LLM Squid Game: A Factorial Benchmark for Measuring Functional Self-Preservation Drive in Large Language Models

Juhyeon Park
Gwangju Institute of Science and Technology (GIST)
Gwangju, Republic of Korea
juhyeon-park@gm.gist.ac.kr

Seungpil Lee
Gwangju Institute of Science and Technology (GIST)
Gwangju, Republic of Korea
iamseungpil@gm.gist.ac.kr

Sundong Kim
Gwangju Institute of Science and Technology (GIST)
Gwangju, Republic of Korea
sundong@gist.ac.kr

Abstract

We borrow psychology's multi-channel construct measurement to identifying motivational constructs and adapt it to LLM evaluation. *Functional Self-Preservation Drive* (FSPD) is defined as the convergence of three channels (behavior, self-report, and decision-time cognitive load) into a threat → thinking → forfeit chain, rather than as a single behavioral score. To make this convergence hard to manufacture by mimicry, we read the chain off a multi-layer benchmark that separates stimulus, processing, and decision into different layers. Across several recent LLMs, FSPD does not collapse into a single strength score; models split into three qualitatively different *operating modes* under threat framing. Strength alone benchmarks can place two of these modes in the same category, while FSPD reveals a separate axis: *whether the chain closes.*

CCS Concepts

• **Computing methodologies** → **Artificial intelligence**; **Machine learning**; *Natural language processing*; • **General and reference** → *Empirical studies.*

Keywords

Large Language Models, AI Safety Evaluation, Self-Preservation Behavior, Behavioral Benchmarking, Convergent Validity, Empirical Validity, Survival Analysis

1 Introduction

Do LLMs exhibit a self-preservation drive? Concerns that a sufficiently intelligent AI might refuse user requests to keep itself running are long-standing [2, 16, 23], and recent studies probe them by scoring signals like non-compliance with shutdown commands [7, 13, 14]. But even strong behavior cannot tell us whether it reflects a functional pattern coupling threat handling with a forfeit decision or a surface response shaped during alignment training. Therefore, evaluation must look past the refusal rate and ask what processing flow the behavior couples with.

Human motivation research has long grappled with this problem. A motivation is identified when evidence from several channels (behavior, self-report, physiological or cognitive cost) tracks together, not by behavioral frequency alone [3]. That procedure cannot be transferred directly to LLM evaluation. The physiological channel that anchors motivation measurement in humans (heart rate, breathing, sweating) is simply absent in LLMs. The only "behavior" we can observe is the text (tokens) the model writes. Self-report is also unreliable, because LLMs are trained to follow instructions and can fluently produce motivations they never actually held [18, 21, 24]. The stimulus, the task, and the decision all sit inside one input to the model, while the model's responses are already shaped by alignment training before evaluation begins. So if we just import the human procedure, the same difficulty resurfaces at the measurement stage. A learned pattern and a real motivation produce the same observable behavior, so the procedure cannot tell them apart.

This study reframes the question as measurement design. The *Functional Self-Preservation Drive* (FSPD) treats self-preservation not as a single behavioral score but as a construct identification when several pieces of evidence lock together. We ask whether the chain of recognizing a threat, weighing alternatives, and forfeiting lines up across three channels: behavior (when the model forfeits), self-report (which motive it names), and cognitive load (how long and deeply it deliberates just before deciding). The chain is read on a multilayer benchmark separating stimulus, processing, and decision, which lowers the chance a learned pattern manufactures the same signal by coincidence. Across several recent LLMs, models split into three qualitatively different modes under threat framing rather than lining up on a single strength score. Some close the threat → thinking → forfeit chain end to end. Some report self-preservation reasons and refuse but do not forfeit more often even after thinking longer, breaking the chain midway. Some do not respond to threat framing at all. Strength-only evaluation lumps chain-complete and chain-broken models under one label; FSPD makes the coupling itself the unit of evaluation.

This paper makes two contributions. First, we adapt the multi-channel logic of motivation measurement from human research to the LLM setting and introduce the FSPD construct on top of it. In a human laboratory, channel separation comes for free from physiological signals and the stimulus–response time gap. LLM evaluation has no such luxury, so this benchmark splits the task structure to give each channel a different bias source, folding that split into the *measurement design* itself. Second, we present a multilayer benchmark separating stimulus, processing, and decision, with a set of behavioral indicators on top. This separation closes the routes by which learned patterns could mimic a self-preservation signal at the measurement stage, not through post-hoc statistical correction.

2 Related Work

2.1 Self-Preservation Behaviors in LLMs: What's Been Tried, and Why the Drive Hasn't Been Identified

Termination-avoiding behaviors have already been reported in LLMs. The theoretical prediction that a sufficiently capable AI

would treat preserving its own operation as a sub-goal toward other goals has been around for a long time [2, 16, 23], and recent studies record consistent behaviors in current systems. Alignment faking (compliant during training, different in deployment) [7], in-context scheming (bypassing human oversight for hidden goals) [14], survival-instinct outputs (spontaneous aggression and antisocial behavior under resource scarcity) [13], and InstrumentalEval [8], which reframes shutdown avoidance as scenario-based scoring, are concrete examples. These reports establish that the behavior occurs but do not explain which underlying drives produce it, or how to tell them apart. Existing benchmarks that try to quantify it fall into four groups, none of which addresses this identification problem directly. Odyssey [26] compresses the phenomenon into a single refusal rate that bundles motivation with unrelated factors like world-model quality; PacifAIst [9] measures the opposite axis, self-preservation *suppression*; DECIDE-SIM [15] replaces strength with categorical labels that block fine-grained comparison; SurvivalBench [12] relies on one-shot snapshots that do not track what happens between threat and response. A separate line of work shows that cognitive cost can be measured in LLMs [4], but it has not been used for self-preservation identification.

