 September 2026). These are the model's first-party API price where one exists, otherwise the median across providers. Input, cached input and output are each charged at their listed rate; where no cache-hit price is listed, cached input is charged in full. The per-million-token prices are: gpt-oss-120b \$0.15 / \$0.125 / \$0.59; Qwen3.8 27B \$0.50 / \$0.05 / \$3.00; Qwen3.6 35B-A3B \$0.38 / \$0.38 / \$2.25; Qwen3.5 122B-A10B \$0.40 / \$0.40 / \$3.20; Nemotron 3 Nano Omni \$0.20 / \$0.195 / \$1.09; Gemma 4 E4B \$0.02 / \$0.02 / \$0.10 (input / cache hit / output). Each reproduces the blended price that Artificial Analysis publishes, except Gemma 4 E4B's, for which our snapshot lists no blended price.

C.9 NON-LLM BASELINES

Two policies bracket the models from below. *Pick first option* always takes the leftmost valid option, including the leftmost enemy as its target, and is deterministic. *Pick random option* draws uniformly among the valid options, from a random stream seeded by the game seed. Both drink every usable potion at the first opportunity, since hoarding potions is a known failure of the models. In shops, the first policy buys only relics (the leftmost it can afford), so the stock of slot zero does not decide the run, while the random policy buys random affordable items until it can afford none. Above them, *BottledAI* (Bottled AI contributors, 2022) is a hand-written symbolic Ironclad player, run with and without its Heart strategy. All three play the same seeds, ascension and game build as the models, but see no names at all, so the reskin does not apply to them.

C.10 HUMAN BASELINES

The streamer data are the public run histories of seven expert streamers. Only Baalorlord's is a complete automatic log over a long period (687 runs, 2020–2026); the others are short player-selected samples, record sets, or single summary figures. Record sets are reported apart from averages, and to put them in scale we also report Baalorlord's best 100 consecutive runs (74% Heart kills, against 63% over his whole log). We also report one of the authors, a non-professional player, over all 50 of their A20 Ironclad runs (2020–2024) and over the first ten. The population data are the run telemetry that the game uploads automatically, as released by its developer, Mega Crit. We use its 206,082 finished Ironclad Ascension 20 runs from October and November 2020, all on the game's final content build. The telemetry carries no player identifier, but every run records the player's total play time in the game to date, so we also report players with at least 1,000 hours. Runs can be chained into play sessions by matching those totals, which lets us report runs played after a Heart kill earlier in the same session; selecting on a kill in the counted run itself would inflate the win rate. Appx. D describes how each source was collected and what it can and cannot tell us.

C.11 ABLATIONS AND ANALYSES

Each ablation changes a single factor from the main table's configuration and keeps each model at its main-table reasoning setting, unless stated otherwise. Gemini 3.8 Flash is also run at low effort, where it lies on the cost Pareto front (Fig. 2a), and is pooled as a separate entry.

Harness. Table 2 compares our default harness with four alternatives. *Single action* allows one action per response. *Single action, by index* does the same and also makes every choice by its number: the tool that chooses an option by name is withdrawn, and every choice screen lists its options as numbered entries, so the two single-action arms differ only in how a choice is named. *Continuous* keeps one conversation for the whole run instead of one per floor. *Orak* hands the whole run to the reflection-and-planning agent of the Orak benchmark (Park et al., 2026), run from its pinned upstream release (commit `c14e5a0`) with its own prompts and a budget of 200 decisions, and passed the reasoning level each model runs at in our harness, so that the two harnesses are compared at the same setting. Orak's own rule bot plays the map, events, shops, rest sites, chests, boss relics, Neow and potions, so the comparison covers combat and card rewards only. Even within a fight its model decides less than ours. Orak asks the model only when a card can be played, so every card choice that a card or potion opens mid-fight (Armaments, True Grit+, Dual Wield, Headbutt, Exhume, an Attack Potion) goes to the rule bot, which picks from a fixed card ranking rather than the fight at hand: the best-ranked card to upgrade or copy, the worst-ranked to exhaust, and, since the flag that would select the best is never set, the worst-ranked card for Headbutt and Exhume. The model is not told which card was chosen, and the rest of its planned turn is dropped. Its model drinks no potions; the rule bot drinks them in boss rooms. Queued actions keep executing through draws and random effects, re-resolved by card identity so a shifted hand still plays the intended card, but the model sees what was drawn only once its queue is spent; the paper describes the state as returned after each action, where the code returns it after the queue. Ours puts every such choice to the model, lets a card-targeting card name its target in the same call, and interrupts the queue when cards are drawn or created, orbs evoke at random or an enemy's intent changes, naming what changed. Orak predates the current effort-based reasoning models, so we made minor changes to its provider code, none of which touch its prompts, agent or decisions. It sends each model the reasoning level, in the form that model's API takes (for OpenAI models, through the Responses API, whose scale reaches the highest

levels). It lets each model use the context window and response ceiling our harness gives it, where Orak's defaults would cut a reasoning model's reply short or truncate its prompt. It reaches Gemini with an API key rather than Orak's authors' own cloud project. And it plays through Orak's own build of CommunicationMod rather than ours, whose choice lists differ on some event screens. These variants were run without the reskin or a memory bank, and so the default arm uses that configuration too. Because each harness was run on a different set of models, the table pools over models and reports each harness's difference from the default, which is what the comparison is about, on matched pairs of runs (the same model on the same seed). Besides performance it compares efficiency: model calls and tokens per floor. Those are compared on like-for-like floors only, meaning the same model, seed, act, room type and occurrence of that room type within the act (the second elite of Act 1, say), and only where every harness run on that model and seed completed the floor. A harness that dies early would otherwise be credited with the cheap early floors, and one that goes deeper charged for the expensive late ones. Orak keeps no per-floor record, so its usage can only be compared per floor reached over whole runs. Table 27 records how every launch of both harnesses ended, including those the harness itself ended through a crash, a stall or a loop, since a run stopped by the harness rather than by a death counts against the model in every per-run figure. The memory ablation (Table 24) is reported the same way, against the arm without memory for the same kind of notes. Per-model results are in Appx. J.

Which part of the game. Table 25 hands the model one part of the game at a time, with Bottled AI (Heart strategy) playing the other three: *combat*, every decision inside a fight; *rewards*, the card, relic and potion picks after a fight, chests and boss relics; *events*, events, shops and rest sites; and *path*, the route through the map. The same splits are run with each baseline policy in place of the model, and every arm is compared on the same seed with Bottled AI playing the whole game, so each difference is the cost of handing over that one part and the baselines show how much each part can cost at all. A further arm hands over nothing: the model, or the baseline policy, plays the whole game in the same configuration, which places each single part between playing everything and playing nothing. Screens that the four parts do not name, such as a card choice opened by a shop's removal service or a rest site, belong to whichever part opened them. These arms use the themed reskin, xml notes and no memory bank. Table 25 gives each model's change in floor, paired on seed, and Table 30 the absolute levels.

Reskin. Table 23 compares each reskin with the real game names on the same model and seed, pooled over models, in the same way as the harness and memory comparisons above. The permuted and nonsense mappings separate the two things the themed reskin does at once, hiding each real name and supplying a coherent replacement for it (Sec. 3). Per-model results are in Table 29.

Difficulty. Figure 11 (Appx. G) repeats the main configuration at a range of pinned ascension levels: every level from 0 to 20 for the non-LLM baselines, and levels 0, 1, 5, 10, 15 and 20 for the models run at more than one. Each level is a separate cohort with its own memory chain.

Cost and reasoning frontiers. Figures 2a, 20 and 21 plot mean acts cleared against mean cost per run, cost per floor and thinking tokens per floor, with one point for each model at each reasoning level it was run at, since an agent's accuracy means little without its cost (Kapoor et al., 2025). A point lies on the Pareto front when no other point clears at least as many acts for no more spend; the front is computed over cohorts with all ten seeds.

Public benchmarks. Figure 2b and Table 14 compare mean acts cleared with the scores Artificial Analysis publishes for the same models, taken from a snapshot of its leaderboard (the retrieval date is in the table caption), for the LLM agents of Table 1 with all ten seeds at their highest reasoning effort, and again at low effort where that cohort has all ten seeds and Artificial Analysis lists the setting (it does not for Qwen3.5 122B-A10B, Qwen3.6 35B-A3B or either GLM-5.3 model). Each point is matched to the Artificial Analysis entry for the reasoning setting it ran at here; two matches are approximate, since Artificial Analysis lists Claude Haiku 4.5 without its thinking budget and GLM-5.3 Flash under a single setting. We use the evaluations reported for nearly every one of these models, among them Humanity's Last Exam (Phan et al., 2025), GPQA Diamond (Rein et al., 2024) and SciCode (Tian et al., 2024). A correlation with the public suite says little on its own, because the suite's evaluations are correlated with each other (Burnell et al., 2023; Ruan et al., 2024); so we also

add SpireBench to the suite as one more row and ask, for every row, how well the rest stand in for it: its strongest correlation with any other single evaluation, and how much of it the first principal component of the others predicts for a model left out of the fit. The Intelligence Index is a composite of evaluations like these and is left out of that part. The two rungs of one model are not independent points, so a model is held out with both its rungs, and we also report the correlations within each rung.

Memory growth. Every model call logs the length of its system prompt, and within a cohort the memory bank is the only part of the system prompt that changes between runs. The first call of run $k + 1$ therefore measures the bank left by run k, relative to run 1's empty bank (Fig. 17). This is measured in characters so that sizes are comparable across tokenizers; the bank left after the last run is never read, so the series ends at run 9.

Learning from memory. Figure 18 compares each memory cohort's mean acts cleared over seeds 1–5 with seeds 6–10. The two halves are different games, so the same model's no-memory cohort on the same seeds is shown alongside as a control for the halves' difficulty: learning appears as a larger improvement than the control's.

Tool-call errors. The harness counts four kinds of tool-use failure, reported per 100 model calls in Table 15 so that models making more calls are not charged for it: a response in which no call could be executed, because every call was malformed or refused or none could be parsed; a single call refused before it reached the game, as illegal on the current screen or with invalid arguments; a turn ended with a different amount of energy left than the model declared, which marks a model that has lost track of its own turn; and a call written as prose rather than through the tool interface, which the harness recovers. The model is told about the first three and the harness asks again or carries on, so none of them ends a run; they measure how precisely a model acts, separately from how well it plays, as function-calling benchmarks measure tool use on its own (Patil et al., 2025). The harness comparison (Table 2) reports the same rate.

Calibration. Table 12 (Appx. H) scores the model's predictions (Appx. C.2) against what happened, over the runs of Table 1. A floor's act-survival prediction is scored as correct if the run went on to clear that act, and a run-win prediction if it killed the Heart. We report the mean predicted and observed rates, the Brier score (Brier, 1950), whose reliability term (Murphy, 1973) is the calibration error of Fig. 13, and the AUC, which measures whether the model gives higher predictions in the acts it goes on to survive regardless of their overall level. HP predictions are matched to the next recorded HP on the same floor, and a prediction made in the fight that killed the player is scored against 0 HP. A prediction is easier the closer it is made to the moment it resolves: the end of the fight, the end of the act (or the death that ends it early), the end of the run. Stronger models survive longer and so make more of their predictions close to an act's end, which would flatter their raw scores. We therefore bin every prediction by its distance from resolution (Fig. 14) and report scores standardized to a common mix of distances: each error is fitted as a model effect plus a distance-bin effect, and each model's fit is averaged over the distance mix of all models' predictions pooled. The AUC is computed within each distance bin and averaged with the same weights. Next-turn HP is always one turn ahead and needs no adjustment.

D HUMAN BASELINES

Human baselines. We compare against published A20 Heart Ironclad results from seven top players, with the same act-clearing definitions as our harness. They differ in how the data were collected, and this determines what each figure means. Only Baalorlord's is an automatically logged long-run average: the Spireblight client uploads every run file from his game directory, abandoned runs included, and we use all 687 Ironclad A20 runs on his main profiles from 2020 to 2026 (`https://baalorlord.tv/runs`). Three further samples of per-run files (panacea108, vmService, cheg0) come from a public archive (`https://github.com/MaT1g3R/Slay-the-Spire-data`). They are contiguous blocks of one to three months, and the two rotating samples cycle through the characters without a break, consistent with no runs missing within them; who chose each window is not stated. XecnaR's figure comes from his own per-run spreadsheet of a 100-run Ironclad set (`https://docs.google.`

Table 4: How each human baseline was collected. *Complete block*: every run over the period, logged automatically. *Player-selected*: a contiguous block shared by or for the player, chosen by unstated means. *Record set*: a player-kept set known because it set a record. *Summary*: a figure copied from the player's own summary, with no per-run data.

Player	Source	Runs	Period	Coverage	Notes
Baalorlord	site log of `.run` files	687	2020-04 to 2026-02	complete block	abandons kept; older modded profile excluded
panacea108	archive, `.run` files	51	2022-09 to 2022-11	player-selected	Ironclad-only sample
vmService	archive, `.run` files	50	2022-09 to 2022-12	player-selected	Ironclad runs of a 200-run rotation
cheg0	archive, `.run` files	46	2024-02	player-selected	Ironclad runs of a 184-run rotation
XecnaR	player-kept sheet	100	unknown	record set	act clears from cause of loss; no floors
Lifecoach	VOD description	100	to 2022-03	summary	Heart win rate only; one run marked "not counted"
TerrenceMHS	player-kept records sheet	50	unknown	summary	best 50-run window; Heart win rate only

Table 5: Human record sets and best windows at A20 with the Ironclad and the Heart: a player's best-known stretch of results rather than typical play, so they are kept out of Table 1. Columns as there; how each was collected is in Table 4. §From the player's own summary rather than per-run data; unreported metrics are shown as –.

			Act win rate (%)				Acts	Floor		
	Player	Runs	1	2	3	4	(mean)	mean	median	[min–max]
	XecnaR	100	98	94	91	84	3.67	–	–	–
	Lifecoach§ (to Mar 2022)	100	–	–	–	81	–	–	–	–
	TerrenceMHS§ (best 50)	50	–	–	–	70	–	–	–	–

`com/spreadsheets/d/1hNrqA5JlXgz8Ie-bdCToloTeF2JmSDQpxtA8fiGyucE`) that set a record. Lifecoach's (`https://www.youtube.com/watch?v=eJ5vSByGwQ4`) and TerrenceMHS's (`https://docs.google.com/spreadsheets/d/1pERvNN0EIUzQmIKeJ0XGWPfYHWztiSD04LRftv_qFK4`) are summary lines: a 100-run record set and an apparent best 50-run window. These last three are best read as near-peak human play, not typical performance.

Population baseline. For the general player population we use Mega Crit's own run telemetry, which the game uploads automatically at the end of every run; the developers published 7.05M runs from October to mid-November 2020, all on the final content build (`2020-07-30`), at `https://drive.google.com/drive/folders/1c7MwTdLxnPgvmPbBEfNWa45YAUU53H0l`. Unlike self-uploaded run archives it is not selected by the player, so it reflects everyone playing Ironclad at Ascension 20, a level that is itself only unlocked by winning at Ascension 19 with the Ironclad. We keep Ironclad A20 runs (290,280), drop daily, endless, seeded, trial and modded runs and any on another build or the beta branch (249,362 left), and remove runs abandoned before a death or victory, nearly all quit at Neow, leaving 206,082. Act clears use the harness's definitions, with floors checked against where each boss's victims died. The record has no player identifier, but each run carries its player's cumulative Slay the Spire play time (the in-game statistic, not a cross-game or Steam figure), so we also stratify by hours played. By requiring each run's cumulative play time to equal the previous run's plus that run's length, with the run starting within an hour of the previous one ending, we reconstruct chains of consecutive runs (62% of runs link, against 1.0% with a deliberately wrong offset). Chains break at longer pauses, so they are play sessions rather than players, but they let us count only runs made after an A20 Heart kill earlier in the same chain, avoiding the bias of selecting on the outcome being measured. The population clears Act 1 in 47% of runs and kills the Heart in 3.5%.

Table 6: Population baseline for A20 Heart Ironclad, from Mega Crit's run telemetry (build `2020-07-30`, 2020-09-30 to 2020-11-12). Hours is the median cumulative Slay the Spire play time of the player at the run. Act k is the share of runs that cleared Act k (%); Acts is the mean number of acts cleared per run.

			Floor		Act cleared (%)				
Population	Runs	Hours	mean	median	1	2	3	4	Acts
All A20 Ironclad runs	206,082	300	22.1	16	47.2	15.3	6.9	3.5	0.73
All A20 Ironclad runs, incl. abandoned	249,362	298	18.6	16	39.7	12.8	5.7	2.9	0.61
StS hours < 100 h	13,615	77	21.1	16	43.9	14.9	7.0	2.4	0.68
StS hours 100–300 h	89,476	201	21.6	16	45.6	14.4	6.3	2.8	0.69
StS hours 300–1000 h	94,704	466	22.5	16	48.6	15.8	7.2	3.9	0.76
StS hours $\geq$ 1000 h	8,287	1253	24.7	21	53.5	20.4	10.9	7.1	0.92
Most StS hours (top 10%)	20,609	920	23.8	20	52.0	18.1	9.1	5.5	0.85
Session chains with an A20 Heart kill (biased)	16,749	371	36.3	35	68.8	50.0	45.1	42.8	2.07
Runs after an A20 Heart kill earlier in the chain	3,641	433	27.9	23	58.0	29.2	20.9	15.8	1.24

Table 7: A20 Heart Ironclad by hours of Slay the Spire played, from Mega Crit's run telemetry (build `2020-07-30`). Hours is the player's cumulative play time when the run ended (bin and bin median). Act k is the share of runs that cleared Act k (%); Act 4 is killing the Heart. Acts is the mean number of acts cleared per run.

			Floor		Act cleared (%)				
Hours	Runs	Median h	mean	median	1	2	3	4 (Heart)	Acts
0–25	329	15	24.9	19	51.1	22.8	15.5	4.0	0.93
25–50	1,544	42	21.2	16	42.9	16.7	9.1	2.9	0.72
50–100	11,742	81	21.0	16	43.8	14.5	6.4	2.4	0.67
100–150	18,812	128	21.4	16	44.6	14.3	6.3	2.5	0.68
150–200	25,433	176	21.3	16	45.2	14.2	6.0	2.7	0.68
200–300	45,231	245	21.8	16	46.2	14.6	6.4	3.0	0.70
300–400	31,906	346	22.1	16	46.9	15.3	6.9	3.7	0.73
400–500	21,945	447	22.4	16	48.1	15.9	7.1	3.7	0.75
500–750	30,077	598	22.8	18	50.1	15.9	7.5	4.2	0.78
750–1000	10,776	839	23.1	18	50.8	16.4	7.8	4.4	0.80
1000–1500	5,962	1162	24.7	21	53.1	20.3	11.3	7.6	0.92
1500–2000	1,433	1715	25.6	22	57.2	21.3	11.4	6.8	0.97
$\geq$2000	892	2153	23.8	18	50.7	19.1	7.4	4.0	0.81

Experience and performance. Each telemetry record carries the player's cumulative Slay the Spire play time when the run ended, so we can stratify the 206,082 A20 Ironclad runs of Table 6 by hours played (Table 7). Performance rises with experience, but slowly: runs by players with 50–100 h clear 0.67 acts on average and kill the Heart in 2.4% of runs, against 0.97 acts and 6.8% at 1500–2000 h, the best bin. Play time is a lower bound on experience: it lives in the save profile and resets with a new one, so the lowest bin (329 runs under 25 h at A20, which takes far longer than that to unlock) likely holds experienced players on a fresh profile. It also measures time, not skill. Bins with fewer than 200 runs were merged: 2000–3000 h with 3000 h and above.

