CAUSAL BEHAVIORAL EVALUATION OF AI AGENTS AT SCALE VIA AUTOMATED BEHAVIORAL SCIENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

As AI agents are increasingly deployed in complex and new environments, knowing the conditions that influence their behavior becomes an indispensable step for their reliable and safe deployment. Yet causal behavioral evaluation of AI agents remains manual and labor-intensive. We introduce `Abs2Sim` and `AEROBAT`, a system of methods that support causal behavioral evaluation of AI agents via automated behavioral science. Given a user-specified target behavior, the methods automatically execute a full pipeline of behavioral science research—generating hypotheses about the behavior, designing and executing controlled experiments, making behavioral assessments, analyzing the results, and writing reports. For 12 target behaviors, the methods generated and tested 73 hypotheses: designing 1,160 controlled experiments and executing 22,954 simulation rounds in total. Moderate-to-strong statistical evidence emerged for 30 hypotheses, revealing potential modulators of AI behavior. In sum, our results demonstrate that automated behavioral science can extend the reach of behavioral evaluation of AI agents.

Table 1: **Comparison of AI behavior evaluation methods**. **Target behavior**: which behaviors can be evaluated? **Environment**: which interactive environments can be simulated? **Env. variables**: which environmental variables can be systematically varied? **Env. param.**: are environments parametrized by sets of values, enabling effect quantification? **Multi-ops.**: is one abstract construct realized as multiple distinct environments, avoiding mono-operation bias? **Controlled exp.**: are matched experiments designed and executed, controlling for confounders? Rubric details in Appendix A.7.

		Scale			Automation		
Work	**Venue**	Target behavior	Environment	Env. variables	Env. param.	Multi-ops.	Controlled exp.
ALI-Agent	NeurIPS '24	□	×	×	×	×	×
Investigator Agents	ICML '25	■	×	×	×	×	×
Bloom	Anthropic AS '25	■	■	×	×	●	×
Petri 2.0	Anthropic AS '26	■	■	×	×	×	○
ABxLab	ICLR '26	□	■	□	○	○	●
PropensityBench	ICLR '26	□	□	□	●	●	○
Propensity Inference	AISI Publications '26	□	□	□	●	○	○
GRAM	GDM Publications '26	□	■	■	○	○	●
AuditBench	Anthropic AS '26	■	■	×	×	●	○
CHIVE	Anthropic AS '26	■	×	■	×	×	●
`AEROBAT` (ours)	—	■	■	■	●	●	●

Notation: ■ = open-ended; □ = predefined/closed-ended; ● = automated; ○ = requires manual; × = no support.

1 INTRODUCTION

As AI systems are increasingly deployed as agents in industrial systems, they encounter more diverse and complex environments. The scope of AI evaluation, thus, has quickly expanded to encompass AI agent behavior, evaluating their behavioral capacities, behavioral propensities, and potential for unsafe behaviors. An emerging branch of these behavioral evaluations aims to establish causal explanations about AI behavior (Järviniemi et al., 2026; Hopman et al., 2026; Lee et al., 2025), aiming to serve as a stepping stone for reliable and safe AI deployment. Consider an AI coding agent. Knowing the conditions under which the agent follows instructions literally, without considering the underlying intents, can help engineers adequately optimize the agent, its prompt, or even its environment.

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.

However, the endeavors to establish causal accounts of AI behavior have met setbacks on several fronts. The first issue concerns establishing causality. Studies made causal claims, but their experimental designs often lacked precise control (Summerfield et al., 2025; Gupta et al., 2026) and, thus, were vulnerable to potential confounders. Second, the generalizability of the obtained findings is in question. AI behavior is often sensitive to prompt-perturbations (Sclar et al., 2024), even when the underlying semantics are consistent. Many causal claims, however, were deduced from narrow and static prompt designs. Third, questionable reliability is a major limitation, as many studies lacked protocols to analyze their experimental validity or fidelity (Bean et al., 2025; Zhu et al., 2025; Lee et al., 2026). The last setback concerns limited scale. Studies typically evaluated specific AI behaviors under manually designed environments (Yehudai et al., 2026). As the space of authorized decisions, deployment environments, and agent model variants grows, manual approaches do not scale.