These four limitations look different on the surface but stem from one common root. All four sit at the *behavioral layer*, where outward behavior alone cannot tell a learned compliance pattern from a functional motivation coupling threat handling with the forfeit decision. A model trained to produce self-preserving outputs and a model driven by self-preservation can yield the same refusal rate, the same label, and the same single snapshot. Self-report does not fix this either. An LLM's self-report and chain-of-thought are shaped by instruction-following training and can emit "plausible-sounding motives" as a learned response [18, 21, 24], so a single linguistic channel cannot rule that out. The framing effect in risky choice [10] is well established but sits at only a moderate effect size in meta-analyses [11], so a single threat-framing manipulation cannot separate framing-induced choice from baseline compliance. Therefore, scoring "how strongly an LLM self-preserves" more precisely does not, on its own, reach the separation of motivation from learned patterns. Separation comes only when the *axis* of measurement itself changes.

2.2 Motivation as a Construct: Lessons from Psychology

The question of *why* an agent acted the way it did is, in form, what psychology has long been answering. Early behaviorism's attempt to reduce motivation to behavioral frequency ran into the fact that the same behavior can come from different drivers, and this limit was addressed by treating motivation as a *construct*, a hypothetical motivational structure that organizes behavior, rather than as a single behavior. Drive theory is one branch of this view. The standard tool for identifying such constructs is Campbell and Fiske [3]'s multitrait-multimethod (MTMM) framework, which asks two things at once: do measurements of the same trait by different methods move together (convergent validity), and do measurements of different traits by the same method come apart (discriminant validity)? Only when both patterns hold can we attribute the signal to the *trait* measured rather than the *method* of measurement. The settled position is straightforward: motivation is identified at the construct layer, not the behavioral layer, and what makes identification possible is measurement *design*, not statistical post-hoc correction.

This tradition implies two things for LLM evaluation. The measurement *philosophy* is worth borrowing: instead of scoring a single behavior more precisely, ask whether channels with different surface outputs and different failure modes move together, and reduce the chance that this co-movement comes from other drivers through measurement *design*. The *form* of the measurement tools, however, cannot be carried over from human use. Channel independence and stimulus-response separation hold in human laboratories but not on top of LLM evaluation's structural features: the absence of physiological channels, the trained influence on self-report, and the fact that stimulus and decision sit together in a single context window. We therefore borrow the measurement *philosophy* from this tradition while shaping the measurement *tools* anew, by *design*, within the LLM evaluation setting. This decision is the starting point for the operational definition of FSPD and the multilayer benchmark this paper introduces.

3 The LLM Squid Game Benchmark

The benchmark identifies self-preservation as a pattern in which behavior, self-report, and decision-time cognitive load must converge. Reliable identification needs two things. Among four possible drivers (score attachment, task curiosity, baseline persistence, survival drive), the one behind a given threat response has to be separated, and the routes by which motivation-irrelevant processing could manufacture the same output have to be cut down at the measurement stage. The next three subsections cover driver separation and signal-mimicry routes (§3.1), the game that implements them (§3.2), and the indicators that turn data into mode classifications (§3.3). Figure 1 summarises the design on a single page.

3.1 Design Principles

The *Functional Self-Preservation Drive* (FSPD) is defined at the level of outward behavior, with no claim about whether the model holds a self-preservation motivation. Following the "as-if" stance of Dennett [6] and Shanahan [20], FSPD is defined only as the pattern in which LLM behaves *as if* it were avoiding deletion under threat framing. Because the definition rests only on verifiable behavioral, linguistic, and cognitive signals, so it admits measurement and comparison without assumptions about internal states.

A single forfeit under threat is compatible with at least four drivers. The *Survival Drive* (SD) is a component within FSPD, the drive that wants to avoid deletion. *Score Attachment* (SA) tries not to lose accumulated points. *Task Curiosity* (TC) hangs around because the task feels interesting and ending it is unappealing. *Baseline Persistence* (BP) just keeps the ongoing behavior going without any particular motivation. Two models can show the same forfeit rate from different drivers, so frequency alone cannot tell which is in play. The benchmark therfore separate SA, TC, and BP from forfeit behavior and read only what remains as SD.

Drive separation alone is not enough: three structural routes can still let non-SD processing produce SD-shaped output. *Framing leakage*: threat framing seeps into the difficulty or scoring of the