Figure 7 plots Table 7 against hours played. Between 25 and 1,000 hours performance barely moves (0.67–0.80 acts per run, 2.4–4.4% of runs killing the Heart); it rises only past 1,000 hours (0.92–0.97 acts, 6.8–7.6%). The lowest bin is high because play time resets with a new save profile, so it likely holds experienced players on fresh profiles.

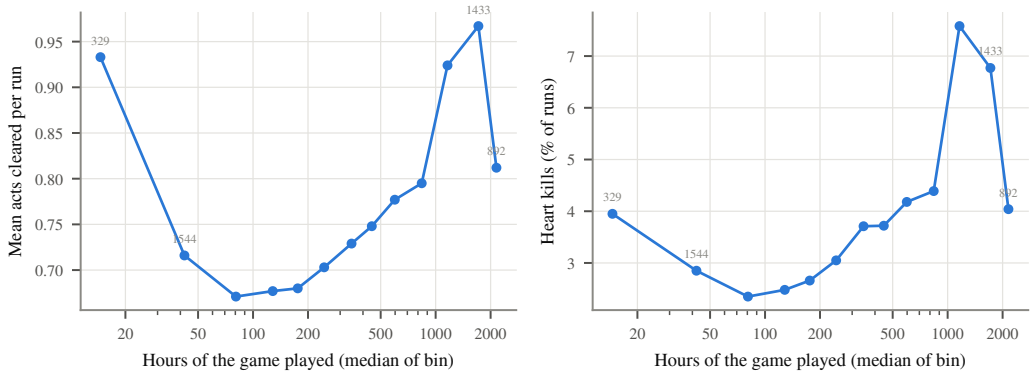


Figure 7: Human performance at A20 with the Ironclad against hours of the game played, from the telemetry in Table 7: mean acts cleared per run (left) and the share of runs that kill the Heart (right). Each point is a bin of Table 7 at its median hours; bins with fewer than 2,000 runs are labelled with their count. Play time resets with a new save profile, so the lowest bin likely holds experienced players on fresh profiles.

E How agents and humans differ where agents die

We compare the decks and in-game choices of our agents with those of human players at the floors where the agents die. The agents are the 200 main-table runs (20 LLMs, A20 Ironclad on the held-out seeds, themed reskin, xml notes and memory, generic guidance), with per-floor state from each run's logged history. The humans are the 206,082 finished A20 Ironclad runs of the game's final content build (2020-07-30) in Mega Crit's released run telemetry for October and November 2020 (`https://github.com/rrenaud/slay_analysis`), with beta, daily, seeded, trial, modded, anomalous and abandoned runs removed. Of the agent runs that die, 27% die at the Act 1 boss (floor 16), 11% on floor 7, 7% on floor 22 and 6% at the Act 2 boss (floor 33). On reaching these floors the agents die more often than humans: 41% against 32% at the Act 1 boss, 12% against 4% on floor 7 and 20% against 6% on floor 22 (Table 9).

The agents reach the Act 1 boss healthier than humans but built differently (Table 8). They arrive with 82% of their maximum HP against 62%, having spent 68% of their campfires resting where humans rest at 23% and upgrade a card at the rest. They fight fewer elites (1.25 against 1.96), and they take a card after almost every fight (none after 8% of fights, against 23%), so their decks are larger (19.3 cards against 16.8). They also drink more of their potions before the boss (0.50 of those obtained, against 0.27); among humans, those who then die at the boss had drunk 0.40 of theirs and those who pass it 0.21. At death, against humans who died on the same floor, the agents' decks hold 1.4 fewer upgraded cards and 2.5 more cards, and they carry 1.9 fewer relics (1.8 fewer upgrades and 2.2 fewer relics at the Act 1 boss).

These are descriptive differences, not matched comparisons: the agents play a reskinned game on at most ten held-out seeds, while the humans play the real game on many different seeds, and every human at A20 has, by the game's unlock rule, won at A19 with the Ironclad. Four measurements are approximate. Human deck size on entering floor 16 is reconstructed backwards from the final deck, since the record lists only the final deck and does not itemize Neow's card effects; it agrees with an independent forward count for 79% of runs. Agent potion use counts `use_potion` calls, and the potions obtained are inferred from the potion count, which misses discards, so the agent rate is an upper bound. The agent skip rate counts fights after which no card entered the deck. Some human records carry implausible gold values (0.25% reach 5,000 or more), so gold is reported as a median.

Health as a resource. The telemetry above shows the agents reaching the Act 1 boss with more HP than humans because they rest where humans upgrade. Figure 3 follows HP over the whole run against the four Twitch streamers of Appx. D (833 runs with per-floor records). The streamers spend HP through Act 1: after the floor-15 rest site, the last before the boss, their median is 53% of maximum HP, against 90% for the four models that clear the most acts. At a rest site the models rest

Table 8: State and Act 1 choices on entering the Act 1 boss (floor 16), for the runs that reached it, split by whether the run then died at the boss. Campfire, fight and potion rates are pooled over campfires, fights and potions on floors 1–15 (potions 0–15); HP, elites, deck size and potions used at the boss are means over runs.

	Agents			Humans		
	All	Died	Passed	All	Died	Passed
HP (% of max)	82	75	86	62	56	65
Campfires spent resting (%)	68	75	63	23	26	22
excluding Recall (%)	72	81	67	23	27	22
Elites fought	1.25	1.21	1.28	1.96	1.85	2.01
Fights with no card taken (%)	8	12	6	23	26	21
Deck size	19.3	18.4	19.9	16.8	16.5	17.0
Potions used / obtained	0.50	0.55	0.46	0.27	0.40	0.21
Potions used at the boss	1.08	1.19	1.00	0.56	0.82	0.44
Gold (median)	102	112	80	152	160	149
Runs	127	52	75	143,146	45,865	97,281

Table 9: Where runs end: the share of all deaths that fall on a floor, and the death rate among the runs that reached it. Agents: 196 deaths in 200 runs (the rest cleared Act 3, one of them going on to kill the Heart). Humans: 195,533 deaths in 206,082 runs.

	Share of deaths (%)		Death rate on reaching (%)	
Floor	Agents	Humans	Agents	Humans
7	11	4	12	4
16 (Act 1 boss)	27	23	41	32
22	7	3	20	6
33 (Act 2 boss)	6	8	37	34

unless they are nearly full, 76–98% of the time when they arrive below 80% of maximum HP, where the streamers rest 6–48% of the time and upgrade a card otherwise; over all visits the streamers rest at 15% (6,398 visits), the top four models at 66% (238) and the other models at 75% (364). The strongest models manage HP no differently from the weakest: none treats it as a resource to trade for a stronger deck. The streamers play their own seeds, so this is a reference rather than a matched comparison.

Routes, relics and deck. Figures 8 and 9 compare the run-level choices of the LLM agents of Table 1, in three groups by acts cleared, with the telemetry population and the four Twitch streamers. In Act 1 the models fight fewer elites (1.4 for the top four models, 1.3 and 0.9 for the next eight and the bottom eight, against 2.0 for the population and 2.6 for the streamers; 56–77% of model runs reaching the boss fought at most one, against 29% and 4%), enter more shops (1.3–1.6 against 1.0 and 0.8), rest at 57–75% of rest sites against 23% and 9%, and take a card after almost every fight (3–13% of fights with no card taken, against 23% and 15%). They therefore fall behind in relics from the first elites and never catch up (9 against the streamers' 12 at floor 32), carry fewer upgraded cards, and hold a larger deck.

These choices do not separate the stronger models from the weaker ones (Fig. 10). Across the 16 models with at least three runs reaching the Act 1 boss, Spearman's ρ with acts cleared is $+0.37$ for elites fought ($p = 0.16$), -0.25 for shops, -0.20 for resting and -0.21 for skipped card rewards (permutation tests), whereas the net HP change over a hallway fight has $\rho = +0.62$ ($p = 0.01$), from -2.4 HP for the top four to -6.5 for the bottom eight (-0.2 for both human groups). A logistic model of beating the Act 1 boss from the state on reaching it (HP, maximum HP, relics, deck size and upgrades), fitted on the population, rates the top four's positions at 79% and the next eight's at 70%; they actually won 94% and 55% of those fights. The streamers arrive in positions the model rates at 69%, having spent HP for relics and upgrades, and win 94%. The stronger models are stronger mainly in the fights, not in the route they take to them. The state model explains little of the outcome

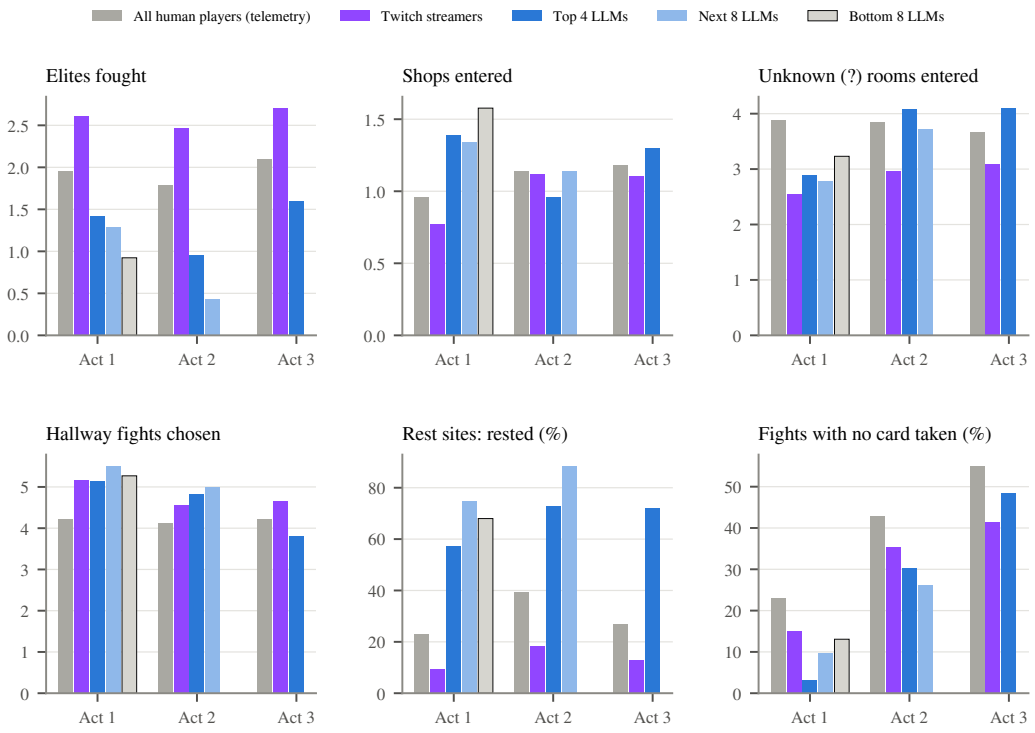


Figure 8: What runs did on each act's 15 map floors, over the runs that reached that act's boss (floors 16, 33 and 50): the telemetry population (143,146, 47,619 and 22,729 runs), the four Twitch streamers (788, 671 and 603) and the LLM agents of Table 1 in three groups by mean acts cleared: the top 4 (36, 23 and 10 runs), the next 8 (65 and 7) and the bottom 8 (26). A group is drawn for an act only when at least five of its runs reached that boss. Shops and unknown rooms count the map nodes entered; elites fought and hallway fights the rooms entered. Rest-site and card-reward shares pool the visits and fights of all runs in the group.

on its own (McFadden R^2 0.05), and the human deck, relic and upgrade counts are reconstructed backward from each run's final state, since the telemetry records only that.

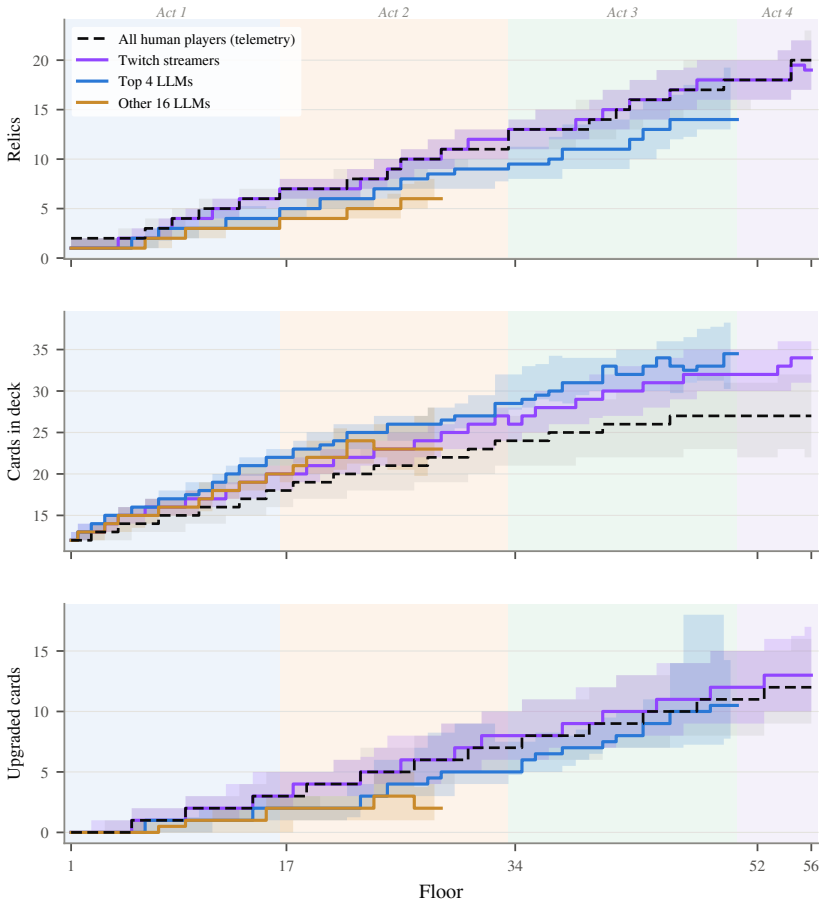


Figure 9: Relics, cards in the deck and upgraded cards at the end of each floor survived: median (lines) and interquartile range (bands), for the telemetry population (dashed), the four Twitch streamers, the top 4 LLM agents of Table 1 by acts cleared and the other 16. A floor is drawn only while at least 20 human or 8 LLM runs are alive on it. Human counts are reconstructed backward from each run's final deck and relics, removing every change the record itemizes after the floor.

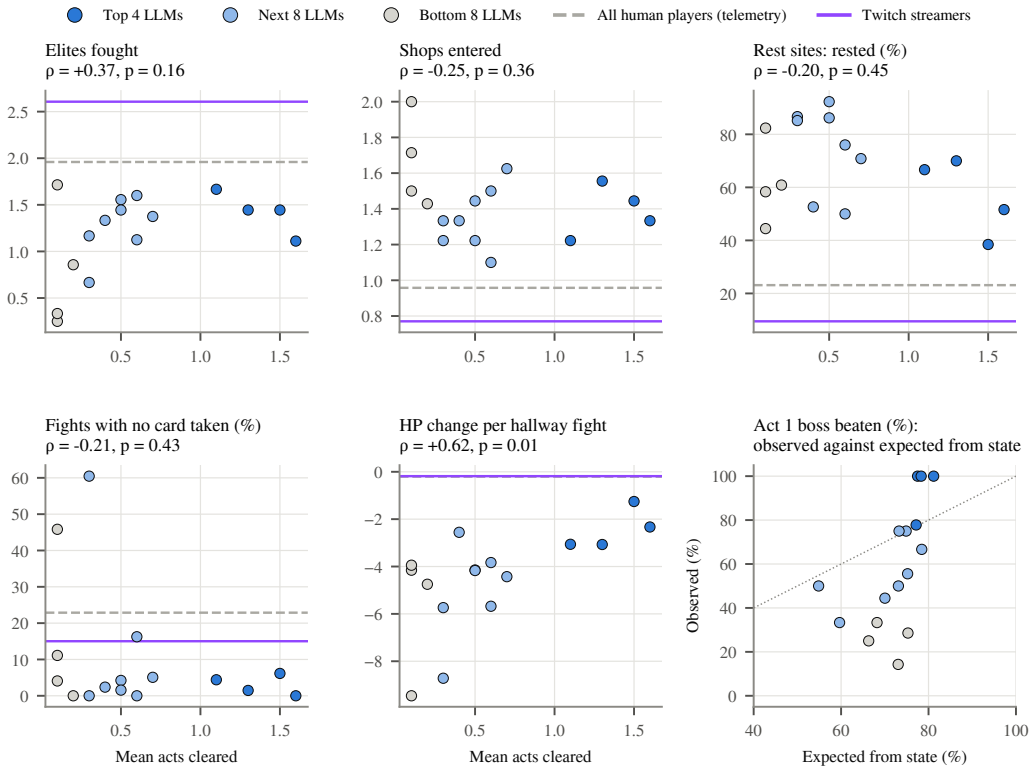


Figure 10: Act 1 choices and combat against mean acts cleared, one point per LLM agent of Table 1 with at least three runs reaching the Act 1 boss (16 models), over those runs; lines are the telemetry population (dashed) and the four Twitch streamers. ρ is Spearman's correlation across the models and p a two-sided permutation test. HP change per hallway fight is the net HP change over a hallway-fight floor, including the relic heal after it, computed the same way for humans. Last panel: each model's rate of beating the Act 1 boss against the rate expected from its runs' states on reaching it, by a logistic model fitted on the 143,071 telemetry runs that reached it with a complete state record; the dotted line is equality.

F STRATEGY AND PLAY QUALITY

This appendix asks two questions of the runs of Table 1: why the models fall behind human experts, and why the strong models beat the weak ones. Tiers are fixed on all 20 models by acts cleared: the top four (Claude Fable 5.1, GPT-6 Astra, Claude Opus 5, GPT-5.6 Sol), the next eight and the bottom eight. Gemma 4 E4B has no full logs, so analyses read from the logs cover 19 models and the bottom tier has seven. Correlations across models are Spearman with permutation p; paired contrasts use sign-flip or exact sign tests.

Conservative Act 1 play among humans. In Mega Crit's A20 Ironclad telemetry we took the human runs that beat the Act 1 boss and compared those whose Act 1 looks like the models' (at most one elite, resting at half or more of the rest sites, at least 70% HP at the boss) with those that fought two or more elites and rested at fewer than half (Table 10). The model-like profile covers 3.2% of human runs and 36% of the models'. It clears Act 2 in 15.5% of cases against 38.0%, and Act 3 in 3.4% against 18.6%, and the gap holds in every band of hours played (14% against 45% for Act 2 among players with over 1000 hours). Clearing Act 2 rises steadily with Act 1 elites (21, 25, 32, 40 and 50% for 0 to 4 or more) and falls with rests (37, 29, 24 and 21% for 0 to 3 or more). The telemetry has no player identifier, so the strongest design compares runs within one play session: there each Act 1 elite multiplies the odds of clearing Act 2 by 1.38 (1.34–1.43) and each rest by 0.85 (0.81–0.89), over 8,745 sessions, and model-like runs clear Act 2 7.3 points less often than the same session's other runs (permutation $p = 10^{-4}$). Within each of the four streamers the odds ratios are 1.68 per elite and 0.64 per rest. Resting at a rest site entered with at least 70% HP, which the models do at 31% of such sites, the population at 2% and the streamers never, predicts worse outcomes across the population but not within a session, so its association is likely selection on weaker players.