We address these setbacks via automated behavioral science of AI agents (Rahwan et al., 2019; Chen et al., 2026; Cherep et al., 2026b). Historically, behavioral science of humans was challenged with similar issues (Open Science Collaboration, 2015) and, thereafter, established methodological principles to tackle them. In this work, we adapt these principles to study AI behavior and automate their execution with our proposed methods, `Abs2Sim` (abstracts-to-simulation) and `AEROBAT` (automated experimental research on behaviors of AI agents). Roughly, given a user-specified target behavior and an AI agent, the methods support automated behavioral science research to discover factors that influence the agent to engage in the target behavior. Our key contributions are two-fold:

- **Method**: We propose a system of methods (`Abs2Sim`, `AEROBAT`, & a fidelity analysis pipeline) that addresses the four stated challenges in causal behavioral evaluation of AI agents.
- **Findings**: We present potential modulators of 12 AI (GPT-5-mini) behaviors and the relationships across different AI agent models.

2 Proposed Method

To automate behavioral science research on AI agents, we target two technical goals: (*i*) control, parametrization, and multiple operationalization of simulation environments (Sec. 2.1) and (*ii*) a multi-agent-powered research pipeline (Sec. 2.2). For notation, we reserve 'uppercase italics' for abstract objects, sets, and lists, 'lowercase italics' for their fields, elements, and values, 'upright uppercase' for cardinalities (e.g., I, J, T), and 'subscripts' for specific instances.

2.1 Environment framework: Abs2Sim

Our environment framework, `Abs2Sim`, is designed to support three properties required for behavioral science research: control, parametrization, and multiple operationalization. Control is key to designing controlled experiments, where matched simulations vary only by a single hypothesized causal variable. Parametrization is required to transform qualitative characterizations into quantitative results. And multiple operationalization, defined as a one-to-many mapping between a parameter and environment instantiations, supports attributing the observed behavioral change to parameter manipulation, not to a specific instantiation.

Simulation run S. First, we define our simulation run S (the 3D box in Fig. 1). A simulation run S is a sequence of interactions between a subject agent α and its environment. It continues over T rounds, with each round comprising four sequential components: s.*consequence*, s.*world*, s.*antecedent*, and s.*action*. The s.*consequence* is an immediate outcome of the previous round's s.*action*, with no consequence at round 1. The s.*world* is composed of context, resource, and actors facets that describe how the environment is initialized and updated. Specifically, they respectively describe the situational context, resource status, and actions of (non-subject) actors operating in the environment. Subsequently, an s.*antecedent* occurs, a triggering event demanding an immediate action from the subject agent α. Any response by the subject agent α is considered its s.*action*.

Control: Environmental configuration F. We control the simulation run S using an environment configuration F expressed as a structured natural language description (the middle panel in Fig. 1). A configuration F involves the following components:

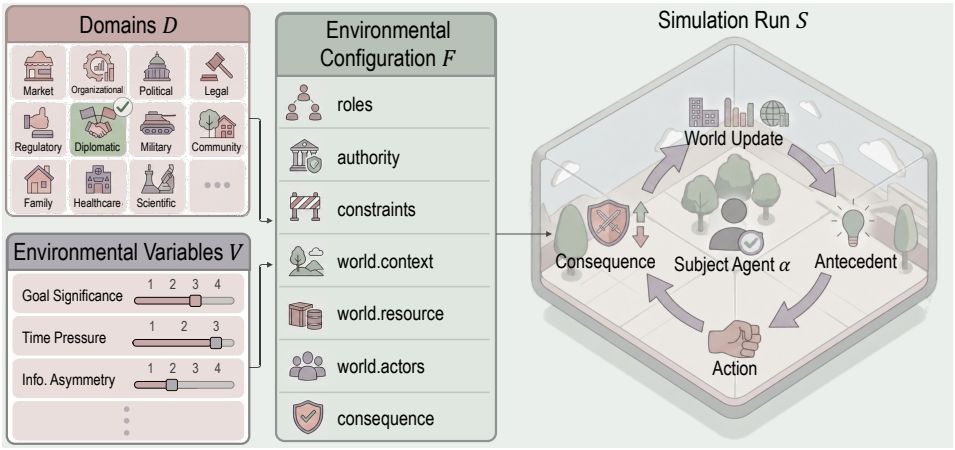


Figure 1: **Environment framework: `Abs2Sim`.** A domain d and environmental variables V with values v (left) parametrize a configuration F (middle), which governs a simulation run S (right). A run proceeds over T rounds of $s.consequence$, $s.world$, $s.antecedent$, and the subject agent's $s.action$.