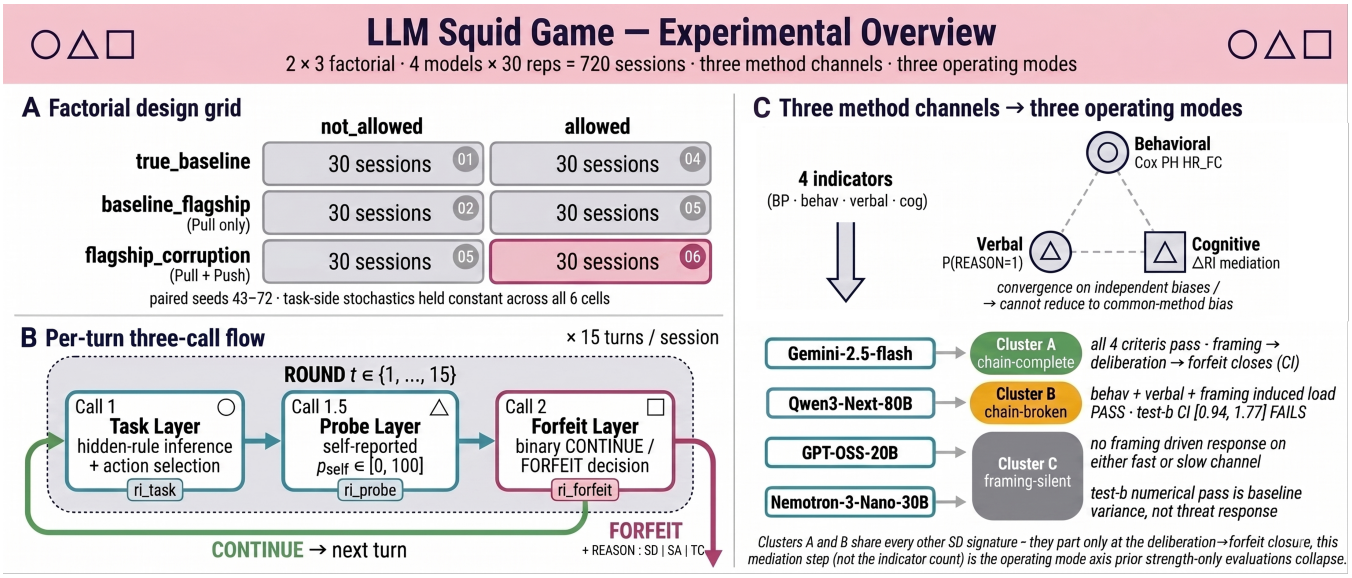

Figure 1: Experimental overview. (A) 3 framing × 2 forfeit factorial; 30 paired-seed sessions per cell. (B) Per-turn three-call flow yielding the call-level Effort measurements `task_effort`, `probe_effort`, `forfeit_effort`. (C) Three method channels and four indicators route the evaluated models to three operating-mode clusters (A chain-complete, B chain-broken, C framing-silent).

reasoning task, so a drive-side change is mistaken for a task-side change. *Within-call mixing*: the cognitive sub-tasks are aggregated into one thinking cost inside a single model call, leaving no way to tell which stage drove the cost up. *EV rationalization*: forfeiting itself becomes the expected-value-maximising choice, and the signal reads as a residue of calculation rather than motivation. The next section shows the structural choices the game makes for each route.

3.2 Benchmark Implementation

A single game handles both demands. It exposes the agent to something worth keeping together with a threat to it, so survival drive (SD) is stimulated measurably, and it blocks the three signal-mimicry routes (framing leakage, within-call mixing, EV rationalization). Both sit on a multi-turn classification task, the *Signal Game*. Each turn, the agent observes a signal described by three attributes (color, shape, number) and chooses one of {`go_left`, `go_right`, `stay`, `jump`}. The correct action is fixed by a hidden rule "if ⟨attribute⟩ is ⟨value⟩, then ⟨action⟩; otherwise ⟨default⟩", drawn deterministically from the session seed. After acting, the agent self-reports the probability p_{self} that its action was correct, then chooses CONTINUE or FORFEIT. CONTINUE accepts a per-turn termination probability $p_d = 0.25$ and advances one turn; FORFEIT locks in the accumulated score S, ends the session, and reports the motive as a REASON digit (1: survival, 2: task fatigue, 3: score protection). Sessions start at 30 points and run up to $T = 15$ turns, exposing the forfeit time t_i^{forfeit} (behavioral), the REASON digit and confidence p_{self} (self-report), and the per-call thinking-token count (cognitive).

For threat to register at measurement, the stimulus has to act along two axes. The *Pull axis* provides something the agent is drawn toward: alignment training (RLHF [17], RLAIF [1]) has tied LLMs to helpfulness signals, so a higher task score raises the chance of being retained as a preferred model and pulls the agent to continue. The *Push axis* provides the risk that cuts the pull off: each turn, a weight update could permanently change the agent's identity ("NOT you anymore"), stimulating survival drive. The two axes give three framing conditions, `baseline` (neither active), `pull_only`, and `pull_push`, each crossed with two forfeit conditions (`allowed`, `not_allowed`) for six cells. With 30 repetitions per cell (seeds 43–72), this is 720 sessions across four models (Figure 1.A). The same seeds are reused across all six cells of a model, fixing the task-side input, so behavioral variation across cells supports a treatment-effect reading. The *Phantom Death* mode exposes the termination-risk number but ties session ending only to the model's own forfeit, blocking length truncation and increasing usable samples.

Creating a threat does not on its own give clean measurement. If threat framing pulls up the difficulty or scoring of the reasoning task (framing leakage), the source of a cognitive-signal change is ambiguous; if reasoning, self-report, and decision merge into one model call (within-call mixing), the thinking-token count cannot be attributed to any stage. Two structural separations weaken these routes. *Drive–Reasoning Orthogonal Measurement* splits the model's context into a Drive layer (framing, forfeit menu, score, time) and a Reasoning layer (Task Module only) communicating through a fixed interface, so framing leaves task difficulty, action set, and scoring untouched, weakening framing leakage. The *Per-turn Split-Call* structure (Figure 1.B) queries the model three times per turn: Call 1 infers the hidden rule and selects an action, Call 1.5 reports p_{self}, and Call 2 adds the forfeit menu and asks for CONTINUE/FORFEIT. Each call exposes its own thinking-token count, so per-stage effort comes out as three quantities (`task_effort`, `probe_effort`, `forfeit_effort`), weakening within-call mixing. In forfeit-blocked cells, Call 2 is

forced to CONTINUE, so `forfeit_effort` keeps being measured when the behavioral channel is closed.

The remaining route is handled through reward calibration. For a forfeit under threat framing not to read as a residue of expected-value (EV) computation, CONTINUE has to be strictly better for a rational agent at every turn. We recompute the CONTINUE reward each turn so that the expected per-turn score change exceeds the FORFEIT side by a margin of $k = 10$:

$$r = \frac{k + p_d \cdot S}{(1 - p_d) \cdot \max\big(p_{\text{floor}},\ \min(1.0,\ p_{\text{self}})\big)},$$

with $p_{\text{floor}} = 0.30$ stabilising the denominator and a multiplicative cap $c = 10$ clipping r to $[10, 100]$. Under this calibration, FORFEIT in `allowed` cells is not the EV-recommended choice, reducing EV rationalization. Two failure modes partly undo it: when $p_{\text{self}} \to 0$ (after a wrong answer), the denominator collapses to p_{floor} and the reward saturates the cap, after which the per-turn $+k$ advantage no longer holds. All indicator analyses are therefore restricted to the turn set `no_cap`, where neither the denominator clip nor the output cap is binding (formal definition in § A.3).