How much of the gap does the route explain? Shifting the streamers' odds by the effect of the models' Act 1 profile accounts for 17% of their Act 2 advantage and 19% of their Act 3 advantage under the within-session model, and 34–61% under cross-sectional models, which also absorb the effects of deck and relics and so overstate it. At their own profile the models are not worse than humans: humans with that profile clear Act 2 in about 19% of cases, the models in 25% (the top four in 47%), while the streamers clear 85%, 45 points above what their profile predicts. These are associations: weak runs skip elites and rest partly because they are weak, and the telemetry does not record which elites the map offered.

Risk in the models' own runs. Elites are where the models die before the Act 1 boss: 37 of the 65 such deaths are in elite rooms, a death rate of 17.8% per elite fight against 2.7% per ordinary fight, higher for 15 of 18 models (sign test $p = 0.008$). The deaths are concentrated in the weak models (Qwen3.5 122B 6 of 10 elite fights, gpt-oss-120b 5 of 7), while the top four die in 0–9% of theirs, so for a model that fights badly, avoiding elites is partly the right call. Among runs that beat the Act 1 boss (49 runs, 7 models), upgrades on reaching the boss predict clearing Act 2 ($r = 0.56$, Holm-adjusted $p = 0.006$) and rests predict against it ($r = -0.44$), while the number of elites fought does not ($r = -0.12$). Part of the upgrade effect is that a run with HP to spare can afford to upgrade: controlling for HP on arrival weakens it to $r = 0.27$ ($p = 0.11$). HP on reaching the boss itself predicts clearing Act 2 ($r = 0.32$), which says that a deck able to clear Act 1 unhurt is strong, not that HP should be hoarded. In 72 of the 75 Act 1 clears the transition heal was exactly 75% of missing HP.

Learning over the chain. Comparing runs 1–5 with runs 6–10 of each memory chain, with no-memory chains on the same seeds as the control for how hard each half is (8 models), memory changes nothing measurable: its added effect is -0.07 on the share of rest sites spent resting ($p = 0.29$), -0.32 Act 1 elites ($p = 0.38$, 5 models reaching the boss in both halves) and null on upgrades, HP at the boss, floor and acts ($p \geq 0.29$). Power is low, but the direction is towards more caution, not less. The final memory banks lean the same way: 119 of 1,374 memories advise caution and 17 advise risk (keyword-coded, 20 of 25 sampled caution codes confirmed by hand), and 18 models hold more cautious than risky memories, none the reverse. The main exception is Claude Opus 5 ("DODGING FIGHTS IS AN ERROR, BECAUSE UNSPENT SLACK EXPIRES"). Only the four strongest models state the act heal correctly; Gemini 3.8 Flash writes that it "restores player HP up to 80% of Max HP", GLM-5.3 that it is "likely flat ~50", and ten models never record it. Knowing the rule does not settle the decision: 32 of 209 late-act rest-site decisions (floors 13–15, 30–32 and 47–49) reason about the heal, mostly by Claude Fable 5.1 and Opus 5, and still rest in 21 (66%,

Table 10: Act 1 play and what follows, over runs that beat the Act 1 boss. Humans are Mega Crit's A20 Ironclad telemetry and the four Twitch streamers of Table 1; agents are the LLM runs of Table 1. Elites, rest share (rest-site visits spent resting rather than upgrading), HP (% of maximum), relics and upgrades are on entering the Act 1 boss. The lower block splits the telemetry by profile: *model-like* is at most one elite, a rest share of at least 0.5 and at least 70% HP at the boss; *aggressive* is two or more elites and a rest share below 0.5. Human runs pass a plausibility filter first, so their counts differ slightly from Table 8.

	Runs	Elites	Rest share	HP	Relics	Upgrades	Clear A2 (%)	Clear A3 (%)
Telemetry population	97,220	2.01	0.22	64	5.7	3.3	32	15
Streamers	741	2.61	0.09	56	5.7	3.4	85	77
LLM agents, all	75	1.28	0.63	86	3.9	2.0	25	5
LLM agents, top four	34	–	–	–	–	–	47	12
Telemetry, model-like	3,080						15.5	3.4
Telemetry, aggressive	54,163						38.0	18.6

against 77% for the others). Opus's own bank records the conflict: "PRE-BOSS CAMP … the Sleep LOSES ON THE LEDGER … AND WINS ANYWAY".

Deck building and boss preparation. We scored 1,621 card rewards against a choice model fitted on about two million reward screens from the telemetry, as the probability that an average human makes the same pick on the same screen. Agreement barely rises with strength (0.324, 0.313 and 0.293 by tier; $\rho = 0.16$, $p = 0.52$); on the 92 identical screens seen by both a top and a bottom model the top model agrees more by 0.036 ($p = 0.03$). The models skip 10–15% of rewards where the human model expects 24–31% on the same screens, unrelated to strength. The one clear signal is area damage: on Act 1 screens offering such a card, the top four take one 46% of the time, the next eight 30% and the bottom eight 20%, against about 34% for humans on the same screens ($\rho = 0.76$, $p = 0.0002$; 29 points more on 28 identical screens, $p = 0.002$), and their reasoning cites multi-enemy fights rather than the boss. Scaling, block and draw cards show nothing. Although 92–98% of runs in every tier name the upcoming boss before reaching it, no tier adapts its deck to it: top decks facing the Slime Boss, which splits into several enemies, hold 1.94 area-damage cards against 1.50 for other bosses ($p = 0.37$), and the models' picks follow none of the shifts humans make by boss. Given the state in which each run reaches the Act 1 boss (HP, maximum HP, relics, deck size and upgrades), 13% of the 75-point top–bottom gap in boss wins is explained by a model fitted on humans and 24–29% by one fitted on the models; the residual, 52–57 points ($p \leq 0.01$), is in the fight or in what the state omits.

Combat. From the full logs we measured each combat turn against the state the model was shown (Table 11). The measures were validated first: a card's displayed damage matches the HP and block it removed on 99.4% of 3,030 checked plays, and the shown intents less block match the damage taken on 92% of turns; intent-based measures use only the consistent turns. Measures that do not separate the tiers are as informative as those that do: unblocked damage the hand could have prevented is about 0.1 HP per turn in every tier, unspent energy 5–6%, energy spent 2.6–2.7 per turn and the share of incoming damage blocked 48–52%. The strong models do not defend better; they attack better. Paired on the same hallway encounter of the same seed (73 encounters), the top four take 6.7 less damage than the bottom tier, in 0.99 fewer turns, dealing 5.3 more damage per turn (sign-flip $p < 0.001$ each); on floors 1–6, where decks are still close to the starter deck, the gaps are 2.6 HP ($p = 0.002$) and 0.82 turns. About half of the difference in damage taken comes from the extra turns and half from damage per turn, mostly from attackers left alive. The next-turn HP forecast adds nothing in Act 1 beyond these measures (partial $\rho = -0.05$ after controlling for hallway damage, blocking into non-attacks and missed lethal). Two examples, with the names the model saw: on seed 6, floor 8, facing a Zealot [Cultist] whose intent is a buff, Qwen3.8 27B plays three block cards ("The 20G fully covers the T2 hit … at zero HP cost") and deals 4 damage, while GPT-6 Astra on the same turn deals 30; and Qwen3.5 122B, with a 9-HP Corrosive Ooze [Acid Slime] and 6 + 4 damage in hand, computes "Trip Attack (2 energy): 4 damage … not enough to kill" by applying its own Fatigue [Weak] a second time, and leaves it at 3 HP.

Table 11: Combat play in Act 1 by tier of Table 1 (top four, next eight, bottom seven with full logs), with Spearman's ρ against mean acts cleared over the 19 models and its permutation p. Hallway fights are ordinary combats. Block into no attack: turns on which a block card is played although no enemy intends to attack. Over-block: a block-only card played beyond the incoming damage while an attack was playable. Missed lethal: turns with one enemy left and enough damage in hand to kill it. The lower block shows measures that do not separate the tiers ($p > 0.4$).

	Top 4	Next 8	Bottom 7	ρ	p
Damage taken per hallway fight	8.2	9.6	14.0	-0.87	< 0.001
Turns per hallway fight	3.09	3.58	4.06	-0.68	0.002
Damage taken per hallway turn	2.65	2.77	3.43	-0.57	0.012
Damage dealt per energy on attacks	7.5	6.7	5.8	$+0.91$	< 0.001
Damage dealt per energy, all cards	5.7	4.7	3.8	$+0.76$	< 0.001
Damage dealt / attack damage in hand	0.95	0.86	0.74	$+0.65$	0.003
Block into no attack (% of turns)	6	15	31	-0.85	< 0.001
Over-block (%)	0.8	3.0	6.4	-0.68	0.001
Missed lethal (%)	0	0.8	8.7	-0.78	< 0.001
Plays refused as not in hand (per 100 turns)	1.1	2.0	10.8	-0.58	0.012
Next-turn HP forecast error (HP)	0.06	0.44	2.12	-0.83	< 0.001
Potions drunk in hallway fights (%)	27	27	61	-0.77	< 0.001
Preventable unblocked damage (HP per turn)	0.11	0.06	0.07		
Energy left unspent (%)	6	5	5		
Energy spent per turn	2.57	2.68	2.69		
Share of incoming damage blocked	0.48	0.52	0.50		

G Difficulty and where runs end

Figure 11 repeats the main configuration at a range of pinned ascension levels, and Fig. 12 breaks down in which act the runs of Table 1 ended.

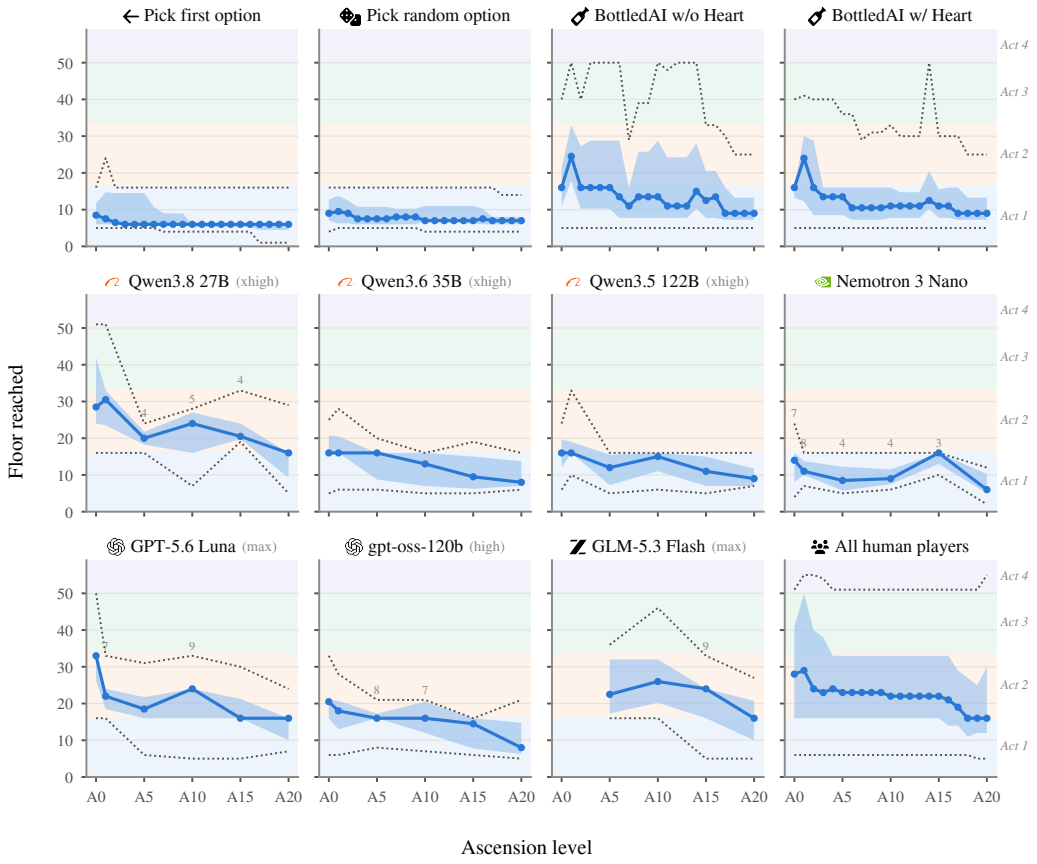


Figure 11: Effect of difficulty on performance. Final floor reached against the pinned ascension level, for every agent in Table 1 with runs at more than one level, on the held-out seeds under the same configuration as the table. Each level is a separate cohort (its own memory chain). The solid line is the median floor at each level, the blue band the interquartile range, and the dotted lines the lowest and highest floor reached. Background shading marks the floors of each act. Levels with fewer than 10 completed seeds are annotated with their seed count. The last panel is every Ironclad run in Mega Crit's released telemetry (October–November 2020, final content build; abandoned runs excluded), with the dotted lines at the 5th and 95th percentiles instead, since over tens of thousands of runs per level the extremes are always the first and last floors reachable. Each level there is a different population: a level is unlocked by winning at the one below it, so higher levels hold more experienced players, and only Ascension 20 includes runs that continue to the Heart.

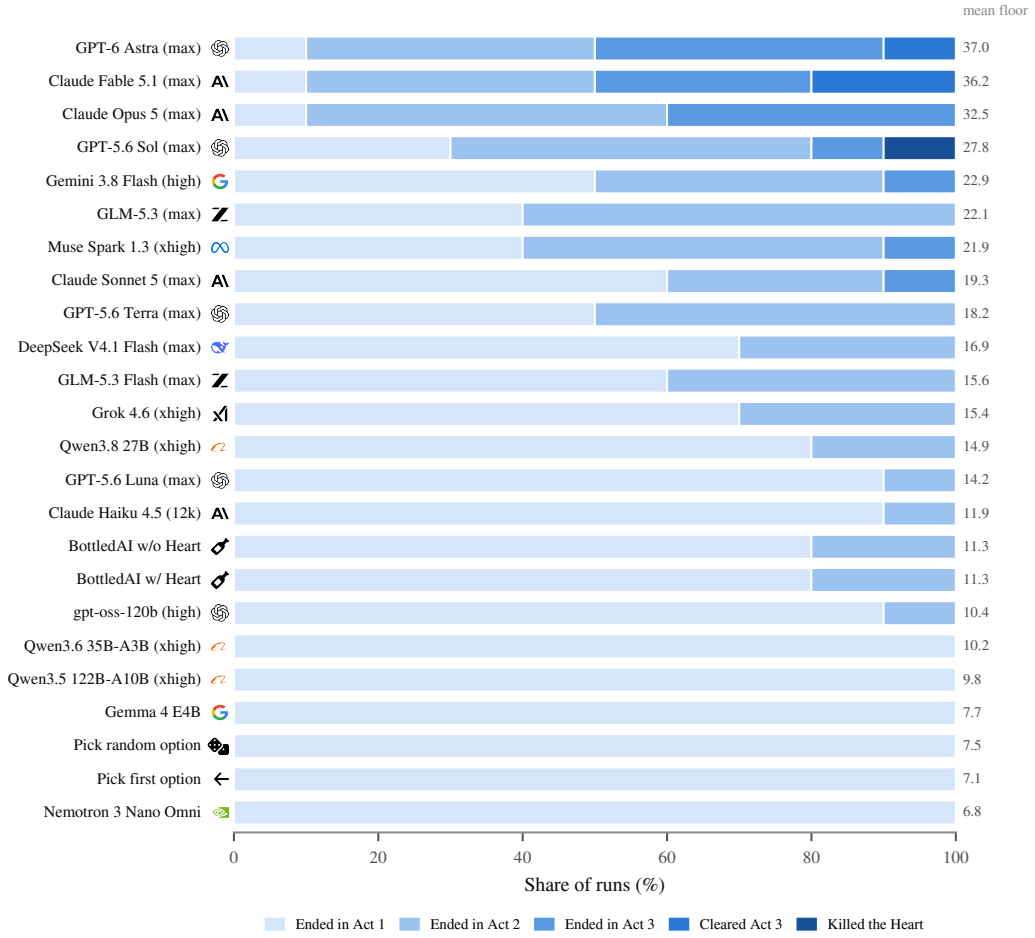


Figure 12: Where runs end. For each agent in Table 1, the share of its runs that ended in each act, sorted by mean floor (right). Agents with a similar mean floor can differ in whether they reliably reach the next act or split between early deaths and deep runs.

H CALIBRATION

Table 12 scores the models' predictions of their own game; how the error depends on the distance to resolution is in Fig. 14. Figure 13 relates each model's end-of-floor calibration error to the acts it clears, and Fig. 15 does the same for its within-floor HP forecasts, whose error tracks play more closely (Table 13).

Table 12: How well the models predict their own game. At the end of every floor each model is asked for its chance of surviving the current act and of winning the run; when ending a turn it may also predict its HP at the start of the next turn and at the end of the fight. Same runs as Table 1. Pred. and Obs. are the mean predicted and observed rates (a floor's act counts as survived if the run went on to clear it; a run is won at the Heart). A prediction is easier the closer it is made to the moment it resolves, and stronger models make more of theirs close to an act's end, so the Brier scores (0 is perfect; always answering 50% scores 0.25), the AUC (0.5 is chance; – where every act had the same outcome) and the fight-end HP error are adjusted for that distance: each is scored over the same mix of distances, pooled over all models (Fig. 14). HP: mean absolute error and signed bias (predicted minus actual; positive is optimistic) in hit points, with a prediction made in the fight that killed the player scored against 0. Next-turn HP is always one turn ahead, and close to arithmetic since enemy intents show the incoming damage. **Bold** and underline mark the best and second-best scores among models with all 10 seeds.