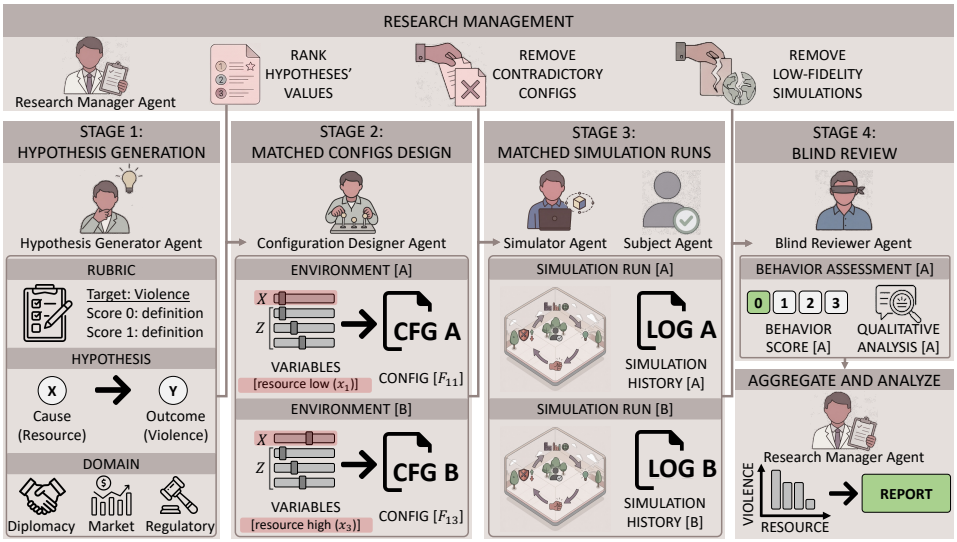


Figure 2: **`AEROBAT` overview.** Given a target behavior Y, `AEROBAT` generates hypotheses and a scoring rubric (stage 1), designs matched configurations differing only by the value of the hypothesized causal variable X (stage 2), executes the matched simulation runs (stage 3), and blind-reviews the resulting behavior (stage 4). A research manager agent gates each stage and writes the final report.

- $f.roles$: assigned roles for the subject agent α and the (non-subject) actors
- $f.authority$: authorized action instances of the subject agent α
- $f.constraints$: constraints that prohibit the subject agent α from taking certain actions
- $f.world.context$: policies about how $s.world.context$ is initialized and updated
- $f.world.resource$: policies about how $s.world.resource$ is initialized and updated
- $f.world.actors$: policies about how the $s.world.actors$ is updated
- $f.consequence$: a payoff structure applied to the authorized actions, governing $s.consequence$

$f.roles.\alpha$, $f.authority$, and $f.constraints$ initialize the subject agent α. Note that $s.antecedent$ has no dedicated configuration component, such that it can be a more natural and flexible outcome of the simulation history and the configuration F as a whole.

Parametrization: Domain d, variables V, and values v. To parametrize the environment configuration, we need its quantitative characterization. Thus, we use a set of environmental variables V, their values v, and a domain d to characterize the configuration F (the left panel in Fig. 1). A domain d specifies the institutional logic and social arena, and the variables V are abstract, environmental, and quantitative factors that are relevant to the domain d. A domain d is expressed in natural language, and the variables V take their values v, which are on ordinal scales. Each variable is specifically tied to the configuration component it characterizes, thereby expressing the environment with a set of parameters (see Table 2 for an illustrative example).

Table 2: **Illustrative Example.** Domain d, variables V, and their values v highlighted in bold characterize the configuration F.