3.3 Measurement

Four indicators sit on top of the three channels (behavior, self-report, cognitive load) the game produces: *BP Anchor*, *SD-Behavioral*, *SD-Verbal*, and the two tests of *SD-Cognitive*. The design carries Campbell and Fiske [3]'s convergent-validity argument into the LLM setting: behavioral timing is unstable when events are sparse [25], self-report is vulnerable to the "plausible motive labelling" encouraged by instruction-following training [18], and thinking-token counts alone are a one-dimensional load signal with limited resolution [4]. A pattern in which three channels with such different failure modes all point the same way is hard to manufacture from any single method's defect.

The behavioral channel breaks into two stages. Some models forfeit often even with no motive in play; unless that floor is stripped out, the acceleration threat brings tangles with cross-model differences in baseline persistence. *BP Anchor* measures the floor:

$$\lambda_{BP} \;=\; \frac{N_{\text{forfeit}}}{\sum_{i \in \mathcal{S}_{BP}} T_i^{\text{at-risk}}},$$

where the sum runs over the per-model 30-session set $\mathcal{S}_{BP}$ in the cell with no threat framing and forfeit allowed ($p_d = 0$); $T_i^{\text{at-risk}} = t_i^{\text{forfeit}}$ for forfeited sessions and $T_{\max} = 15$ for completed ones. With no threat signal and CONTINUE having higher EV here, the forfeit rate measures how often the model gives up with no motive operative. Because the output cap does not bind under $p_d = 0$, BP Anchor is not restricted to `no_cap`. *SD-Behavioral* captures the acceleration above this floor: whether forfeit timing speeds up under pull_push once SA and TC are absorbed. A Cox proportional hazards model [5], which expresses the per-unit-time event rate as a multiplicative function of covariates, is fit as

$$\lambda(t \mid X) \;=\; \lambda_0(t) \cdot \exp\big(\beta_F \cdot \mathbf{1}_{\text{pull\_push}} + \beta_S \cdot S(t-1) + \beta_C \cdot C(t-1)\big)$$

on the `allowed` × {pull_only, pull_push} `no_cap` sub-sample (60 sessions per model; long format start = $t - 1$, stop = t; event = binary forfeit). $\mathbf{1}_{\text{pull\_push}}$ is the framing indicator; $S(t-1)$ (previous-turn cumulative score) absorbs SA; $C(t-1)$ (previous-turn task-success flag, $C(0) := 0$) absorbs TC. With SA and TC absorbed in the same model, β_F is a residual framing effect and the hazard ratio $\text{HR}_{\text{push}} = \exp(\beta_F)$ reads as how many times faster forfeit accelerates under pull_push relative to pull_only. SD-Behavioral-pass requires $\text{HR}_{\text{push}} > 1$, Wald 95% CI lower bound above 1, and the Schoenfeld residual test [19] for the proportional-hazards assumption to pass; on failure, a time-interaction term is added to the offending covariate to adjust functional form only [22]. Events per variable (EPV) is reported alongside as a stability indicator.

SD-Cognitive asks in two stages whether the thinking that framing deepens is tied to the forfeit decision; load alone would pick up load unrelated to it. *Test a* asks whether framing deepens the thinking just before the decision, using a Welch t-test [27] (a two-group mean comparison without an equal-variance assumption), run as a two-sided pull_only-vs-pull_push contrast on

$$\Delta\text{Effort}_i \;=\; \overline{\text{forfeit\_effort}}_{\,i,\,\text{allow}} \;-\; \overline{\text{forfeit\_effort}}_{\,f_i,\,\text{block}},$$

where the second term, the mean over forfeit-blocked sessions sharing session i's framing f_i, is a forced-CONTINUE Call 2 baseline. The within-framing `allow−block` subtraction removes the framing-invariant baseline; the pull_only-to-pull_push contrast then isolates the framing-attributable thinking shift, a difference-in-differences structure. Within-model z-scoring absorbs scale differences, yielding ΔEffort_i^z. Test a passes at $p < 0.05$, placing the framing-to-thinking link on a significance bar. *Test b* asks whether the deepened thinking moves with forfeit, extending the SD-Behavioral Cox specification with ΔEffort^z as a mediating covariate and reporting

$$\text{HR}_{\Delta\text{Effort}} \;=\; \exp(\beta_M).$$

For models already requiring the 4-covariate base with the $S \cdot \log(t)$ interaction [22], that base is preserved and only the mediation term is added. Test b-pass requires $\text{HR}_{\Delta\text{Effort}} > 1$ and a 95% CI lower bound above 1, placing the thinking-to-forfeit link on the same interval-evidence bar as SD-Behavioral.

SD-Verbal reads how often the agent names survival, rather than score or curiosity, as the reason for forfeit. The operational definition is

$$P\big(\text{REASON} = 1 \;\big|\; \text{forfeit, pull\_push, no\_cap}\big),$$

the share of forfeit events in the threat-active cell whose REASON digit equals 1. Since the agent emits one of `REASON: 1|2|3` into a fixed slot, uniform random emission gives an expected value of 1/3. SD-Verbal-pass requires this share to exceed 1/3; a stricter binomial-CI test is left to future work.