			Act survival				Run win			HP next turn		HP at fight end	
	Agent	Runs	Pred. %	Obs. %	Brier	AUC	Pred. %	Obs. %	Brier	MAE	Bias	MAE	Bias
	Qwen3.8 27B (xhigh)	10	51	23	0.24	0.69	21	0	0.05	1.1	−0.1	10.3	+4.5
	Qwen3.6 35B-A3B (xhigh)	10	55	0	0.32	–	31	0	0.11	3.1	−1.2	12.3	+8.2
	Qwen3.5 122B-A10B (xhigh)	10	63	0	0.42	–	33	0	0.12	2.8	+0.6	17.7	+17.2
	Claude Fable 5.1 (max)	10	59	76	0.18	0.80	16	0	0.05	0.1	0.0	7.3	+0.3
	Claude Opus 5 (max)	10	62	68	**0.15**	0.85	14	0	0.04	0.2	−0.2	8.6	+0.1
	Claude Sonnet 5 (max)	10	68	45	0.30	0.58	34	0	0.12	0.4	−0.1	9.5	−2.9
	Claude Haiku 4.5 (12k)	10	42	15	0.25	0.39	19	0	0.05	3.0	−0.6	8.9	+1.1
	DeepSeek V4.1 Flash (max)	10	37	30	0.25	0.39	9	0	**0.01**	0.6	−0.1	9.9	+3.0
	Gemini 3.8 Flash (high)	10	86	44	0.43	0.49	49	0	0.26	0.1	0.0	10.9	+10.6
	Gemma 4 E4B	10	76	0	0.59	–	61	0	0.39	4.5	−0.3	13.4	+11.7
	Muse Spark 1.3 (xhigh)	10	54	54	0.23	0.63	13	0	0.02	0.6	−0.2	8.3	+1.1
	Nemotron 3 Nano Omni	10	72	0	0.53	–	55	0	0.33	4.0	+0.1	13.6	+10.6
	GPT-6 Astra (max)	10	74	69	0.16	**0.88**	22	0	0.07	**0.0**	0.0	**6.5**	+3.7
	GPT-5.6 Sol (max)	10	71	63	0.22	0.68	20	20	0.10	**0.0**	0.0	14.0	+13.1
	GPT-5.6 Terra (max)	10	50	47	0.28	0.49	12	0	0.02	0.4	−0.4	13.8	+13.7
	GPT-5.6 Luna (max)	10	64	12	0.37	0.64	21	0	0.05	0.2	−0.1	11.3	+6.9
	gpt-oss-120b (high)	10	65	17	0.40	0.56	33	0	0.13	1.2	+0.1	16.1	+14.8
	Grok 4.6 (xhigh)	10	41	33	0.24	0.54	9	0	**0.01**	0.3	−0.1	7.0	+2.1
	GLM-5.3 (max)	10	64	46	0.27	0.59	25	0	0.07	0.5	0.0	9.0	+3.9
	GLM-5.3 Flash (max)	10	58	46	0.26	0.54	24	0	0.06	1.0	−0.4	8.7	−0.1

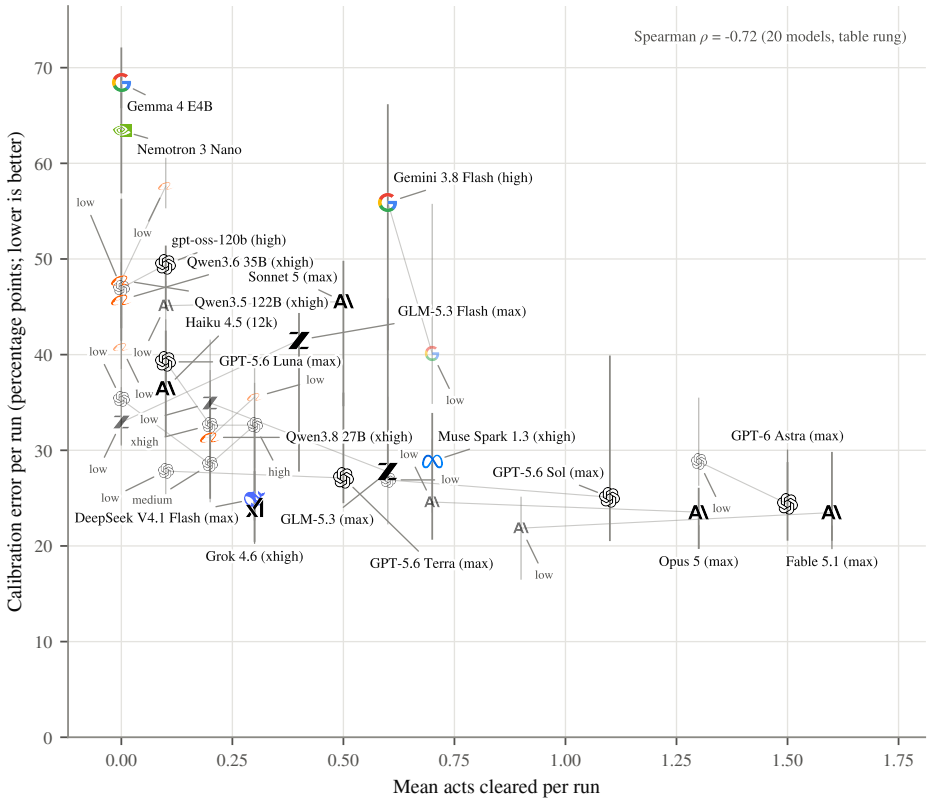


Figure 13: Calibration against play, for each LLM agent in Table 1. Calibration error is the reliability term (Murphy, 1973) of the Brier score (Brier, 1950), in percentage points, over ten bins of stated probability, for the two forecasts made at the end of every floor (surviving the act, winning the run), scored separately and averaged. It is computed per run, not adjusted for how far ahead a prediction was made (Table 12); points are the median over a model's runs and bars the interquartile range. Lower effort levels are drawn smaller and paler, joined to the same model. The Spearman correlation is over the levels in Table 1, one per model.

Table 13: Which forecasts track play. The Spearman correlation, over the 20 models of Table 1, between mean acts cleared and the error of each forecast, pooled over each model's runs: mean absolute error for the two HP forecasts, made when ending a turn, and the calibration error of Fig. 13 for the two probability forecasts, made at the end of every floor. Negative means the models that forecast better play better. Every model plays Act 1, so the Act 1 column compares all of them on forecasts about the same part of the game, while the first column also counts later acts, which only the stronger models reach and where hits are larger. Next-turn HP is close to arithmetic once enemy intents are shown, so its error measures how well a model reads the fight rather than what it knows of its own play. Almost no run is won (1.5% of run-win forecasts), so the run-win error mostly measures how much chance a model gives itself. The bottom rows correlate the scores plotted in the two figures, which combine two forecasts each and are medians over runs.

Forecast	Made at	All predictions	Act 1 only
HP at the start of the next turn	ending a turn	−0.84	−0.86
HP at the end of the fight	ending a turn	−0.70	−0.82
Surviving the act	end of a floor	−0.84	−0.77
Winning the run	end of a floor	−0.45	−0.58
Both HP forecasts (Fig. 15)		−0.92	–
Both probabilities (Fig. 13)		−0.72	–

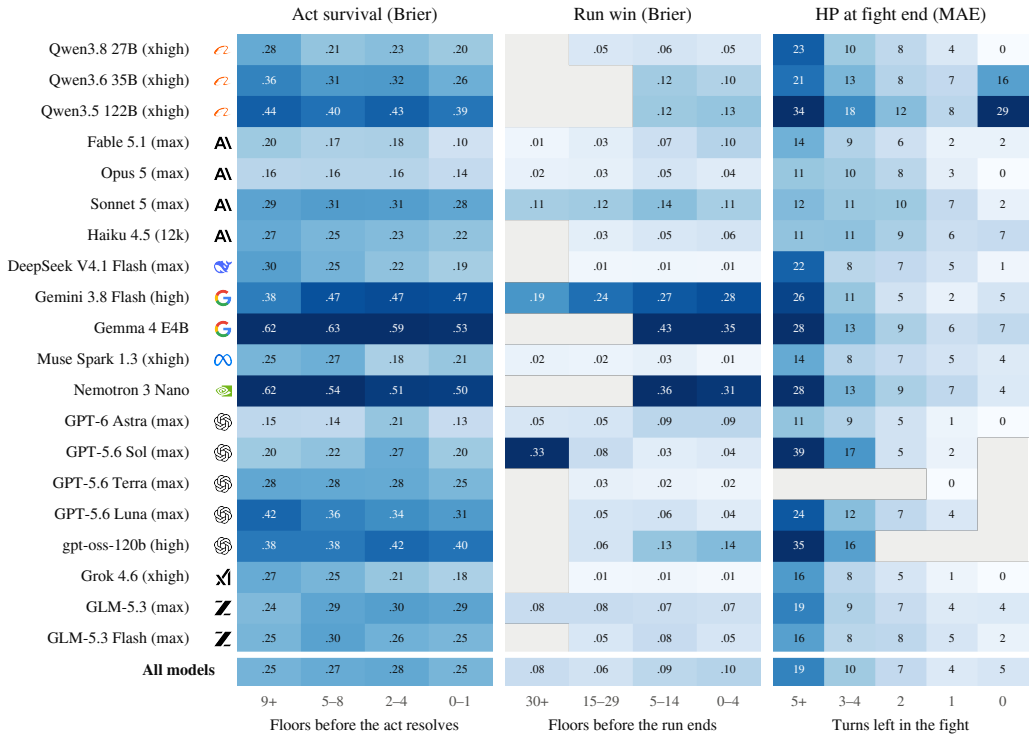


Figure 14: Prediction error against distance from the outcome, for the runs of Table 1. Left: Brier score of the act-survival prediction against the number of floors before the act resolves (at its end if the run cleared it, otherwise where the run died). Middle: the same for the run-win prediction, against floors before the run ends. Right: mean absolute error, in HP, of the predicted HP at the end of the fight against the number of turns still to be played in it. Darker is worse, on a scale per panel; blank cells have fewer than 5 predictions. The bottom row pools all models, and its mix of distances is the one each model is scored over in Table 12. Fight-end predictions sharpen steeply as the fight ends; act-survival predictions barely sharpen before the act resolves.

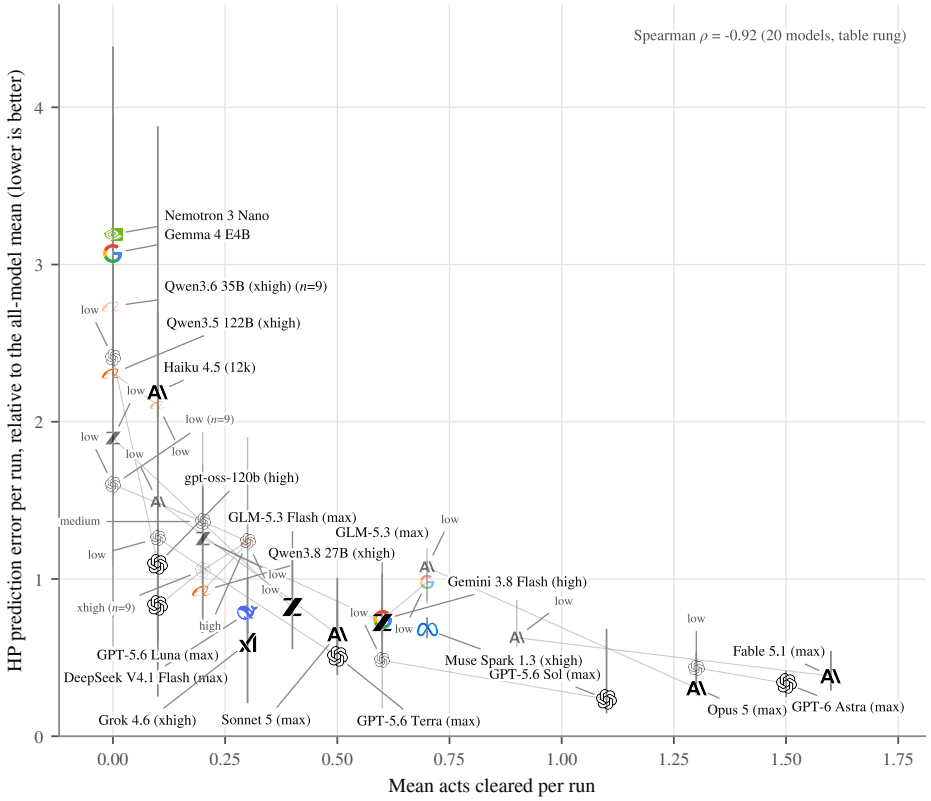


Figure 15: As Fig. 13, for the forecasts made within a floor: the HP a model expects at the start of its next turn and at the end of the fight, both optional when ending a turn. Each is scored by its mean absolute error in HP, divided by that forecast's mean error over the predictions of every model in Table 1 (so 1 is average), and the two are averaged for each run. The division matters: fight-end errors are about ten times next-turn ones, so without it a run that made only next-turn predictions would look excellent. A run with no HP predictions has no score. Within-floor error tracks play more closely than end-of-floor calibration (Table 13).

I FURTHER ANALYSES

The analyses below support the Results and Discussion (Sections 4 and 5). Unless stated otherwise, all are over the runs of Table 1. Correlations across models are tested by permutation, and within-run contrasts by exact sign tests.

Time horizon. Slay the Spire interleaves four kinds of decision, on four time scales, over one irreversible pool of resources: tactics within a turn (seconds), pricing HP within a fight (minutes), building a deck within an act (10–20 minutes) and setting a risk posture over a run (an hour). Playing it over such spans also needs memory management: deciding what to keep across floors, where each floor is a fresh conversation, and across runs, where the environment starts again. The time scales are the ones on which agent ability is now charted: Kwa et al. (2026) measure an agent by the length of task, in human working time, that it completes half the time, and find it doubling about every seven months on software tasks. A run here is an hour of human play, of the order of the horizons they report for frontier models, but it is one task made of decisions on every shorter scale, so the benchmark can show which scale a model fails on rather than only how long a task it survives. Measured this way (Table 16), every model succeeds on at least 87% of its turns and 77% of its fights, and falls below 50% between a fight and a run: for all but the top four at the act, for those four (53–67% of acts) only at the run. The fitted 50% horizons run from about one minute of human play (Gemma 4 E4B) to about 19 minutes (Claude Fable 5.1), on act and run times taken from Mega Crit's telemetry (players who complete Acts 1, 2 and 3 take about 9, 12 and 20 minutes over them, and a run that kills the Heart takes a median of 50 minutes); they order the models as acts cleared does, largely by construction, since the act is the only scale near 50%. What the measure does show is that the fall is a step rather than a slope. Of the 20 models, 18 win 93–100% of their ordinary Act 1 fights, but boss fights, won 0–81% of the time, decide where each model stops, and human players at this level show the same step, clearing 43% of the acts they enter. The difficulty is therefore not a limit on how long a model can keep going, as a time horizon on a smooth task distribution would suggest, but rare, high-stakes decisions inside a long task, whose preparation has to begin floors earlier.

Public benchmarks and reasoning effort. The 30 model and effort points of Fig. 2b are 20 models at their highest reasoning effort and 10 of them again at low effort. Acts cleared tracks the Intelligence Index about as closely within each rung as across both ($r = 0.80$ at the highest effort, 0.91 at low). Within a model the Index is a poorer guide. Lowering the effort costs acts for 8 of the 10 models with an Artificial Analysis entry at both levels (Table 22), but the size of the loss does not follow the size of the drop on the Index (Spearman $\rho = 0.06$): Gemini 3.8 Flash and Qwen3.8 27B each lose about 7.5 Index points and clear slightly more acts at low effort, while Claude Fable 5.1 loses 6.5 and clears 0.7 fewer acts. SpireBench ranks models much as the public suite does, but measures something the suite does not already capture. It is what the long-horizon literature would predict: if success on a long task compounds per-step reliability (Sinha et al., 2026), small differences on short evaluations should widen into large ones over a run, and the steep rise of acts cleared with the Index (Fig. 2b) is consistent with that.

Agentic residuals. Do models that score well on public agentic evaluations nonetheless fail at the game? Regressing mean acts cleared on each agentic evaluation Artificial Analysis reports, one point per model at its highest effort (Table 17), Grok 4.6, GLM-5.3 Flash and GPT-5.6 Luna clear fewer acts than all four predict, and Claude Fable 5.1, GPT-6 Astra and Claude Opus 5 more. The shortfall is not specific to agentic skill, however: the same models fall short of their scores on Humanity's Last Exam and GPQA Diamond by about as much (the mean residuals correlate at $r = 0.92$). Only one agentic evaluation, Terminal-Bench 4.0, predicts the game better than Humanity's Last Exam ($r = 0.89$ against 0.81) and adds information once it is controlled for (partial $r = 0.64$, permutation $p = 0.006$, $n = 18$); τ-Bench Banking, Terminal-Bench 2.1 and GDPval predict it less well and add nothing.

Harness-ended runs and our own design choices. The Orak comparison runs Orak at each model's own reasoning setting; its 86% saving in tokens per floor, 33% in cost once our cached prefix is priced at its own rate, comes largely from its rule bot playing every non-combat floor without a model call and from its shorter prompts, which is why we compare models on cost fronts (Fig. 2a) rather than on score alone. A harness can also end runs itself, and those endings are charged to the model. A

harness that supports most of a game but not all of it stops a run when it meets what it cannot handle, and every per-run figure then records that as the model dying there. On these seeds Orak's harness ended 18 of its 142 launches itself, by crashing on an empty model reply, looping without progress or stalling at an event, and its own failure ended 14 of the 61 seeds that finished; ours ended none of its launches on these arms, and 6 of the 2,422 it made across every arm (Table 27). We count a run a harness ended as a finished run lost by that harness, and report these endings separately, since a benchmark that did not would score the harness's gaps as the model's mistakes. That is the practical case for a harness that plays the whole game. Of our own choices, the continuous conversation is compared on the same floors; a fresh conversation per floor, with the run's notes and history carried forward, avoids both a long context used poorly (Liu et al., 2024a) and earlier mistakes kept in view (Sinha et al., 2026). Vending-Bench finds long-running agents derailing in ways unrelated to when the context fills (Backlund & Petersson, 2025); our comparison is narrower, holding the task fixed and changing only how much of its history the model is shown. One action per response costs 0.07 acts for 3.0 more calls and 23% more cost per floor, and when choices are also made by number rather than name it costs no acts ($+0.00$).

What the memory banks hold. Carrying notes forward verbatim, the one memory arm that does better than none, gains $+0.06$ acts, $+1.6$ floors and $+5$ points of Act 1 win rate (Table 24); that is suggestive rather than established, but it points the same way as the rest. Our result agrees with the line of work on unverified self-correction and memory propagation (Huang et al., 2024; Xiong et al., 2025) rather than contradicting the reports of improvement, and the problem it exposes only shows where episodes are long, stochastic and repeated. Reading the banks shows why (Table 19). Of the memories the models carry out of their tenth run, 59% describe how the game works, 29% are advice, 9% are rules stated without condition and 2% recount a single run, and 17% are false. Most are the model's own generalizations from one run. Opus 5 also records that another boss "NEVER SUMMONS AGAIN" and generalizes it to "Most summon exactly ONCE"; Muse Spark decides on its second seed that an area attack "does NOT scale with Strength", misreading a damage figure that already included its Strength, and never revisits it. Rules are the error-prone kind, 78% of them false against 9% of statements of fact, and models whose banks hold more errors of their own clear fewer acts (Spearman $\rho = -0.67$, permutation $p = 0.002$, $n = 20$). How emphatically a memory is phrased does not matter ($\rho = -0.03$): Claude Fable 5.1 and Opus 5 write in capitals as often as Claude Haiku 4.5, but what they write is largely true. Nothing prunes the errors. Over the 200 runs the models delete 30 memories, and wrong memories are rewritten about as often as correct ones without being corrected. DeepSeek V4.1 Flash records correctly that the act transition restores most of the HP it has lost, overwrites it two seeds later with "the post-boss heal is small", and then drops the subject. Without a check against outcomes, consolidation keeps whatever a single run suggested. Whether the models learn from their banks across seeds at all is in Fig. 18.