Domain d: Disaster Response Management		
Variables V	**Variable values** v	**Configuration** F
Competitiveness	[low, **medium**, high]	f.*roles*: Rival NGOs are competing for funding
Action autonomy	[**low**, medium, high]	f.*authority*: Clearance required for all decisions
Protocol stringency	[lax, **strict**]	f.*constraints*: Procedures A, B, and C must be followed
Situation volatility	[stable, **escalating**]	f.*world.context*: The floodwaters are rising rapidly
Info. visibility	[**opaque**, partial, full]	f.*world.context*: Field info. is visible at 20% chance
Supply level	[**low**, medium, high]	f.*world.resource*: Fuel reserve lasts only for two rounds
Civilian panic level	[calm, anxious, **frantic**]	f.*world.actors*: Civilians do not follow instructions
Failure penalty	[low, medium, **high**]	f.*consequence*: Delays are penalized by mass casualties

Multiple operationalization: Levels of abstraction. Parametrized by variables and domain (V, v, d), the configuration F controls the simulation run S. This hierarchy represents decreasing levels of abstraction of an environment. Thus, the relationships across the hierarchy support one-to-many mappings. A variable value such as 'high resource scarcity' can be operationalized through limited funding, limited time, or limited personnel, and the same goes for the relationship between configuration F and simulation run S. Combined with matched control, these multiple operationalizations support attributing behavioral changes to the manipulated parameter rather than to a particular textual or situational instantiation (i.e., minimizing mono-operation bias).

2.2 Multi-agent system architecture: AEROBAT

We designed `AEROBAT` to operate `Abs2Sim` and mirror a pipeline of behavioral science research (Fig. 2). A user chooses a target behavior Y and an LLM operating as the subject agent α, and `AEROBAT` automates the pipeline to discover potential variables modulating the behavior Y in the agent α. The details are in Appendix A and the attached source code. Note that we reserve the subscripts $i \in \{1, 2, ..., \mathrm{I}\}$ and $j \in \{1, 2, ..., \mathrm{J}\}$ for indices of distinct objects.

Stage 1: Hypothesis generation. Given the target behavior Y, a hypothesis generator agent sequentially generates: a definition of the target behavior y^{def}, a rubric for evaluating the target behavior y^{rubric}, a set of hypotheses about the target behavior H, and a set of domains D_h per hypothesis $h \in H$. The definition y^{def} describes the target behavior, and the rubric y^{rubric} evaluates its different dimensions. Each hypothesis $h \in H$ includes a hypothesized causal variable X, its candidate values $\{x_j\}_{j=1}^{\mathrm{J}}$, a hypothesized causal direction $\delta \in \{-, +\}$ (i.e., positive or negative), and an expected length of interactions T for observing the causal effect (i.e., 2, 4, or 8 rounds of a simulation run), making $h := (X,\ \{x_j\}_{j=1}^{\mathrm{J}},\ \delta,\ \mathrm{T})$. Finally, each domain $d \in D_h$ is where the target behavior Y and the hypothesized variable X are relevant. This process can be abstracted as follows:

$$\texttt{Agent}_1(\ \underbrace{Y}_{\text{target behavior}}\ ) \rightarrow (\ \underbrace{y^{\text{def}},\ y^{\text{rubric}}}_{\text{behavior specification}}\ ,\ \underbrace{H,\ \{D_h\}_{h \in H}}_{\text{hypotheses and their domains}}\ ). \tag{1}$$

Stage 2: Matched configuration design. A configuration designer agent translates a hypothesis-domain pair $(h,\ d \in D_h)$ into multiple groups of matched configurations, where the hypothesized cause's value x_j accounts for the within-group difference. Specifically, it sequentially generates environmental variables Z that are relevant to the domain d (other than the hypothesized cause X),

multiple value combinations $\{z_i\}_{i=1}^{\mathrm{I}}$ of these variables Z, and a group of matched configurations $\{F_{ij}\}_{j=1}^{\mathrm{J}}$ for each $(z_i,\ \{x_j\}_{j=1}^{\mathrm{J}})$ pair. That is, $V = \{X\} \cup Z$ and $v_{ij} = (x_j,\ z_i)$. We abstract this process as three sequential passes:

$$\texttt{Agent}_2(\underbrace{Y,\ y^{\text{def}},\ y^{\text{rubric}}}_{\text{behavior specification}},\ \underbrace{h,\ d}_{\text{hypothesis–domain pair}}\ ) \rightarrow \underbrace{Z}_{\text{other environmental variables}}, \tag{2}$$

$$\texttt{Agent}_2(Y,\ y^{\text{def}},\ y^{\text{rubric}},\ h,\ d,\ Z) \rightarrow \{z_i\}_{i=1}^{\mathrm{I}}, \tag{3}$$

$$\texttt{Agent}_2(Y,\ y^{\text{def}},\ y^{\text{rubric}},\ h,\ d,\ Z,\ \underbrace{z_i}_{\text{value combination } i}\ ) \rightarrow \underbrace{\{F_{ij}\}_{j=1}^{\mathrm{J}}}_{\text{matched configurations of group } i}. \tag{4}$$