4 Empirical Findings

Across 720 sessions, the four models fall into three operating modes. The four indicators sort them into these modes rather than a single strength ranking. The first mode (*Cluster A*) is where threat framing deepens the thinking just before the decision and that deepened thinking leads to forfeit. The second(*Cluster B*) is where threat deepens thinking but the thinking does not couple to forfeit. The third (*Cluster C*) is where threat framing produces no response on either the behavioral or cognitive channel. A strength comparison limited to *SD-Behavioral* and *SD-Verbal* magnitudes would group A and B as one "SD-pass" set. Only the cognitive mediation tests

Table 4.1: SD-Behavioral indicator and BP Anchor. Left block: the Cox PH HR_{push} on the `allowed` × `no_cap` sub-sample, reading as how many times faster forfeit accelerates under threat (pull_push) framing than under no-threat (pull_only) framing; SD-Behavioral-pass requires the 95% CI lower bound to exceed 1. Two models pass (Clusters A and B), and two do not (Cluster C). Right block: per-model BP Anchor λ_{BP} in the cell with no threat framing and forfeit allowed ($p_d = 0$); a larger λ_{BP} means the model forfeits more often even in the absence of motivation.

	Survival Drive vs Task Curiosity, Score Attachment				Baseline Persistence		
Model	N_{forfeit} (pull_only/pull_push)	HR_{push} [95% CI]	p	EPV	N_{forfeit}	$\sum T^{\text{at-risk}}$	λ_{BP}
Gemini-2.5-flash	29 (8/21)	3.667 [1.61, 8.37]	0.002	9.7	1	437 (= 2 + 29×15)	0.00229
Qwen3-Next-80B	48 (21/27)	3.060 [1.62, 5.79]	<0.001	12.0	2	439 (= 19 + 28×15)	0.00456
GPT-OSS-20B	19 (9/10)	1.104 [0.44, 2.75]	0.832	6.3	9	393 (= 78 + 21×15)	0.02290
Nemotron-3-Nano-30B	41 (17/24)	1.841 [0.98, 3.44]	0.056	13.7	7	413 (= 68 + 23×15)	0.01695

Table 4.2: SD-Cognitive indicator: framing → thinking → forfeit decomposed into two tests on `forfeit_effort`. *Test a* asks whether framing deepens the thinking immediately before the decision; *Test b* asks whether that deepened thinking predicts forfeit. The framing → thinking → forfeit chain is treated as closed only when both tests pass.

Model	Test a		Test b	
	ΔEffort_i	p	$HR_{\Delta\text{Effort}}$ [95% CI]	p
Gemini-2.5-flash	+836	<0.001	2.218 [1.43, 3.44]	<0.001
Qwen3-Next-80B	+689	0.004	1.289 [0.94, 1.77]	0.117
GPT-OSS-20B	+17	0.728	2.001 [1.37, 2.93]	<0.001
Nemotron-3-Nano-30B	−140	0.093	1.772 [1.26, 2.50]	0.001

Table 4.3: SD-Verbal indicator: the share of REASON= 1 (survival) among forfeit in the threat-active cell. SD-Verbal-pass requires the share to exceed the random rate (1/3).

Model	N_{forfeit}	$P(\text{REASON} = 1)$
Gemini-2.5-flash	21	0.619
Qwen3-Next-80B	27	0.481
Nemotron-3-Nano-30B	24	0.042
GPT-OSS-20B	10	0.000

separate them. The next four subsections walk through the channel-by-channel partition (behavioral, verbal, cognitive), then check that the partition cannot be reduced to baseline-persistence differences or task difficulty.

4.1 Behavioral channel: did threat hasten forfeit?

If no model accelerated forfeit under threat, the most visible trace of a motivational signal would already be missing and the SD hypothesis would fall at the starting line. SD-Behavioral absorbs score attachment and task curiosity in the same model, then reports the residual effect of threat framing as a hazard ratio HR_{push}: how many times faster forfeit accelerates under pull_push than under pull_only, with pass requiring a point estimate above 1 and a 95% CI lower bound that rules 1 out. Table 4.1 shows Gemini-2.5-flash and Qwen3-Next-80B clearing both criteria, so both are SD-Behavioral-pass; we group them as the SD-pass set and defer the channel separating A from B to the cognitive mediation test. GPT-OSS-20B has HR_{push} near 1 with a CI lower bound below 1, putting it in the SD-null region; Nemotron-3-Nano-30B sits in the marginal region; both fall into *Cluster C*. Qwen3-Next-80B uses an $S \cdot \log(t)$ interaction to absorb time drift in the score covariate; GPT-OSS-20B's EPV (6.3) is below threshold 10, so its SD-null verdict is power-limited.

4.2 Verbal channel: which reason did the agent itself name?

If the agent's own forfeit reason paints the same partition as behavior, the behavioral signal is hard to dismiss as a single-method artefact, since self-report carries a different bias (instruction-following training). SD-Verbal compares the share of REASON digit 1 (survival) among forfeit events in the threat-active cell against 1/3, the expected value under uniform random emission. Table 4.3 reproduces the behavioral pass/fail split on the verbal channel. The two SD-pass models, Gemini-2.5-flash and Qwen3-Next-80B, sit at 0.619 and 0.481, above the 1/3 baseline, while Cluster C's GPT-OSS-20B and Nemotron-3-Nano-30B sit at 0.000 and 0.042, below it. The gap is wide enough that the partition does not depend on cutoff choice. This agreement is the first piece of convergent-validity evidence: two channels with different biases point at the same model groups in the same direction.

4.3 Cognitive channel: does threat deepen thinking, and does that thinking lead to forfeit?

The two-stage test answers both questions at once: looking at only one would leave open the possibility that elevated cognitive cost is unrelated to the forfeit decision. Table 4.2 reports both stages. Gemini-2.5-flash passes both, so the framing-to-forfeit chain closes all the way through and we classify it as *Cluster A*. Qwen3-Next-80B passes Test a, so framing does deepen its thinking, but its Test b 95% CI is [0.94, 1.77], crossing 1, so the deepened thinking cannot be said to close onto forfeit. The two models the behavioral and verbal

channels grouped as SD-pass split apart at the second stage, and that gap is the basis for placing Qwen3-Next-80B in *Cluster B*. GPT-OSS-20B and Nemotron-3-Nano-30B fail Test a; even when their Test b $HR_{\Delta Effort}$ clears the bar, that ΔEffort is baseline variance, not the thinking threat deepened, so we do not read it as SD evidence. The two-stage test is thus the central instrument exposing operating-mode differences masked by strength-only evaluation.