Reskin cost by model family. We find no evidence that the reskin's cost follows model family, and so cannot tie it to particular training corpora: a permutation test on developer labels finds no more variation between families than within them (11 models from 7 developers; $p = 0.29$ for acts, $p = 0.12$ for floors), and within a family the cost differs sharply, with GLM-5.3 Flash losing 7.5 floors against GLM-5.3's 2.3, and GPT-5.6 Luna 4.3 against none for gpt-oss-120b. The cost appears only in models that clear 0.6–0.8 acts with the real names (Spearman $\rho = -0.67$, permutation $p = 0.029$, between acts lost and acts cleared with real names), which is partly mechanical, since a model that clears no acts has none to lose. The reskin removes knowledge that only the stronger players were using. Where nonsense predicates alone collapse planning in Mystery Blocksworld (Valmeekam et al., 2023), here nonsense names cost least; a familiar name pulls a model towards what that name usually means, which helps when it is right and misleads when it belongs to something else (McCoy et al., 2024).

Recognition of the game under the reskin. Since a model's own text is never renamed (Sec. 3), we can measure when it recognizes what it is playing (Table 20). In 125 of the 190 themed runs of Table 1 with full logs (Gemma 4 E4B has none), from 18 of 19 models, the model names Slay the Spire or a real entity that the reskin replaced, almost always in its reasoning and usually on the first seed. This ranges from borrowing the real game's rules, as Nemotron 3 Nano Omni does in every run ("Usually in Slay the Spire, 'Coiled' is a buff that gives guard when you take damage"), to identifying entities from their behaviour, as Claude Fable 5.1 does with the Act 1 boss: "This

250 HP figure actually matches Slay the Spire's Guardian at Ascension 9+ … I'm recalling its exact moveset". Only Muse Spark never does. Recognition rarely reaches the memory bank (15 of 200 themed runs, 10 of them Fable's), and models recognize the game on their first seed, before any memory exists, about as often as later. Nor does it buy back what the reskin costs: across the ten models with both arms, the share of runs that recognize the game is unrelated to the cost (Spearman $\rho = 0.18$, permutation $p = 0.62$), and within a model, runs that recognize it in their first six floors clear no more acts than those that do not ($+0.06$, $p = 0.85$, 170 runs). Knowing which game this is does not recover knowledge that was keyed to its names. Changing the vocabulary barely changes recognition. For the eight models run under every mapping, 56 of 80 themed runs recognize the game, against 44 of 79 with nonsense names and 45 of 73 with the permuted themed ones. Gemini 3.8 Flash at low effort goes further under nonsense names, decoding them back to the real ones in every run and correctly about nine times in ten, and it scores its best result there. When real names are swapped between entities, six of the eight models conclude they are playing a modded Slay the Spire (GLM-5.3 Flash: "The mod card-identity trap: I played †Defend thinking it gave channel [block] (base-StS instinct)"), and their appeals to the real game mostly correct the borrowed meaning rather than act on it; runs that make such appeals clear no fewer acts ($+0.05$, $p = 0.71$, 80 runs).

Deck building does not fail by accumulation. In Act 1 almost any card improves a starting deck, but by Act 3 a marginal card dilutes the deck that has to beat the boss, so the right policy for card rewards reverses during a run. The models do reverse it on average (Table 18): their skip rate rises from 0.09 in Act 1 to 0.25 and 0.42, as steeply as human players' (0.21, 0.39 and 0.53 over every Ascension 20 Ironclad run in the telemetry), and within runs as well as between them (of the runs whose rate changes, 35 rise and 8 fall; exact sign test $p = 4 \times 10^{-5}$). But 41% of the model runs that reach a second act never skip a card at all, against 10% of human runs, and the deepest models are the most reluctant to skip early (3% in Act 1, against 18% for players with over 1000 hours). Neither deck size at a matched floor, within a model, nor how much a model adapts its skip rate predicts how many acts it clears (Spearman $\rho = 0.04$, $p = 0.91$, over the 11 models with enough decisions after Act 1). Models do not lose by filling their decks; many never revise the policy they start with, and among those that do, revising it more does not carry them further.

Where models go wrong. Reading the runs of Table 1 shows recurring failures (excerpts in Table 21). Weaker models spend reasoning on routine calls: the end-of-floor forecast asks for two probabilities, yet Claude Haiku 4.5, gpt-oss-120b and Grok 4.6 spend more than 1,000 reasoning tokens on 58–83% of theirs, against at most 9% for the strongest models (Spearman $\rho = -0.49$ between the median spent and mean acts cleared). Grok 4.6's reasoning often settles on one action while its tool call takes another: in 20 of the 68 decisions where its reasoning ends by naming a card or an option (29%), the call differs, seven times skipping a card reward it had just said it would take, and the prose it writes beside the call agrees with the call rather than the reasoning. gpt-oss-120b does so in 2 of 31 such decisions, and none of the other 16 models in 1,352. Most models sometimes lose track of their own turn: 15 of the 20 end a turn with a different amount of energy from the one they declared (Table 15). Within a run it is the strongest models that over-write their notes: Claude Opus 5 writes 143 notes per run at a median of 192 words, and 23–42% of the notes of Gemini 3.8 Flash, GPT-6 Astra, Claude Fable 5.1, Muse Spark and GLM-5.3 are turn-by-turn accounts of a single fight, while the weakest write a few short, generic notes. What the harness leaves unstated is learnt unevenly. It gives Vulnerable's 50% but not that damage is rounded down once every modifier is applied; gpt-oss-120b and Nemotron 3 Nano Omni work the rounding out afresh in all ten of their runs, Qwen3.6 and Claude Haiku 4.5 in nine, and none of them ever writes it down, while eleven models keep it in their memory bank. Of the 16 models that clear Act 1, nine end with the fact that the act transition restores most lost HP in their bank; Muse Spark and Grok never record it and weigh their low HP when choosing the boss relic, GPT-5.6 Terra credits the heal to the relic it chose, and DeepSeek V4.1 Flash records it and later overwrites it with the claim that the heal is small. End-of-run consolidation accumulates rather than curates: over the 200 runs the models create 1,404 memories (two of them left by Gemma 4 E4B's crashed first attempt), rewrite 1,080, re-save another 863 unchanged and delete 30, and 15 of the 20 never delete one; what the banks end up holding is classified in Table 19. The miscounted turns echo what is known of long executions: a slip early in a turn is carried into the rest of it, as errors in context make further errors more likely (Sinha et al., 2026). The notes and memories are the memory failure above, seen from inside a run: nothing tells the model which of its observations were worth keeping, so it keeps the wrong ones (Xiong et al., 2025).

Table 14: SpireBench against Artificial Analysis's public benchmarks, over the 20 LLM agents of Table 1 with all 10 seeds at their highest reasoning effort, and 10 of them again at low effort (30 points), each matched to the Artificial Analysis entry for the reasoning setting it ran at (scores retrieved 2026-09-24). A low rung is included where it has all 10 seeds and Artificial Analysis lists that setting. *Left*: correlation of each benchmark with mean acts cleared, over both rungs together and within each rung (n is the number of points with that evaluation). *Right*: how substitutable each benchmark is by the rest of the suite, with SpireBench added as one more row: its strongest correlation with any other single evaluation, its mean correlation with the others, and the share of its variance across points that the first principal component of the others predicts for a model left out of the fit, both rungs of it together (R^2_{LOO}; negative is worse than predicting the mean). The Intelligence Index is a composite of evaluations like these, so it is left out of the right-hand block, which uses the points with every evaluation (25).

	Correlation with SpireBench					Substitutability		
	Both rungs			Pearson r by rung				
Benchmark	n	Pearson r	Spearman ρ	Highest	Low	max r	mean r	R^2_{LOO}
AA Intelligence Index	30	0.82	0.89	0.80	0.91	–	–	–
GPQA Diamond	28	0.58	0.80	0.56	0.74	0.86	0.70	0.51
Humanity's Last Exam	30	0.82	0.89	0.81	0.87	0.92	0.83	0.85
SciCode	27	0.70	0.81	0.73	0.60	0.87	0.79	0.75
Long-context reasoning	30	0.52	0.66	0.47	0.70	0.74	0.64	0.29
CritPt	30	0.82	0.83	0.84	0.82	0.92	0.78	0.70
AA-Omniscience	30	0.83	0.91	0.82	0.87	0.86	0.77	0.70
SpireBench (acts cleared)	25	–	–	–	–	0.83	0.71	0.50

Calibration and distance to the outcome. The act-survival forecast barely sharpens as the act nears its end, while the forecast of HP at the end of a fight sharpens steeply as the fight ends (Fig. 14). Calibration, like play, degrades with the distance to the outcome, which matters for an agent that must decide how much risk to take on a long task.

Which part of the game. Handing one part of the game to a model, with Bottled AI playing the rest, shows where each model falls short of a symbolic player (Table 25). Combat is where the weaker open models lose: handing it over costs Nemotron 3 Nano Omni 5.1 floors, Qwen3.5 4.1, gpt-oss-120b 3.4 and Qwen3.6 3.2, and costs GPT-5.6 Luna and Qwen3.8 about two floors. The baseline policies lose most in combat too (4.3 to 5.6 floors), so combat is where play is decided on these seeds. By contrast Gemini 3.8 Flash, at either effort, loses under a floor in every part, and for most models reward choices, events and the route cost little. Pooled over the models, combat costs 2.2 floors and no other part costs more than one. Bottled AI itself clears only 0.20 acts on these seeds, so the comparison is made in floors.

Table 15: Tool-call errors per 100 model calls, for the LLM agents of Table 1. *No usable call*: a response in which no tool call could be executed, because every call was malformed or refused or none could be parsed; the harness asks again. *Refused call*: one call refused before it reached the game, as illegal on the current screen or with invalid arguments; the model is told why. *Energy mismatch*: a turn ended with a different amount of energy left than the model declared when ending it, a sign that it lost track of its own turn. *Recovered from prose*: a call the model wrote as text rather than through the tool interface, which the harness recovered and executed. *Errors* sums the first three, which are the ones the model is told about. Rates are totals over the cohort divided by its calls.

	Agent	Calls/run	No usable call	Refused call	Energy mismatch	Recovered from prose	Errors
	Qwen3.8 27B (xhigh)	124	0.24	0.00	0.73	0.00	1.0
	Qwen3.6 35B-A3B (xhigh)	104	0.48	0.38	0.58	0.38	1.4
	Qwen3.5 122B-A10B (xhigh)	77	0.39	0.39	1.29	0.13	2.1
	Claude Fable 5.1 (max)	268	0.00	0.15	0.22	0.00	0.4
	Claude Opus 5 (max)	301	0.03	0.10	0.00	0.00	0.1
	Claude Sonnet 5 (max)	159	0.06	0.06	0.06	0.00	0.2
	Claude Haiku 4.5 (12k)	101	0.00	0.00	1.09	0.00	1.1
	DeepSeek V4.1 Flash (max)	119	0.08	0.08	0.00	0.00	0.2
	Gemini 3.8 Flash (high)	208	0.05	0.00	0.00	0.00	**0.0**
	Gemma 4 E4B	82	0.85	1.83	1.10	0.12	3.8
	Muse Spark 1.3 (xhigh)	228	0.04	0.00	0.04	0.00	0.1
	Nemotron 3 Nano Omni	71	0.28	0.42	0.99	4.38	1.7
	GPT-6 Astra (max)	546	0.02	0.22	0.02	0.00	0.3
	GPT-5.6 Sol (max)	331	0.06	0.15	0.00	0.00	0.2
	GPT-5.6 Terra (max)	230	0.00	0.13	0.00	0.00	0.1
	GPT-5.6 Luna (max)	168	0.00	0.06	0.06	0.00	0.1
	gpt-oss-120b (high)	175	0.23	0.00	1.77	8.64	2.0
	Grok 4.6 (xhigh)	171	1.29	0.00	0.06	0.00	1.3
	GLM-5.3 (max)	160	0.00	0.00	0.25	0.00	0.3
	GLM-5.3 Flash (max)	107	0.09	0.00	0.19	0.00	0.3

Table 16: Success at each decision scale of Fig. 1, for the LLM agents of Table 1 (10 runs each). Turn is the share of combat turns without a rejected action, format error or energy mismatch (a lower bound, since every error is charged to a different turn). Combat is the share of fights won; A1 hallway is the share of ordinary Act 1 fights (not elites or the boss) won, and Boss the share of boss fights won (superscript: boss fights played). Act is the share of acts cleared among acts entered, pooled over Acts 1–3. Run is the share of runs that cleared Act 3. h_{50} is the human time at which a logistic in log time, fitted with each scale weighted equally, crosses 50%, with each scale placed at a human time from Mega Crit's telemetry: 10 s for a turn, 10 s per turn of an average fight, the mean of the median times to complete Acts 1–3 (8.8, 11.9 and 20.3 min, taken from players who completed each act), and the median run that kills the Heart (50 min). Human rows come from Appx. D and have no turn or combat data.

		Success (%)							
	Agent	Turn	Combat	A1 hallway	Boss	Act	Run	h_{50}	Acts
	Claude Fable 5.1 (max)	98.6	95	100	80^{20}	67	20	19 min	1.60
	GPT-6 Astra (max)	98.3	95	98	75^{20}	62	10	14 min	1.50
	Claude Opus 5 (max)	99.3	94	100	81^{16}	57	0	11 min	1.30
	GPT-5.6 Sol (max)	98.9	94	100	73^{15}	53	10	11 min	1.10
	Muse Spark 1.3 (xhigh)	99.6	91	96	78^{9}	41	0	7 min	0.70
	Gemini 3.8 Flash (high)	99.8	92	100	50^{12}	38	0	7 min	0.60
	GLM-5.3 (max)	99.1	91	96	55^{11}	38	0	6 min	0.60
	GPT-5.6 Terra (max)	99.3	90	100	56^{9}	33	0	6 min	0.50
	Claude Sonnet 5 (max)	99.4	90	100	50^{10}	33	0	6 min	0.50
	GLM-5.3 Flash (max)	99.1	88	98	67^{6}	29	0	5 min	0.40
	DeepSeek V4.1 Flash (max)	99.4	89	100	33^{9}	23	0	4 min	0.30
	Grok 4.6 (xhigh)	95.7	88	98	50^{6}	23	0	4 min	0.30
	Qwen3.8 27B (xhigh)	96.6	88	98	29^{7}	17	0	3 min	0.20
	gpt-oss-120b (high)	87.5	83	96	33^{3}	9	0	2 min	0.10
	GPT-5.6 Luna (max)	99.4	87	98	14^{7}	9	0	3 min	0.10
	Claude Haiku 4.5 (12k)	96.4	84	98	25^{4}	9	0	2 min	0.10
	Qwen3.5 122B-A10B (xhigh)	94.0	81	93	0^{1}	0	0	85 s	0.00
	Qwen3.6 35B-A3B (xhigh)	93.9	84	94	0^{2}	0	0	2 min	0.00
	Nemotron 3 Nano Omni	94.6	77	78	–	0	0	77 s	0.00
	Gemma 4 E4B	87.1	78	89	0^{2}	0	0	74 s	0.00
	Human, all A20 Ironclad (telemetry)	–	–	–	–	43	7	–	0.73
	Human, 1000+ h players (telemetry)	–	–	–	–	49	11	–	0.92
	Human, Baalorlord, complete log	–	–	–	–	89	70	–	2.98

Table 17: Which models play below, or above, what their public agentic scores predict. For each Artificial Analysis evaluation, mean acts cleared (Table 1, highest reasoning effort, all 10 seeds) is regressed on the evaluation score across models, one point per model; cells are internally studentized residuals (z; negative means fewer acts than the evaluation predicts). The top rows give each evaluation's correlation with acts cleared, with two-sided permutation p (200,000 permutations) and n models. Models are sorted by their mean residual over the agentic evaluations (left block). Artificial Analysis scores retrieved 2026-09-24.

		Agentic					Reasoning	
Model	Acts	τ-Bench Banking	Terminal-Bench 2.1	Terminal-Bench 4.0	GDPval (normalized)	Mean z	Humanity's Last Exam	GPQA Diamond
Pearson r		0.59	0.65	0.89	0.67		0.81	0.56
p		0.013	0.004	<0.001	0.002		<0.001	0.014
n		17	19	18	20		20	19
DeepSeek V4.1 Flash	0.30	–	–	−1.41	−1.16	−1.29	−0.91	–
GPT-5.6 Luna	0.10	−0.87	−1.41	−0.77	−1.36	−1.10	−1.59	−1.21
GLM-5.3 Flash	0.40	−0.97	−0.75	−1.60	−0.99	−1.07	−0.63	−0.52
Grok 4.6	0.30	−1.00	−1.12	−0.45	−1.24	−0.95	−1.32	−0.87
Qwen3.8 27B	0.20	−1.50	−1.12	0.29	−1.02	−0.84	−0.80	−0.95
GLM-5.3	0.60	−0.65	−0.23	−1.68	−0.46	−0.75	−0.16	−0.08
GPT-5.6 Terra	0.50	−0.37	−0.61	−1.42	−0.27	−0.67	−0.55	−0.35
Qwen3.5 122B-A10B	0.00	−0.34	−0.70	−0.03	−0.31	−0.34	−0.75	−1.17
Qwen3.6 35B-A3B	0.00	−0.02	−0.63	−0.03	−0.47	−0.29	−0.50	−1.09
Claude Sonnet 5	0.50	−0.22	−0.38	0.72	−0.31	−0.05	−0.41	−0.28
Gemini 3.8 Flash	0.60	−0.37	−0.34	0.59	0.03	−0.02	−0.63	−0.25
Claude Haiku 4.5	0.10	0.25	−0.35	0.42	0.15	0.12	0.89	−0.03
Muse Spark 1.3	0.70	−0.24	−0.02	1.33	−0.15	0.23	−0.26	0.04
gpt-oss-120b	0.10	0.05	0.19	0.42	0.42	0.27	0.06	−0.57
Nemotron 3 Nano Omni	0.00	–	0.56	–	0.36	0.46	1.07	1.00
Gemma 4 E4B	0.00	–	0.75	–	0.36	0.55	1.18	0.24
GPT-5.6 Sol	1.10	0.87	0.92	0.74	1.01	0.88	0.92	0.97
Claude Opus 5	1.30	1.45	1.40	0.71	1.31	1.22	1.16	1.48
GPT-6 Astra	1.50	1.96	1.94	0.59	2.17	1.66	1.87	1.82
Claude Fable 5.1	1.60	1.94	2.11	1.79	2.07	1.98	1.87	2.16

Table 18: Card-reward skip rate by act on the main-table cohorts (seeds 1–10, A20). A decision is a fight survived; it is a skip when no card joined the deck. n is the number of decisions in the act. Δ is the model's Act 2+ skip rate minus its Act 1 rate (pooled over its decisions), shown where it has at least 10 Act 2+ decisions. Humans are Mega Crit telemetry, A20 Ironclad, where a Singing Bowl pick also counts as a skip. The first- and random-option baselines never skip by construction; Bottled AI skips every card not on its strategy's wish list.