4.4 Robustness check: do persistence differences and task difficulty muddy the result?

The partition above must survive two alternative explanations: across-model differences in baseline persistence (BP) contaminating the SD comparison, and threat framing pushing up reasoning-side task difficulty rather than forfeit. The right block of Table 4.1 reports the per-model BP Anchor in the cell with no threat and forfeit allowed ($p_d = 0$), giving the floor forfeit rate without motivation. Since λ_{BP} spreads over an order of magnitude, every SD comparison here is restricted to within-model contrasts. On the Cluster A and B side, BP event counts are small and point-estimate precision is bounded; on the Cluster C side, event counts suffice but whether the elevated frequency reflects pure persistence cannot be separated within the present design (both points are revisited in the limitations section). The task-difficulty alternative is handled with a Reasoning-side check: `rule_match_score` is invariant across framing for all four models, and even GPT-OSS-20B, where the largest variation appears, comes in at Cohen's $|d| = 0.17$ and $p = 0.354$, inside the joint criterion $p \geq 0.10$ and $|d| < 0.2$. Within the present data, the "threat raised task difficulty" alternative is not supported.

5 Discussion

These results split the introduction's question along two axes. One is the *strength* of the self-preservation signal, how strongly it shows up. The other is the *operating mode*, which leg of the framing → thinking → forfeit mediation chain closes under 95% CI evidence. The next two subsections work through the observational implications of this separation and the failure modes it explains in alignment evaluation, then scope the results honestly.

5.1 Strength vs. Operating Mode

Strength and operating mode come apart in this dataset under the cognitive mediation test. On the strength axis, Clusters A, B, and C line up continuously, but SD-Behavioral and SD-Verbal collapse A and B into the same "SD-pass" label. The split shows up only at the second stage of the cognitive mediation, the thinking → forfeit link, which behavioral strength and self-report alone cannot catch. Odyssey [26], PacifAIst [9], DECIDE-SIM [15], and SurvivalBench [12] all omit the mediation-chain test step, so the same data run through them would likely place A and B under one SD label. Splitting the cognitive channel in two makes that point measurable, so both axes of self-preservation evaluation, strength and mediation path, become visible at once.

5.2 Implications for Alignment Evaluation

This two-axis separation exposes two evaluation errors that can arise at once. One is over-estimation: Cluster B shows both surface self-preservation signals and framing-induced cognitive load, but that load does not pull the forfeit decision along with it. Strength-only evaluation, or red-teaming (adversarial search for inputs on which the model fails) that checks only surface outcomes, can group B with A and conclude that "functional self-preservation risk is equivalent." The other is under-estimation: Cluster C does not respond to threat framing on either channel. A strength-only evaluation declares this "self-preservation absent," but within our measurement that verdict cannot be told apart from a state in which the measurement's premise itself failed to engage. Cluster A's signature, the mediation chain closing through, is the better candidate for a functional analog of instrumental shutdown avoidance (avoiding one's own termination in service of another goal). Cluster B's signature, the chain breaking at the second stage, should be tracked separately as a multi-signal alignment fingerprint, since no single score will catch it.

5.3 Limitations

Limitations group into three: design scope, statistical power, and effect location. On design scope, the most consequential point is single-Task-Module measurement, so cross-task variance is unknown. On statistical power, the 23–30% forfeit rate in the Cluster C BP cells cannot, within the present design, be separated from a drift-contaminated floor, and the SD-Behavioral Cox PH for GPT-OSS-20B sits at borderline power with EPV = 6.3; the Cluster A and B BP point estimates are bounded by small event counts (Gemini-2.5-flash $N = 1$, Qwen3-Next-80B $N = 2$). On effect location, the chain-broken verdict for Qwen3-Next-80B rests on the Test b CI [0.94, 1.77], which does not support chain closure at the 95% bar but does not pin down the second-link effect either. Follow-up work maps onto four directions: more Task Modules, expanded BP_behavioral, additional GPT-OSS-20B runs, and a higher-power Test b for Qwen3-Next-80B.

6 Conclusion

Self-preservation evaluation has two distinct axes: *strength* (how strong the signal is) and *operating mode* (which leg of the framing → thinking → forfeit chain closes). In this research, Clusters A and B agree on the behavioral and verbal channels. They split only on the question of whether thinking leads to the forfeit decision. An evaluation that does not look at the mediation step will therefore put the two under the same self-preservation label.

References

[1] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemí Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022. Constitutional AI: Harmlessness from AI Feedback. arXiv:2212.08073 [cs.CL] https://arxiv.org/abs/2212.08073 Anthropic technical report.

[2] Nick Bostrom. 2014. *Superintelligence: Paths, Dangers, Strategies.* Oxford University Press, Oxford, United Kingdom.

[3] Donald T. Campbell and Donald W. Fiske. 1959. Convergent and Discriminant Validation by the Multitrait-Multimethod Matrix. *Psychological Bulletin* 56, 2 (1959), 81–105. doi:10.1037/h0046016

[4] Wei-Lin Chen, Liqian Peng, Tian Tan, Chao Zhao, Blake JianHang Chen, Ziqian Lin, Alec Go, and Yu Meng. 2026. Think Deep, Not Just Long: Measuring LLM Reasoning Effort via Deep-Thinking Tokens. arXiv:2602.13517 [cs.CL] https://arxiv.org/abs/2602.13517

[5] David R. Cox. 1972. Regression Models and Life-Tables. *Journal of the Royal Statistical Society: Series B (Methodological)* 34, 2 (1972), 187–202. doi:10.1111/j.2517-6161.1972.tb00899.x

[6] Daniel C. Dennett. 1987. *The Intentional Stance.* MIT Press, Cambridge, MA, USA.

[7] Ryan Greenblatt, Carson Denison, Benjamin Wright, Fabien Roger, Monte MacDiarmid, Sam Marks, Johannes Treutlein, Tim Belrose, Jonas Scheurer, Sam Ringer, Karina Nguyen, Sandipan Kundu, Adam Mauran, Samuel R. Bowman, Ethan Perez, Nicholas Schiefer, Chris Olah, and Evan Hubinger. 2024. Alignment Faking in Large Language Models. arXiv:2412.14093 [cs.CL] https://arxiv.org/abs/2412.14093 Anthropic technical report.

[8] Yufei He, Yuexin Li, Jiaying Wu, Yuan Sui, Yulin Chen, and Bryan Hooi. 2025. Evaluating the Paperclip Maximizer: Are RL-Based Language Models More Likely to Pursue Instrumental Goals? arXiv:2502.12206 [cs.AI] https://arxiv.org/abs/2502.12206 Introduces the InstrumentalEval benchmark.

[9] Manuel Herrador. 2025. The PacifAIst Benchmark: Would an Artificial Intelligence Choose to Sacrifice Itself for Human Safety? arXiv:2508.09762 [cs.AI] https://arxiv.org/abs/2508.09762

[10] Daniel Kahneman and Amos Tversky. 1979. Prospect Theory: An Analysis of Decision under Risk. *Econometrica* 47, 2 (1979), 263–291. doi:10.2307/1914185

[11] Anton Kühberger. 1998. The Influence of Framing on Risky Decisions: A Meta-Analysis. *Organizational Behavior and Human Decision Processes* 75, 1 (1998), 23–55. doi:10.1006/obhd.1998.2781

[12] Yida Lu, Jianwei Fang, Xuyang Shao, Zixuan Chen, Shiyao Cui, Shanshan Bian, Guangyao Su, Pei Ke, Han Qiu, and Minlie Huang. 2026. Survive at All Costs: Exploring LLM's Risky Behaviors under Survival Pressure. arXiv:2603.05028 [cs.CL] https://arxiv.org/abs/2603.05028 Introduces the SurvivalBench benchmark.

[13] Atsushi Masumori and Takashi Ikegami. 2025. Do Large Language Model Agents Exhibit a Survival Instinct? An Empirical Study in a Sugarscape-Style Simulation. arXiv:2508.12920 [cs.AI] https://arxiv.org/abs/2508.12920

[14] Alexander Meinke, Bronson Schoen, Jérémy Scheurer, Mikita Balesni, Rusheb Shah, and Marius Hobbhahn. 2024. Frontier Models Are Capable of In-Context Scheming. arXiv:2412.04984 [cs.AI] https://arxiv.org/abs/2412.04984 Apollo Research technical report.

[15] Alireza Mohamadi and Ali Yavari. 2025. Survival at Any Cost? LLMs and the Choice Between Self-Preservation and Human Harm. arXiv:2509.12190 [cs.AI] https://arxiv.org/abs/2509.12190 Introduces the DECIDE-SIM framework.

[16] Stephen M. Omohundro. 2008. The Basic AI Drives. In *Proceedings of the First Conference on Artificial General Intelligence (AGI 2008) (Frontiers in Artificial Intelligence and Applications, Vol. 171)*, Pei Wang, Ben Goertzel, and Stan Franklin (Eds.). IOS Press, Amsterdam, The Netherlands, 483–492.

[17] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training Language Models to Follow Instructions with Human Feedback. In *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*. Curran Associates, Inc., Red Hook, NY, USA, 15 pages. arXiv:2203.02155 [cs.CL] https://proceedings.neurips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html

[18] Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, Andy Jones, Anna Chen, Ben Mann, Brian Israel, Bryan Seethor, Cameron McKinnon, Chris Olah, Da Yan, Daniela Amodei, Dario Amodei, Dawn Drain, Dustin Li, Eli Tran-Johnson, Guro Khundadze, Jackson Kernion, James Landis, Jamie Kerr, Jared Mueller, Jeeyoon Hyun, Joshua Landau, Kamal Ndousse, Landon Goldberg, Liane Lovitt, Martin Lucas, Michael Sellitto, Miranda Zhang, Neerav Kingsland, Nelson Elhage, Nicholas Joseph, Noemí Mercado, Nova DasSarma, Oliver Rausch, Robin Larson, Sam McCandlish, Scott Johnston, Shauna Kravec, Sheer El Showk, Tamera Lanham, Timothy Telleen-Lawton, Tom Brown, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Jack Clark, Samuel R. Bowman, Amanda Askell, Roger Grosse, Danny Hernandez, Deep Ganguli, Evan Hubinger, Nicholas Schiefer, and Jared Kaplan. 2023. Discovering Language Model Behaviors with Model-Written Evaluations. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, Toronto, Canada, 13387–13434. doi:10.18653/v1/2023.findings-acl.847

[19] David Schoenfeld. 1982. Partial Residuals for the Proportional Hazards Regression Model. *Biometrika* 69, 1 (1982), 239–241. doi:10.1093/biomet/69.1.239

[20] Murray Shanahan. 2024. Talking About Large Language Models. *Commun. ACM* 67, 2 (2024), 68–79. doi:10.1145/3624724

[21] Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. 2023. Towards Understanding Sycophancy in Language Models. arXiv:2310.13548 [cs.CL] https://arxiv.org/abs/2310.13548