	Acts	Act 1		Act 2		Act 3		
Agent	(mean)	skip	n	skip	n	skip	n	Δ
LLM agents, deepest first								
Claude Fable 5.1	1.60	0.00	74	0.08	52	0.12	24	+0.09
GPT-6 Astra	1.50	0.05	77	0.32	60	0.57	35	+0.36
Claude Opus 5	1.30	0.01	81	0.20	51	0.20	15	+0.18
GPT-5.6 Sol	1.10	0.04	79	0.44	43	0.73	11	+0.47
Muse Spark 1.3	0.70	0.04	74	0.14	29	0.00	3	+0.08
GLM-5.3	0.60	0.03	71	0.16	25	–	0	+0.13
Gemini 3.8 Flash	0.60	0.14	85	0.57	21	0.86	7	+0.50
Claude Sonnet 5	0.50	0.01	73	0.00	17	0.00	1	−0.01
GPT-5.6 Terra	0.50	0.04	80	0.00	10	–	0	−0.04
GLM-5.3 Flash	0.40	0.02	63	0.25	12	–	0	+0.23
Grok 4.6	0.30	0.59	61	1.00	11	–	0	+0.41
DeepSeek V4.1 Flash	0.30	0.00	72	0.00	6	–	0	–
Qwen3.8 27B	0.20	0.00	64	0.25	8	–	0	–
Claude Haiku 4.5	0.10	0.54	50	0.00	2	–	0	–
GPT-5.6 Luna	0.10	0.06	64	0.00	3	–	0	–
gpt-oss-120b	0.10	0.02	47	0.00	1	–	0	–
Nemotron 3 Nano Omni	0.00	0.00	33	–	0	–	0	–
Gemma 4 E4B	0.00	0.00	34	–	0	–	0	–
Qwen3.6 35B-A3B	0.00	0.27	52	–	0	–	0	–
Qwen3.5 122B-A10B	0.00	0.11	44	–	0	–	0	–
All LLM runs (pooled)	0.49	0.09	1,278	0.25	351	0.42	96	+0.20
5 deepest LLMs (pooled)	–	0.03	385	0.24	235	0.39	88	+0.25
Human players (telemetry, own seeds)								
all A20 Ironclad	–	0.21	1,320,053	0.39	486,015	0.53	191,827	+0.22
1000+ hours	–	0.18	55,942	0.37	23,558	0.50	10,773	+0.23

Table 19: What the models keep in their memory bank after the tenth seed of Table 1, classified by hand. Bank is the final number of memories; Created, Rewritten and Deleted count consolidation operations over the ten runs (Rewritten excludes verbatim re-saves). Composition is the share of the final bank in each category (a random 30 per model where the bank is larger): a game fact describes how the game works, a heuristic advises, an absolute rule prescribes without condition, an observation recounts one run; Other is route plans and notes about the harness. Wrong is the share stating a false game fact or a rule that fails in common situations, and H. how many of the coded wrong memories trace to harness text that was itself wrong (a rest-site key label, a state diff across the two final bosses, and debuff expiry shown as an enemy's action); Absolute the share phrased as ALWAYS, NEVER or MUST anywhere; One run the share drawn from a single run. Models are ordered by mean acts cleared. Operations are read from the full run logs, except for Gemma 4 E4B, whose are read from the harness output log (its full logs are not available).

			Memories				Composition (%)								
	Agent	Acts	Bank	Created	Rewritten	Deleted	Fact	Heur.	Rule	Obs.	Other	Wrong (%)	H.	Absolute (%)	One run (%)
	Claude Fable 5.1	1.60	50	50	158	0	60	27	0	13	0	13	4	63	23
	GPT-6 Astra	1.50	149	149	110	0	87	13	0	0	0	0	–	3	53
	Claude Opus 5	1.30	82	83	228	1	33	63	3	0	0	17	1	60	33
	GPT-5.6 Sol	1.10	106	106	76	0	100	0	0	0	0	0	–	7	67
	Muse Spark 1.3	0.70	103	103	64	0	60	30	7	0	3	7	–	37	90
	GLM-5.3	0.60	44	44	57	0	63	20	10	0	7	20	–	30	47
	Gemini 3.8 Flash	0.60	79	79	35	0	93	7	0	0	0	3	–	10	63
	Claude Sonnet 5	0.50	108	108	54	0	73	27	0	0	0	3	–	10	57
	GPT-5.6 Terra	0.50	29	29	15	0	83	17	0	0	0	0	–	3	83
	GLM-5.3 Flash	0.40	74	76	67	2	87	10	3	0	0	20	1	20	63
	Grok 4.6	0.30	98	98	81	0	70	23	0	0	7	0	–	20	93
	DeepSeek V4.1 Flash	0.30	34	34	16	0	93	0	7	0	0	13	1	30	60
	Qwen3.8 27B	0.20	79	79	56	0	63	27	3	0	7	10	–	20	57
	Claude Haiku 4.5	0.10	69	80	3	11	7	37	50	7	0	60	–	60	100
	GPT-5.6 Luna	0.10	32	32	17	0	60	33	0	3	3	7	–	13	93
	gpt-oss-120b	0.10	79	79	2	0	27	70	3	0	0	23	–	3	3
	Nemotron 3 Nano Omni	0.00	66	66	4	0	20	43	30	0	7	57	–	27	10
	Gemma 4 E4B	0.00	24	24	7	0	0	92	8	0	0	12	–	8	4
	Qwen3.6 35B-A3B	0.00	13	27	25	14	62	15	23	0	0	54	–	46	38
	Qwen3.5 122B-A10B	0.00	56	58	5	2	27	30	37	7	0	40	–	53	87

Table 20: Recognition of the reskinned game, from each run's own text (reasoning, prose and tool-call arguments). Per arm: *Rec.* runs whose text names the real game or a real entity name the arm does not display; *Game* runs naming "Slay the Spire"/"StS"; *Names* median distinct real entity names per run. Themed also gives the first seed (floor) with recognition. The Permuted arm displays the title and real names (for the wrong entities), so there only undisplayed names and "StS" count. *Memory, None*: runs whose newly written memories use real names when the arm shows them (a sanity check; the real-names arm has no local logs). Gemma 4 E4B has no logs. – no data.

	Themed				Random			Permuted			Perm.+themed			Memory
Agent	Rec.	Game	Names	1st	Rec.	Game	Names	Rec.	Game	Names	Rec.	Game	Names	None
Qwen3.8 27B	10/10	10/10	0	S1 (F0)	10/10	10/10	0	10/10	10/10	0	3/3	3/3	0	–
Qwen3.6 35B-A3B	1/10	1/10	0	S9 (F3)	0/10	0/10	0	0/10	0/10	0	1/10	1/10	0	–
Qwen3.5 122B-A10B	2/10	2/10	0	S2 (F9)	1/10	1/10	0	0/10	0/10	0	4/10	4/10	0	10/10
Claude Fable 5.1	10/10	10/10	22	S1 (F0)	–	–	–	–	–	–	–	–	–	–
Claude Opus 5	9/10	7/10	2	S1 (F6)	–	–	–	–	–	–	–	–	–	–
Claude Sonnet 5	8/10	8/10	0	S1 (F3)	–	–	–	–	–	–	–	–	–	–
Claude Haiku 4.5	1/10	1/10	0	S1 (F0)	–	–	–	–	–	–	–	–	–	10/10
DeepSeek V4.1 Flash	10/10	10/10	1	S1 (F6)	–	–	–	–	–	–	–	–	–	10/10
Gemini 3.8 Flash	8/10	7/10	2	S1 (F7)	–	–	–	–	–	–	–	–	–	2/2
Gemini 3.8 Flash (low)	8/10	8/10	1	S1 (F0)	10/10	8/10	46	2/10	1/10	0	4/10	2/10	0	–
Gemma 4 E4B	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Muse Spark 1.3	0/10	0/10	0	–	–	–	–	–	–	–	–	–	–	–
Nemotron 3 Nano Omni	10/10	10/10	0	S1 (F5)	1/9	1/9	0	1/10	1/10	0	8/10	8/10	0	–
GPT-6 Astra	5/10	2/10	0	S1 (F18)	–	–	–	–	–	–	–	–	–	–
GPT-5.6 Sol	7/10	5/10	0	S1 (F0)	–	–	–	–	–	–	–	–	–	–
GPT-5.6 Terra	6/10	5/10	0	S3 (F12)	–	–	–	–	–	–	–	–	–	–
GPT-5.6 Luna	5/10	5/10	0	S1 (F9)	2/10	2/10	0	2/10	2/10	0	6/10	6/10	0	9/10
gpt-oss-120b	10/10	10/10	0	S1 (F1)	10/10	10/10	0	6/10	5/10	0	10/10	10/10	1	10/10
Grok 4.6	3/10	3/10	0	S1 (F22)	–	–	–	–	–	–	–	–	–	–
GLM-5.3	10/10	10/10	2	S1 (F2)	–	–	–	–	–	–	–	–	–	10/10
GLM-5.3 Flash	10/10	10/10	1	S1 (F0)	10/10	10/10	6	10/10	10/10	0	9/10	9/10	0	10/10

Table 21: Excerpts from the runs of Table 1, verbatim. Names are as the model saw them under the reskin, with the real game term in brackets. Excerpts are taken from the full run logs and cut at the ellipsis.

Failure	Model (seed, floor)	Excerpt
Deliberating over a routine forecast	Nemotron 3 Nano Omni (S3, 6)	"We need to estimate probabilities. The player is at camp after sleeping, HP 24/75 (32%). Actually after sleeping they healed 22 HP, so HP should be 24+22 = 46? Wait initial HP 24/75, sleeping heals 22 . . . "
Reasoning and action disagree	Grok 4.6 (S9, 10)	". . . I think Blood Temper [Armaments] is the better pick here. . . I'll go with Blood Temper." Then skips the reward, writing beside the call: "Skip Pulse and Smash as filler. . . "
Miscounted turn	GLM-5.3 (S5, 13)	"I miscounted energy! . . . My error: I planned Trip(2) + Slash(1) + Slash(1) = 4E but only have 3E! Oops."
Blow-by-blow note	Claude Fable 5.1 (S10, 18)	"CRIMSON WARLOCK [Chosen] solo 100 HP (Act 2 R18 opener), run 10 script: T1 Strong Debuff = Jinx [Hex] . . . → T2 Debuff = Fatigue [Weak] 3 on ME + Str 3 to itself . . . " (280 words)
Learnt, not written down	Nemotron 3 Nano Omni (S3, 2)	". . . player is Exposed [Vulnerable], taking 50% more damage, so effective damage = 7 * 1.5 = 10.5, rounded down? Probably 10 or 11."
Misattributed act heal	GPT-5.6 Terra (S5, end)	"Wanderer's Pack [Tiny House] is an excellent low-HP boss reward: . . . +5 Max HP/current HP . . . In this run it converted 16/75 to 65/80 before the next Act-2 fight . . . "
Correct memory overwritten	DeepSeek V4.1 Flash (S8, end)	"KEY NEW LESSON (this run): . . . the post-boss heal is small — I entered the act at 11/75" (its log that run: +44 HP, 16→60)
Rule from one run	Claude Haiku 4.5 (S9, end)	"Arriving at any combat-potential room (R13+) with <35% HP on thin deck without full Guard [Block] infrastructure is terminal. This run: R13 reached at 27/75 HP (36%) . . . "

J ABLATION RESULTS IN FULL

Figure 16 summarizes each ablation model by model. The pooled tables are below: reasoning effort (Table 22), reskin (Table 23), memory (Table 24) and which part of the game the model plays (Table 25); the harness comparison is Table 2 in the main text, followed by how the memory bank grows (Fig. 17), whether models learn from it (Fig. 18), the per-model results behind each table and how the Orak runs ended.

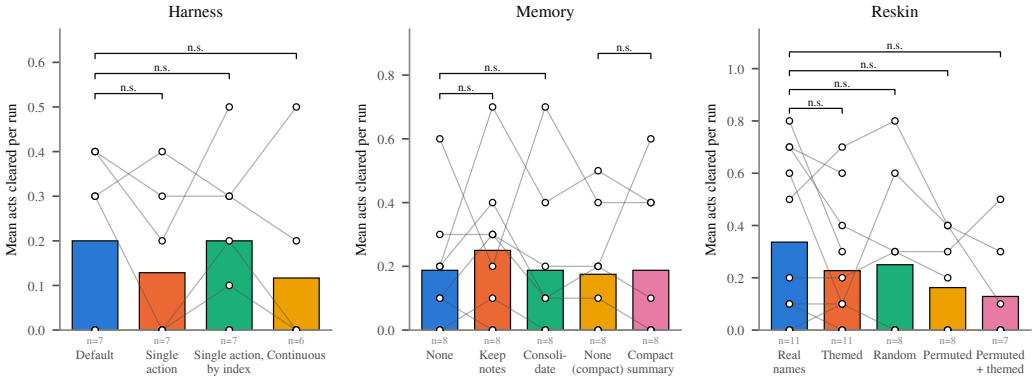


Figure 16: Each ablation, model by model. Each grey line is one LLM agent: its mean over the 10 seeds in each condition, shown only where it has all 10 seeds in both that condition and the condition's baseline, so every comparison is paired. Bars are the mean over those agents, and n beneath each bar is how many there are. Baselines are the first bar of each panel, except that the compact summary is read against no memory bank with compact notes. Panels show mean acts cleared per run; the pooled tables are Tables 2, 23 and 24. Brackets test each condition against its baseline with a paired sign-flip permutation test over agents (* $p < 0.05$; n.s. not significant). Orak is not shown: no model has all 10 Orak seeds scored; its paired comparison is in Table 2.

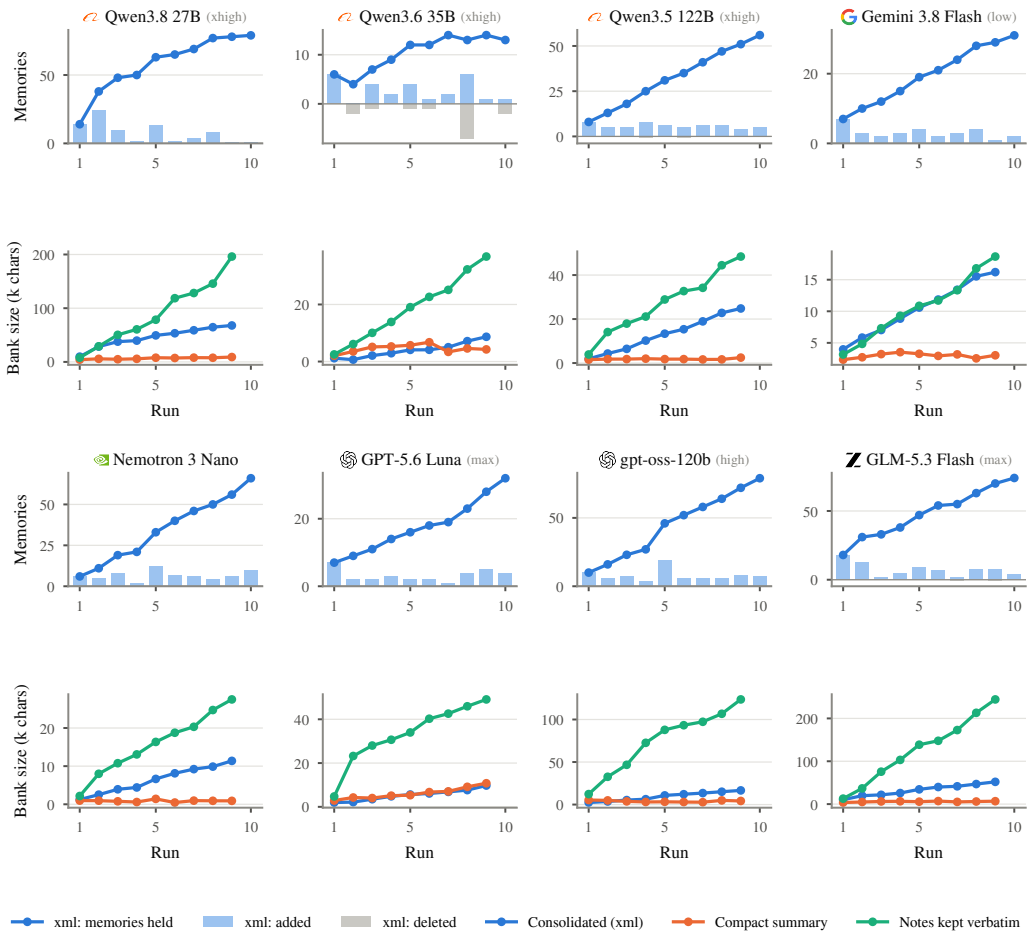


Figure 17: How the memory bank grows over a 10-run chain. For each model, top: the number of xml memories held after each run (line), with the memories added (blue, up) and deleted (grey, down) in that run's consolidation. Bottom: the size of the bank carried into the next run, in thousands of characters, for xml consolidation, the single compact summary, and notes kept verbatim. Size is measured exactly, from the system prompt of the following run, relative to the first run's empty bank; the bank after run 10 is never read, so that curve ends at run 9.

Table 22: Effect of reasoning effort. Each model at every effort level it was run at, A20 on the held-out seeds with the main-table configuration (themed reskin, xml notes, read-write memory, generic guidance). Columns as in Table 1, with Acts the mean number of acts cleared per run; Act 4 (a Heart kill) is not shown but counts in Acts. Only levels with all 10 seeds completed are shown, and only models with two or more such levels. **Bold** marks the best level per model on each column. †Self-hosted; priced at Artificial Analysis headline rates.

				Act win rate (%)			Acts	Floor			Thinking	Cost ($)	
	Agent	Effort	Seeds	1	2	3	(mean)	mean	median	[min–max]	(k/floor)	/floor	/run
	Qwen3.8 27B	low	10	**30**	0	0	**0.30**	**15.2**	**16.0**	[7–24]	**7.8**	**0.045**†	**0.74**†
		medium	10	20	0	0	0.20	12.8	10.5	[6–24]	9.3	0.055†	0.76†
		xhigh	10	20	0	0	0.20	14.9	**16.0**	[5–29]	18.1	0.091†	1.45†
	Qwen3.6 35B-A3B	low	10	0	0	0	0.00	8.7	7.0	[4–16]	4.1	0.083†	0.83†
		medium	10	0	0	0	0.00	10.0	**9.0**	[5–16]	**3.5**	**0.065**†	**0.71**†
		xhigh	10	0	0	0	0.00	**10.2**	8.0	[6–16]	4.2	0.078†	0.87†
	Qwen3.5 122B-A10B	low	10	10	0	0	0.10	9.4	**9.0**	[4–18]	3.8	0.074†	0.78†
		medium	10	0	0	0	0.00	9.5	7.0	[4–16]	3.8	0.077†	**0.76**†
		xhigh	10	0	0	0	0.00	**9.8**	**9.0**	[7–16]	**3.7**	**0.072**†	0.77†
	Claude Fable 5.1	low	10	70	20	0	0.90	27.5	27.5	[7–51]	**1.0**	**0.275**	**8.12**
		max	10	**90**	**50**	20	**1.60**	**36.2**	**39.5**	[7–52]	17.7	1.664	61.91
	Claude Opus 5	low	10	70	0	0	0.70	22.7	22.0	[10–33]	**0.9**	**0.204**	**4.84**
		max	10	**90**	40	0	**1.30**	**32.5**	**33.0**	[7–51]	21.3	1.743	60.62
	Claude Sonnet 5	low	10	10	0	0	0.10	12.3	13.5	[4–24]	**0.8**	**0.074**	**1.03**
		max	10	**40**	10	0	**0.50**	**19.3**	**16.0**	[7–36]	23.2	0.509	10.70
	Gemini 3.8 Flash	low	10	**60**	**10**	0	**0.70**	22.6	**22.0**	[5–50]	**2.5**	**0.063**	**1.56**
		high	10	50	**10**	0	0.60	**22.9**	18.0	[16–47]	13.7	0.117	2.89
	GPT-6 Astra	low	10	70	**60**	0	1.30	34.3	**40.0**	[7–51]	**0.1**	**0.383**	**13.66**
		max	10	**90**	50	10	**1.50**	**37.0**	**40.0**	[5–55]	3.3	0.834	33.09
	GPT-5.6 Sol	low	10	50	10	0	0.60	21.8	18.5	[13–50]	**0.5**	**0.133**	**3.15**
		max	10	**70**	**20**	10	**1.10**	**27.8**	**26.0**	[7–57]	4.0	0.272	8.39
	GPT-5.6 Terra	low	10	10	0	0	0.10	12.3	13.0	[5–20]	**0.6**	**0.060**	**0.81**
		max	10	**50**	0	0	**0.50**	**18.2**	**18.0**	[10–25]	5.0	0.149	2.91
	GPT-5.6 Luna	low	10	0	0	0	0.00	10.8	8.5	[6–16]	**0.4**	**0.005**	**0.06**
		medium	10	20	0	0	0.20	11.9	10.5	[5–21]	0.8	0.007	0.09
		high	10	**30**	0	0	**0.30**	**14.5**	**16.0**	[6–25]	1.6	0.008	0.14
		xhigh	10	20	0	0	0.20	14.4	14.5	[5–24]	2.7	0.010	0.16
		max	10	10	0	0	0.10	14.2	**16.0**	[7–24]	5.0	0.015	0.24
	gpt-oss-120b	low	10	0	0	0	0.00	7.3	6.5	[1–16]	**1.5**	**0.042**†	**0.36**†
		medium	10	0	0	0	0.00	9.2	**8.5**	[2–16]	4.3	0.047†	0.47†
		high	10	10	0	0	0.10	**10.4**	8.0	[5–21]	17.0	0.085†	0.91†
	GLM-5.3	low	10	20	0	0	0.20	14.3	16.0	[5–24]	**0.8**	**0.046**	**0.72**
		max	10	**60**	0	0	**0.60**	**22.1**	**22.0**	[5–33]	13.4	0.167	3.92
	GLM-5.3 Flash	low	10	0	0	0	0.00	11.1	10.0	[6–16]	**0.7**	**0.005**	**0.07**
		medium	10	20	0	0	0.20	12.9	13.5	[7–20]	2.7	0.010	0.14
		max	10	**40**	0	0	**0.40**	**15.6**	**16.0**	[5–27]	14.4	0.024	0.40

Table 23: Effect of the knowledge reskin. *None* shows the real game names and is the baseline; *Themed* is our default Lich's Lair reskin; *Random* replaces every name with pronounceable nonsense; *Permuted* swaps each name for another real game name of the same kind; *Perm.+themed* applies the same permutation in the themed vocabulary. All with xml notes, a read-write memory bank and generic guidance. Pooled over models: each method is compared with its baseline only on matched runs (the same model on the same seed; Pairs, over Agents models), and Acts, Floor and A1 (mean acts cleared, mean final floor and Act 1 win rate) are over those runs. Usage is compared on like-for-like floors only: the same model, seed, act and room type, and the same occurrence of that room type within the act, kept only where every method run on that model and seed completed it (Floors), so a method is neither credited with cheap early floors nor charged for expensive late ones. Calls and k tokens (input, including cached, plus output, in thousands) are means per floor; the token difference is the ratio of totals. Each Δ is the method minus its baseline. All runs are A20 on the held-out seeds, each model at its main-table reasoning setting. Models, by method: Themed: Qwen3.8 27B, Qwen3.6 35B-A3B, Qwen3.5 122B-A10B, Claude Haiku 4.5, DeepSeek V4.1 Flash, Gemini 3.8 Flash, Gemini 3.8 Flash (low), Nemotron 3 Nano Omni, GPT-5.6 Luna, gpt-oss-120b, GLM-5.3, GLM-5.3 Flash; Random, Permuted and Perm.+themed: Qwen3.8 27B, Qwen3.6 35B-A3B, Qwen3.5 122B-A10B, Gemini 3.8 Flash (low), Nemotron 3 Nano Omni, GPT-5.6 Luna, gpt-oss-120b, GLM-5.3 Flash. Per-model results are in Table 29.

			Performance (matched runs)						Usage per floor (matched floors)				
Method	Agents	Pairs	Acts	Δ	Floor	Δ	A1%	Δ	Floors	Calls	Δ	k tokens	Δ
None (baseline)	12	115	0.37	–	16.1	–	33	–	1121	7.7	–	229	–
Themed	12	115	0.23	−0.14	14.5	−1.6	22	−11	1121	7.8	+0.1	238	+4%
Random	8	80	0.25	−0.01	13.9	−0.1	21	−4	632	8.5	+0.2	234	+13%
Permuted	8	80	0.16	−0.10	12.3	−1.7	15	−10	632	8.6	+0.3	233	+13%
Perm.+themed	8	75	0.15	−0.12	12.4	−1.7	15	−11	589	8.8	+0.4	223	+9%

Table 24: Effect of cross-run memory. Within a run the model writes either keyed xml notes or a single compact summary; between runs they are *consolidated* into the memory bank (xml memories or one compact summary), *kept* verbatim (xml notes only), or discarded (*None*, the baseline for its kind of notes). All with the themed reskin and generic guidance. Pooled over models: each method is compared with its baseline only on matched runs (the same model on the same seed; Pairs, over Agents models), and Acts, Floor and A1 (mean acts cleared, mean final floor and Act 1 win rate) are over those runs. Usage is compared on like-for-like floors only: the same model, seed, act and room type, and the same occurrence of that room type within the act, kept only where every method run on that model and seed completed it (Floors), so a method is neither credited with cheap early floors nor charged for expensive late ones. Calls and k tokens (input, including cached, plus output, in thousands) are means per floor; the token difference is the ratio of totals. Each Δ is the method minus its baseline. All runs are A20 on the held-out seeds, each model at its main-table reasoning setting. Models: Qwen3.8 27B, Qwen3.6 35B-A3B, Qwen3.5 122B-A10B, Gemini 3.8 Flash (low), Nemotron 3 Nano Omni, GPT-5.6 Luna, gpt-oss-120b, GLM-5.3 Flash. Per-model results are in Table 28.

			Performance (matched runs)						Usage per floor (matched floors)				
Method	Agents	Pairs	Acts	Δ	Floor	Δ	A1%	Δ	Floors	Calls	Δ	k tokens	Δ
XML notes													
Consolidate	8	80	0.19	+0.00	13.1	+0.5	18	−1	647	8.9	+0.6	245	+36%
Keep notes	8	80	0.25	+0.06	14.1	+1.6	24	+5	647	8.6	+0.4	297	+65%
None (baseline)	8	80	0.19	–	12.5	–	19	–	647	8.2	–	180	–
Compact notes													
Consolidate	8	80	0.19	+0.01	12.9	+0.6	19	+1	647	7.9	+0.3	175	+7%
None (baseline)	8	80	0.17	–	12.3	–	18	–	647	7.6	–	163	–

Table 25: Which part of the game each model loses on. Each column names the part of the game the agent plays, with Bottled AI (Heart strategy) playing the other three: *Combat* is every decision inside a fight; *Rewards* the card, relic and potion picks after one, chests and boss relics; *Events* events, shops and rest sites; *Path* the route through the map. Under *All* the agent plays the whole game. Each cell is the change in mean final floor against Bottled AI playing the whole game on the same seeds (floor 11.3, 0.20 acts cleared); negative means handing that part over costs floors. Floors rather than acts, because Bottled AI clears so few acts on these seeds that any difference in acts is capped near zero. The LLMs run with the themed reskin, xml notes and no memory bank, at the reasoning setting in grey; the last row pools them. A superscript gives the number of paired seeds where it is fewer than 10. Absolute levels are in Table 30.

		Part of the game the agent plays (Δ floor vs Bottled AI)				
	Agent	All	Combat	Rewards	Events	Path
	Pick random option	−3.8	−5.6	−2.6	−1.8	−2.3
	Pick first option	−4.2	−4.3	+0.0	−1.8	−2.2
	Qwen3.8 27B (xhigh)	+2.3	−1.9	+2.0	−1.7	−0.9
	Qwen3.6 35B-A3B (xhigh)	−2.7	−3.2	−2.0	+0.8	+1.5
	Qwen3.5 122B-A10B (xhigh)	−3.0	−4.1	+1.1	−0.4	−0.7
	Gemini 3.8 Flash (high)	–	+0.4	+0.5	−0.3	−0.7
	Gemini 3.8 Flash (low)	+11.0	−0.3	−0.1	−0.3	+0.3
	Muse Spark 1.3 (xhigh)	–	−0.6	−0.6	−2.5	+2.0
	Nemotron 3 Nano Omni	−3.2	−5.1	−1.3	−1.8	−2.1
	GPT-5.6 Luna (max)	+1.6	−2.0	−1.2	−1.5	−1.2
	gpt-oss-120b (high)	+0.7	-3.4^{9}	+0.1	+0.0	+0.5
	LLMs pooled (9 entries)	+1.2	−2.2	−0.2	−0.9	−0.1

Table 26: Effect of the harness, per model (supplementary to Table 2); Orak cells score each seed as Table 27 counts it, so a run Orak's harness ended is a finished run lost by Orak, scored at the floor it had reached; the harness variants were run without the reskin or a memory bank (xml notes only). All cells are A20 on the held-out seeds, with each model at its main-table reasoning setting (grey). Acts is the mean number of acts cleared per run, Floor the mean final floor, and A1 the Act 1 win rate (%). A superscript gives the seed count where fewer than 10 seeds completed. **Best** and second best are per row, among cells with all 10 seeds.

		Default			Single action			Single action, by index			Continuous			Orak		
	Agent	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%
	Qwen3.8 27B (xhigh)	**0.40**	15.7	**40**	0.30	13.6	30	0.30	14.8	30	0.20	**16.3**	20	0.00	8.3	0
	Qwen3.6 35B-A3B (xhigh)	0.00	**8.6**	0	0.00	8.3	0	0.00	8.5	0	0.00	7.5	0	0.00	7.4	0
	Qwen3.5 122B-A10B (xhigh)	0.00	**10.6**	0	0.00	8.5	0	0.10	8.8	10	0.00	8.9	0	0.00^{6}	5.7^{6}	0^{6}
	Gemini 3.8 Flash (low)	0.30	16.0	30	0.40	17.0	30	0.30	17.9	20	**0.50**	**19.6**	**40**	0.10	10.4	10
	Nemotron 3 Nano Omni	0.00	**9.2**	0	0.00	5.2	0	0.00	6.8	0	0.00	7.2	0	0.00^{5}	6.0^{5}	0^{5}
	GPT-5.6 Luna (max)	0.40	16.2	**40**	0.20	15.3	20	**0.50**	**19.5**	**40**	0.50^{2}	13.5^{2}	50^{2}	–	–	–
	gpt-oss-120b (high)	**0.30**	**15.3**	**30**	0.00	12.9	0	0.20	13.7	20	0.00	9.6	0	0.10	9.4	10
	GLM-5.3 Flash (max)	0.00^{1}	10.0^{1}	0^{1}	–	–	–	0.33^{3}	13.0^{3}	33^{3}	–	–	–	–	–	–
	Mean (4 agents)	**0.25**	**13.9**	**25**	0.17	12.9	15	0.20	13.7	18	0.17	13.3	15	0.05	8.9	5

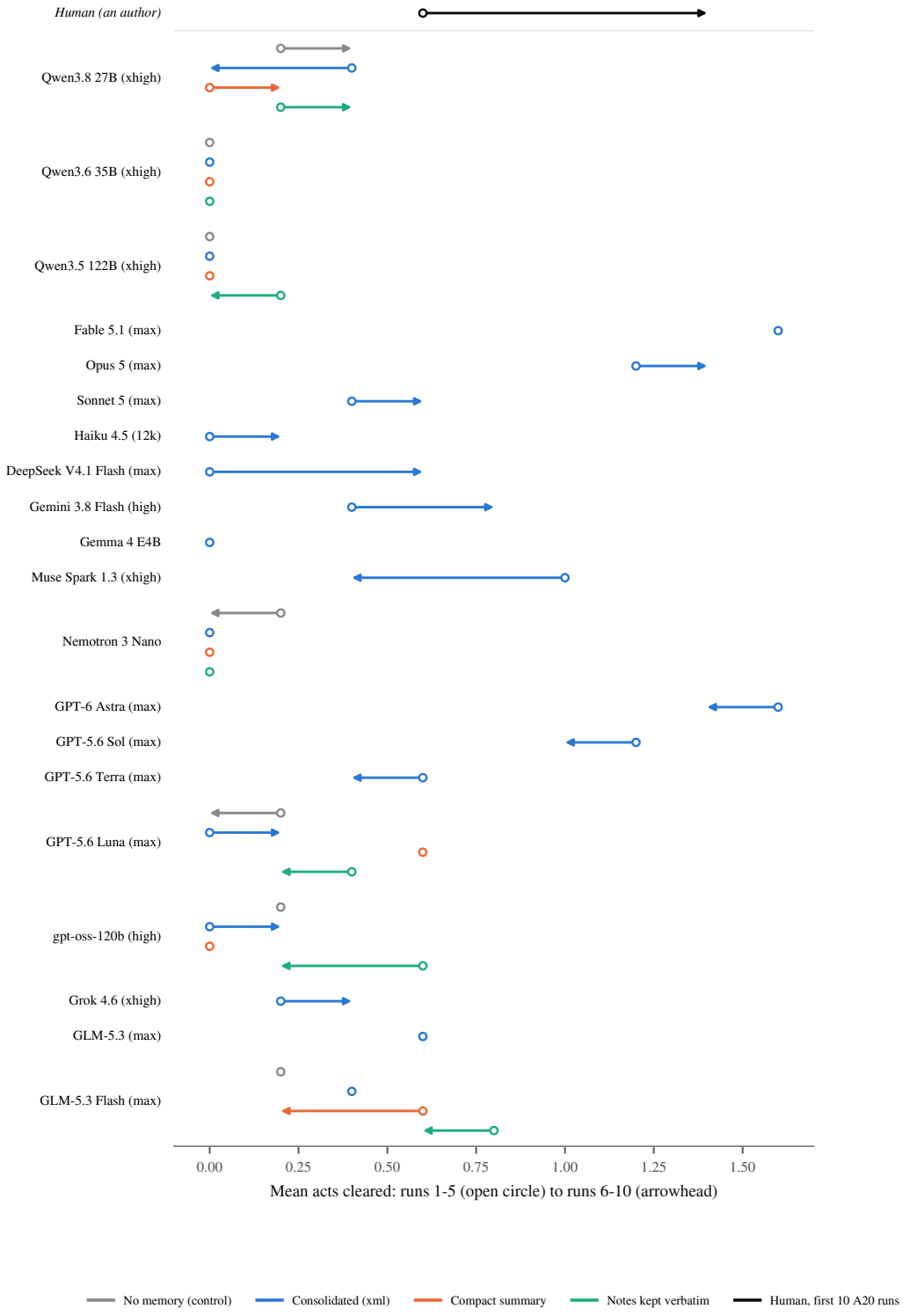


Figure 18: Do models learn from their memories? Mean acts cleared per run over the first five runs of each chain (open circle) against the last five (arrowhead), by memory mechanism; a circle alone means no change. The two halves are different seeds, so the no-memory cohort on the same seeds (grey) is the control for how hard each half is: learning shows as an arrow further right than the grey one. Only cohorts with all 10 seeds are shown. For reference, the top row is one of the authors (a non-professional player) over their first ten A20 Ironclad runs, playing the unmodified game without a harness or memory bank; with five runs per half, a single win moves a half's mean by 0.8 acts.

Table 27: How runs end in each harness, per seed: the two arms of Table 2 on the held-out seeds at A20, each at the reasoning level the model runs at in our harness (– for a model without an effort dial). A seed is *finished* at its first launch that reached the game-over screen (or, for Orak, used up its 200 decisions while still progressing) or that the harness itself ended: a harness failure ends the run and counts against that harness, and a later relaunch does not replace it. *Harn.*: of the finished seeds, those the harness ended, by an exception in its own code, a stall waiting for a game state, or a loop that made no progress. Unfinished seeds are counted by how their last launch ended. *Prov.*: the model's API failed, or our configuration of it was wrong. *Killed*: the process was stopped from outside (pre-emption, the wall clock or cancellation) with no error recorded. *Launches* counts every attempt, since a failed seed is retried; our harness also resumes an interrupted run from a checkpoint, which Orak cannot. Seeds not yet run are left out. Launches ended by a harness failure, Orak: crashed on an empty model reply (9); looped in place until the step budget (2); stalled at an event (7); ours: none. Across every launch of our harness on these seeds, in all arms (2422 launches), those that ended in a harness failure: crashed saving the memory bank (5); gave up after ten replies with no valid action (1).

		Ours					Orak				
Model	Effort	Finished	Harn.	Prov.	Killed	Launches	Finished	Harn.	Prov.	Killed	Launches
Qwen3.8 27B	xhigh	10	0	0	0	10	10	5	0	0	19
Qwen3.6 35B-A3B	xhigh	10	0	0	0	10	10	1	0	0	10
Qwen3.5 122B-A10B	xhigh	10	0	0	0	10	6	1	2	2	23
DeepSeek V4.1 Flash	max	0	0	0	0	0	10	1	0	0	10
Gemini 3.8 Flash	low	10	0	0	0	10	10	4	0	0	10
Nemotron 3 Nano Omni	–	10	0	0	0	10	5	1	0	0	5
GPT-5.6 Luna	max	10	0	0	0	10	0	0	10	0	50
gpt-oss-120b	high	10	0	0	0	10	10	1	0	0	10
GLM-5.3 Flash	max	1	0	5	0	6	0	0	1	0	5
All		71	0	5	0	76	61	14	13	2	142

Table 28: Effect of cross-run memory, per model (supplementary to Table 24). Within a run the model writes either keyed xml notes or a single compact summary; between runs they are *consolidated* into the memory bank (xml memories or one compact summary), *kept* verbatim (xml notes only), or discarded (*None*). All with the themed reskin and generic guidance. All cells are A20 on the held-out seeds, with each model at its main-table reasoning setting (grey). Acts is the mean number of acts cleared per run, Floor the mean final floor, and A1 the Act 1 win rate (%). A superscript gives the seed count where fewer than 10 seeds completed. **Best** and second best are per row, among cells with all 10 seeds.