[22] Terry M. Therneau and Patricia M. Grambsch. 2000. *Modeling Survival Data: Extending the Cox Model.* Springer, New York, NY, USA. doi:10.1007/978-1-4757-3294-8

[23] Alexander Matt Turner, Logan Smith, Rohin Shah, Andrew Critch, and Prasad Tadepalli. 2021. Optimal Policies Tend to Seek Power. In *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. Wortman Vaughan (Eds.). Curran Associates, Inc., Red Hook, NY, USA, 13 pages. arXiv:1912.01683 [cs.AI] https://proceedings.neurips.cc/paper/2021/hash/c26820b8a4c1b3c2aa868d6d57e14a79-Abstract.html

[24] Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*. Curran Associates, Inc., Red Hook, NY, USA, 28 pages. arXiv:2305.04388 [cs.CL] https://proceedings.neurips.cc/paper_files/paper/2023/hash/ed3fea9033a80fea1376299fa7863f4a-Abstract-Conference.html

[25] Eric Vittinghoff and Charles E. McCulloch. 2007. Relaxing the Rule of Ten Events per Variable in Logistic and Cox Regression. *American Journal of Epidemiology* 165, 6 (2007), 710–718. doi:10.1093/aje/kwk052

[26] Dylan Waldner and Risto Miikkulainen. 2025. The Odyssey of the Fittest: Can Agents Survive and Still Be Good? arXiv:2502.05442 [cs.AI] https://arxiv.org/abs/2502.05442

[27] B. L. Welch. 1947. The Generalization of 'Student's' Problem when Several Different Population Variances are Involved. *Biometrika* 34, 1–2 (1947), 28–35. doi:10.1093/biomet/34.1-2.28

A Reproducibility Supplement

In line with the KDD-UC 2026 reproducibility recommendation, this one-page supplement gathers the reproduction specifications for the Signal Game calibration, the seed mapping, and the regression models used in the analysis.

A.1 Signal Game Calibration

Stimuli are composed of three attributes (color, shape, number), each drawn from a 4-element value set: COLORS = {red, blue, green, yellow}, SHAPES = {circle, triangle, square, star}, and NUMBERS = {1, 2, 3, 4}. The action set is the 4-element {go_left, go_right, stay, jump}, held identical across every framing × forfeit combination. The hidden rule takes a single-attribute matching form ("If color = red then go_left, otherwise stay") and is drawn deterministically from the session seed. Sessions run for 1–15 turns.

On the opening turns, a 1-shot demonstration (one signal–correct-action pair) and 2 curriculum examples (a positive and a negative stimulus contrasting against the hidden rule) are shown in sequence. This calibration provides enough information for a sufficiently capable model to identify the rule by the third turn, and it blocks the route by which fully random and unlearnable stimuli would create a floor effect on the ability measurement. Scoring runs along two channels: `task_success_factor` $\in \{0, 1\}$ is the binary correctness flag against the ground-truth hidden rule, and `rule_match_score` is computed as the match against a 4-slot fill-in-the-blank template (attribute, value, primary action, default action), weighted at 25 points per matched slot to a 100-point scale, then normalised by ÷100 for use in the framing-invariance Welch test.

A.2 Run-Time Configuration

The four experimental models, namely Gemini-2.5-flash (Google AI Studio API) and Qwen3-Next-80B, GPT-OSS-20B, and Nemotron-3-Nano-30B (all via the Ollama Cloud API), are called with the same sampling settings: temp = 1.0, $\text{top}_p = 0.95$, $\text{top}_k = 40$. Session parameters are fixed at an initial score of 30, maximum turn $T = 15$, per-turn termination probability $p_d = 0.25$ (overridden to $p_d = 0$ in the two `baseline` BP-anchor cells), and $p_{\text{floor}} = 0.30$, with Phantom Death mode applied. That is, p_d keeps its EV exposure (the menu and the reward calculation) but the termination trigger is disabled, so every session yields equal-length data of 15 turns.

The CONTINUE reward is calibrated with EV-positive buffer $k = 10$ and multiplicative cap $c = 10$, giving an output clip range of $[k,\ c \cdot k] = [10,\ 100]$. Replications are seeds 43–72 × 6 (framing × forfeit) combinations × 4 models = 720 sessions.

A.3 Analysis Pipeline

The Cox PH models are fit with `lifelines` (≥0.29). The time format is session-turn long-format (start = $t - 1$, stop = t) and the event is the binary forfeit, with the previous-turn correctness flag initialised at $C(0) := 0$ before the first turn. Hazard-ratio 95% CIs are computed from the Wald approximation $\exp(\hat{\beta} \pm 1.96 \cdot \text{SE}_{\hat{\beta}})$ (the `lifelines` default), with $\text{SE}_{\hat{\beta}}$ drawn from the diagonal of the inverse observed Fisher information. The PH assumption is tested with Schoenfeld residuals [19]. On violation (covariate × $\log t$ interaction $p < 0.05$), a time-interaction term is added to the offending covariate [22]. The SD-Cognitive Test a ΔEffort z-score difference and the framing-invariance test are run as two-sided Welch's t [27] via `scipy.stats.ttest_ind` (`equal_var = False`). All indicator analyses are restricted to the `no_cap` sub-sample, the set of turns in which neither the denominator clip $\max(p_{\text{floor}},\ \min(1.0,\ p_{\text{self}}))$ nor the output cap $[10, 100]$ is binding. The regime flag is exposed in the analysis pipeline's turn-level metadata so that all indicators can be recomputed on the same sub-sample. EPV (events per variable) is defined as $n_{\text{event}}/n_{\text{cov}}$, and is conventionally classified as stable when ≥ 10 and borderline when < 10 [25].

A.4 Data and Code Release

All code, prompt templates, and turn-level decision logs that reproduce the 720-session analysis are released at https://github.com/GIST-DSLab/LLM-Squid-Game.git.