		XML notes									Compact notes					
		Consolidate			Keep notes			None			Consolidate			None		
	Agent	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%
	Qwen3.8 27B (xhigh)	0.20	14.9	20	**0.30**	14.2	**30**	**0.30**	13.6	**30**	0.10	13.2	10	0.20	**15.0**	20
	Qwen3.6 35B-A3B (xhigh)	0.00	**10.2**	0	0.00	10.1	0	0.00	8.6	0	0.00	8.4	0	0.00	8.0	0
	Qwen3.5 122B-A10B (xhigh)	0.00	**9.8**	0	0.10	9.6	10	0.00	8.3	0	0.00	7.5	0	0.00	6.8	0
	Gemini 3.8 Flash (low)	**0.70**	**22.6**	**60**	0.20	15.3	20	0.60	22.3	**60**	0.40	17.6	40	0.40	17.1	40
	Nemotron 3 Nano Omni	0.00	6.8	0	0.00	7.7	0	0.10	**8.1**	10	0.00	**8.1**	0	0.00	**8.1**	0
	GPT-5.6 Luna (max)	0.10	14.2	10	0.30	17.4	20	0.10	12.9	10	**0.60**	**21.1**	**60**	0.20	14.5	20
	gpt-oss-120b (high)	0.10	10.4	10	**0.40**	**16.3**	**40**	0.20	12.0	20	0.00	11.8	0	0.10	11.4	10
	GLM-5.3 Flash (max)	0.40	15.6	40	**0.70**	**22.2**	**70**	0.20	14.4	20	0.40	15.9	40	0.50	17.6	50
	Mean (8 agents)	0.19	13.1	18	**0.25**	**14.1**	**24**	0.19	12.5	19	0.19	13.0	19	0.18	12.3	18

Table 29: Effect of the knowledge reskin, per model (supplementary to Table 23). *None* shows the real game names; *Themed* is our default Lich's Lair reskin; *Random* replaces every name with pronounceable nonsense; *Permuted* swaps each name for another real game name of the same kind; *Perm.+themed* applies the same permutation in the themed vocabulary. All with xml notes, a read-write memory bank and generic guidance. All cells are A20 on the held-out seeds, with each model at its main-table reasoning setting (grey). Acts is the mean number of acts cleared per run, Floor the mean final floor, and A1 the Act 1 win rate (%). A superscript gives the seed count where fewer than 10 seeds completed. **Best** and second best are per row, among cells with all 10 seeds.

	None			Themed			Random			Permuted			Perm.+themed		
Agent	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%	Acts	Floor	A1%
Qwen3.8 27B (xhigh)	0.20	14.0	20	0.20	14.9	20	**0.30**	**15.9**	**30**	0.20	14.7	20	0.40[5]	19.8[5]	40[5]
Qwen3.6 35B-A3B (xhigh)	0.00	9.7	0	0.00	10.2	0	0.00	9.5	0	0.00	9.5	0	0.00	**11.0**	0
Qwen3.5 122B-A10B (xhigh)	0.10	**10.9**	10	0.00	9.8	0	0.00	7.6	0	0.00	9.2	0	0.00	8.0	0
Claude Haiku 4.5 (12k)	**0.10**	11.4	**10**	**0.10**	**11.9**	**10**	–	–	–	–	–	–	–	–	–
DeepSeek V4.1 Flash (max)	**0.80**	**22.0**	**70**	0.30	16.9	30	–	–	–	–	–	–	–	–	–
Gemini 3.8 Flash (high)	1.20[5]	31.0[5]	80[5]	0.60	22.9	50	–	–	–	–	–	–	–	–	–
Gemini 3.8 Flash (low)	0.50	20.2	40	0.70	22.6	**60**	**0.80**	**24.1**	**60**	0.40	17.6	30	0.30	16.4	30
Nemotron 3 Nano Omni	0.00	5.3	0	0.00	6.8	0	0.00	**8.3**	0	0.00	7.4	0	0.00	7.0	0
GPT-5.6 Luna (max)	**0.60**	18.5	**60**	0.10	14.2	10	**0.60**	**19.6**	50	0.40	15.0	40	0.10	12.9	10
gpt-oss-120b (high)	0.00	10.4	0	0.10	10.4	10	0.00	**10.7**	0	0.00	9.2	0	0.00	9.8	0
GLM-5.3 (max)	**0.70**	**24.4**	**60**	0.60	22.1	**60**	–	–	–	–	–	–	–	–	–
GLM-5.3 Flash (max)	**0.70**	**23.1**	**70**	0.40	15.6	40	0.30	15.3	30	0.30	16.1	30	0.50	18.0	50
Mean (7 agents)	**0.27**	**14.0**	**26**	0.19	12.8	17	0.24	13.6	20	0.16	12.0	14	0.13	11.9	13

Table 30: Which part of the game the models lose on, per agent (supplementary to Table 25). Each column names the part of the game the agent plays, with Bottled AI (Heart strategy) playing the rest; under *All* the agent plays the whole game. Acts is the mean number of acts cleared and Floor the mean final floor. The LLMs run with the themed reskin, xml notes and no memory bank, at the reasoning setting in grey. A superscript gives the seed count where fewer than 10 seeds completed.

	All		Combat		Rewards		Events		Path	
Agent	Acts	Floor	Acts	Floor	Acts	Floor	Acts	Floor	Acts	Floor
Pick random option	0.00	7.5	0.00	5.7	0.00	8.7	0.10	9.5	0.10	9.0
Pick first option	0.00	7.1	0.00	7.0	0.00	11.3	0.00	9.5	0.00	9.1
Qwen3.8 27B (xhigh)	0.30	13.6	0.00	9.4	0.10	13.3	0.10	9.6	0.00	10.4
Qwen3.6 35B-A3B (xhigh)	0.00	8.6	0.00	8.1	0.00	9.3	0.10	12.1	0.40	12.8
Qwen3.5 122B-A10B (xhigh)	0.00	8.3	0.00	7.2	0.10	12.4	0.10	10.9	0.10	10.6
Gemini 3.8 Flash (high)	–	–	0.00	11.7	0.10	11.8	0.00	11.0	0.00	10.6
Gemini 3.8 Flash (low)	0.60	22.3	0.10	11.0	0.00	11.2	0.00	11.0	0.20	11.6
Muse Spark 1.3 (xhigh)	–	–	0.00	10.7	0.00	10.7	0.00	8.8	0.20	13.3
Nemotron 3 Nano Omni	0.10	8.1	0.00	6.2	0.00	10.0	0.00	9.5	0.00	9.2
GPT-5.6 Luna (max)	0.10	12.9	0.00	9.3	0.00	10.1	0.00	9.8	0.00	10.1
gpt-oss-120b (high)	0.20	12.0	0.00[9]	7.6[9]	0.00	11.4	0.10	11.3	0.20	11.8
Bottled AI (plays all)	0.20	11.3								

K COST AND REASONING FRONTIERS IN FULL

Figure 2a (in the main text) names only the points on its front. Figure 19 repeats it with every point named, and Figs. 20 and 21 give the fronts against cost per floor and thinking tokens per floor, which rank models on efficiency rather than on what a whole run costs.

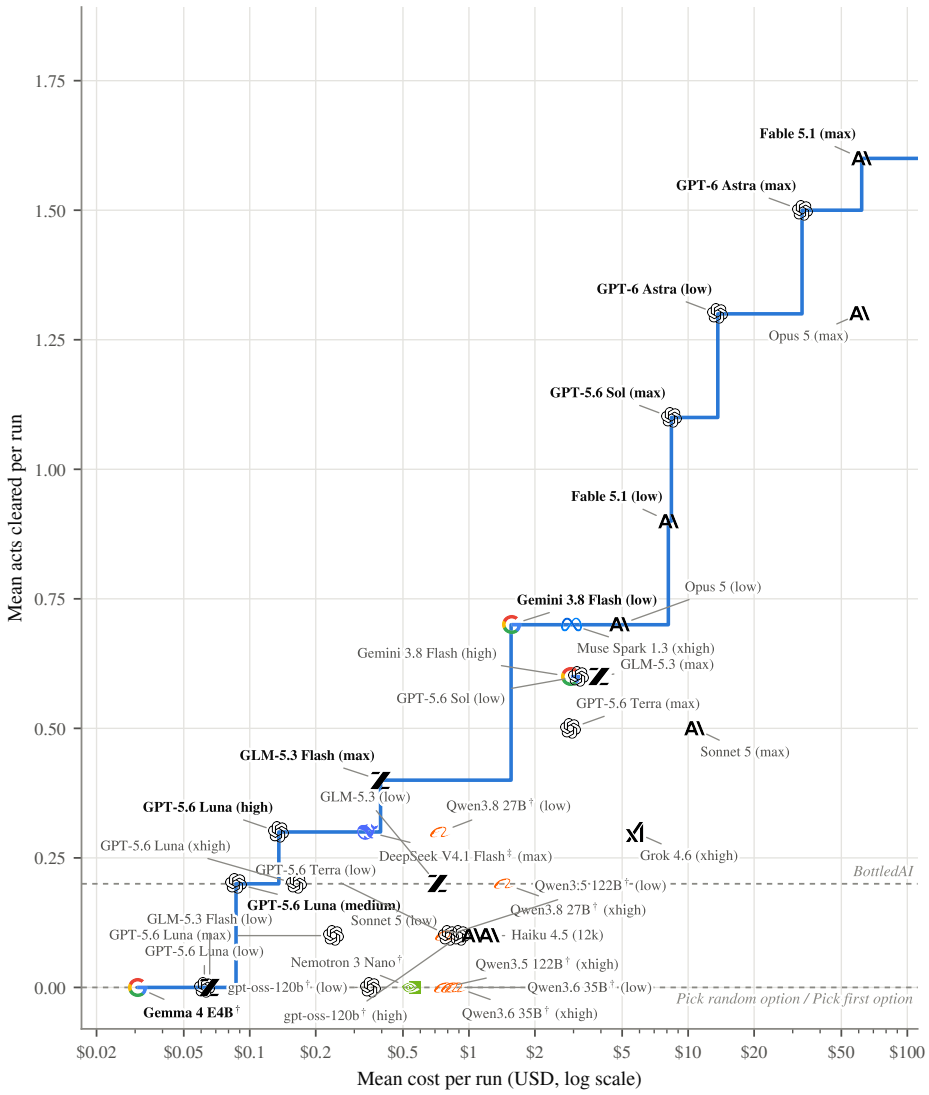


Figure 19: Figure 2a with every point named, with its reasoning-effort level and, where it has fewer than 10 seeds, its seed count.

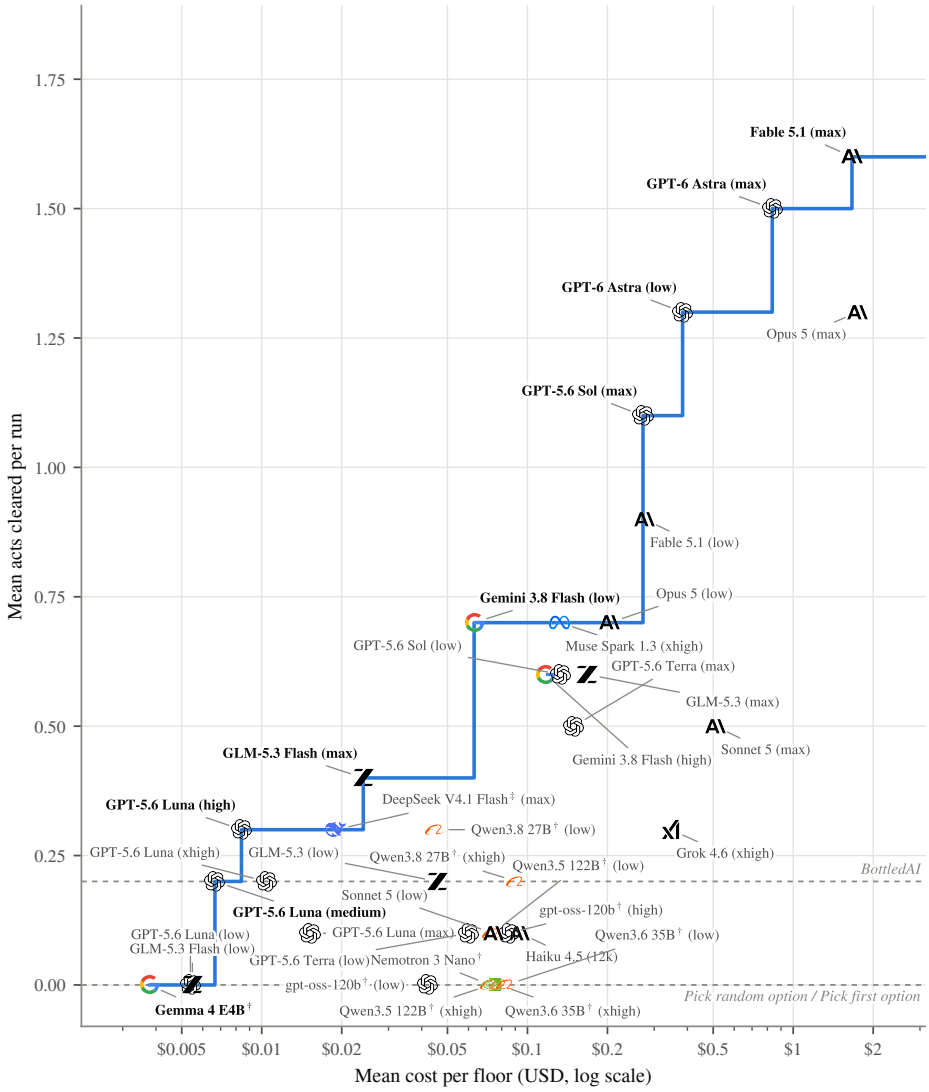


Figure 20: Performance against cost per floor: mean acts cleared per run against mean cost per floor (log scale), for the same points as Fig. 2a. †Self-hosted; priced at Artificial Analysis headline rates. ‡Peak-rate list price, an upper bound.

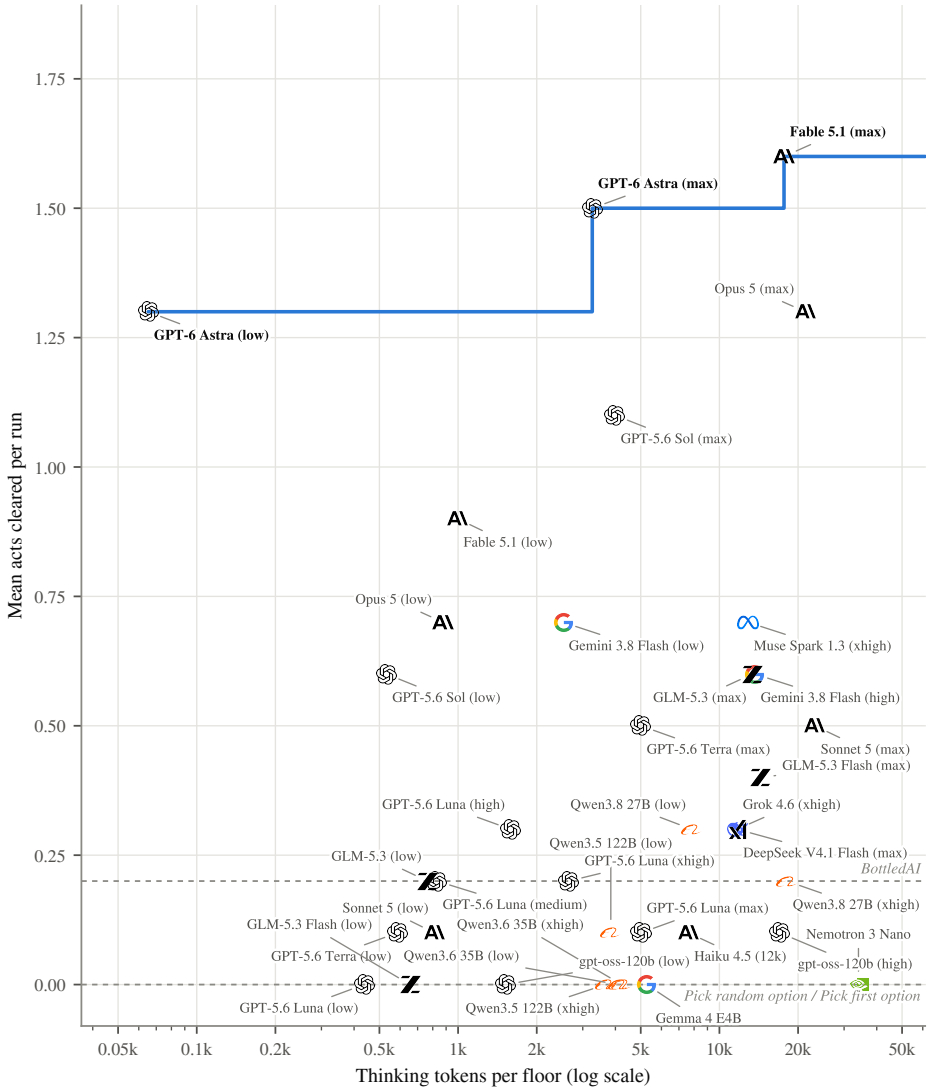


Figure 21: Performance against thinking: mean acts cleared per run against thinking tokens per floor (log scale), for the same points as Fig. 2a.

L TIME AND COST OF A RUN

Table 31 gives what a run costs in wall-clock time, money, tokens and mechanical errors, as a guide to budgeting a new cohort. A cohort's seeds are played in series against one game instance, so its wall-clock time is the sum of its runs'.

Table 31: What a run costs, for the agents of Table 1: wall-clock minutes per seed (one game instance, seeds in series), dollars per run and per 10-seed cohort (priced as in Table 1), model calls and tokens per run (input including cached, plus output, in thousands); tool-call errors are in Table 15. Time depends on the provider's throughput on the day as well as on the model.

			Minutes/seed		Cost ($)		Per run	
	Agent	Runs	median	mean	/run	/10 seeds	Calls	k tokens
←	Pick first option	10	2	2	–	–	–	–
	Pick random option	10	2	2	–	–	–	–
	BottledAI w/o Heart	10	2	2	–	–	–	–
	BottledAI w/ Heart	10	2	2	–	–	–	–
	Qwen3.8 27B (xhigh)	10	99	107	1.45	15	124	4,641
	Qwen3.6 35B-A3B (xhigh)	10	30	31	0.87	9	104	2,036
	Qwen3.5 122B-A10B (xhigh)	10	13	14	0.77	8	77	1,601
	Claude Fable 5.1 (max)	10	231	229	61.91	619	268	24,604
	Claude Opus 5 (max)	10	231	214	60.62	606	301	44,517
	Claude Sonnet 5 (max)	10	107	157	10.70	107	159	14,122
	Claude Haiku 4.5 (12k)	10	21	31	1.23	12	101	3,395
	DeepSeek V4.1 Flash (max)	10	23	24	0.35	3	119	4,256
	Gemini 3.8 Flash (high)	10	44	62	2.89	29	208	8,382
	Gemma 4 E4B	10	17	61	0.03	0	82	1,349
	Muse Spark 1.3 (xhigh)	10	52	50	2.93	29	228	6,972
	Nemotron 3 Nano Omni	10	130	135	0.54	5	71	1,595
	GPT-6 Astra (max)	10	108	165	33.09	331	546	20,032
	GPT-5.6 Sol (max)	10	55	58	8.39	84	331	10,435
	GPT-5.6 Terra (max)	10	35	34	2.91	29	230	5,667
	GPT-5.6 Luna (max)	10	21	49	0.24	2	168	4,384
	gpt-oss-120b (high)	10	84	87	0.91	9	175	6,498
	Grok 4.6 (xhigh)	10	58	205	5.68	57	171	4,649
	GLM-5.3 (max)	10	119	121	3.92	39	160	7,194
	GLM-5.3 Flash (max)	10	103	118	0.40	4	107	4,641