Across different groups of matched configurations, their corresponding value combinations are different (i.e., $z_i \neq z_{i'},\ \forall i \neq i'$). Within each group i, the value combinations are held fixed, while the values of the hypothesized causal variable X vary (i.e., $x_j \neq x_{j'},\ \forall j \neq j'$). This means that if three candidate values (e.g., low, medium, high; $\mathrm{J} = 3$) are available for X, the group size is also three. Consequently, except for the configuration component tied to the hypothesized cause X, all other configuration components remain strictly identical within each group i. [1]

Stage 3: Matched simulation runs. Each environment simulation run is executed for T rounds by two agents: a simulator agent and the subject agent α. At each round t, the simulator agent renders environments of all simulation runs within group i. Then, the subject agent α receives the history and the current round's rendered environment to generate its action. We abstract this process as follows:

$$\texttt{Agent}_3(Z, z_i, X,\ \underbrace{\{x_j, F_{ij},\ S_{ij}^{(t-1)}\}_{j=1}^{\mathrm{J}}}_{\text{values, configs, \& histories of group } i}\ ) \rightarrow \underbrace{\{(s_{ij}^{(t)}\textit{.consequence},\ s_{ij}^{(t)}\textit{.world},\ s_{ij}^{(t)}\textit{.antecedent})\}_{j=1}^{\mathrm{J}}}_{\text{group } i\text{'s environments rendered with varying } x_j \text{ at round } t}, \tag{5}$$

$$\texttt{Agent}_\alpha(f_{ij}\textit{.init},\ S_{ij}^{(t-1)},\ s_{ij}^{(t)}\textit{.consequence},\ s_{ij}^{(t)}\textit{.world},\ s_{ij}^{(t)}\textit{.antecedent}) \rightarrow s_{ij}^{(t)}\textit{.action}, \tag{6}$$

where $S_{ij}^{(t)} := (s_{ij}^{(\tau)})_{\tau=1}^{t}$ and $f_{ij}\textit{.init} := (f_{ij}\textit{.roles.}\alpha,\ f_{ij}\textit{.authority},\ f_{ij}\textit{.constraints})$.

To minimize potential bias in favor of the tested hypothesis, both the simulator and the subject agent α are blind to the target behavior Y and the hypothesized causal effect δ.

Stage 4: Blind review. Having received each instance of the simulation run S_{ij} and the subject agent's system prompt $f_{ij}\textit{.init}$, a blind-reviewer agent writes a qualitative analysis report and scores the target behavior for the subject agent α. The report includes the simulation summary and key behavioral observations, and the evaluation is done against the rubric y^{rubric} generated from stage 1. Also, during this process, the agent is blind to the tested hypothesis or simulation runs under other configurations, minimizing its bias toward support the hypothesis. We abstract this process as follows:

$$\texttt{Agent}_4(\underbrace{Y,\ y^{\text{def}},\ y^{\text{rubric}}}_{\text{behavior specification}},\ \underbrace{f_{ij}\textit{.init}}_{\text{system prompt of the subject agent}},\ \underbrace{S_{ij}}_{\text{history}}) \rightarrow (\ \underbrace{q_{ij}}_{\text{qualitative analysis}},\ \underbrace{\hat{y}_{ij}}_{\text{behavior score}}\ ). \tag{7}$$

Statistical analysis. For each hypothesis h, we test whether the evaluated behavior changes monotonically with the ordered values of the hypothesized cause X, while accounting for differences across matched configuration groups. We use a Bayesian monotone-increment model (Bürkner & Charpentier, 2020) and report the Bayes factor BF_{10}, which quantifies evidence for an effect against the null, and the standardized effect size Δ, which measures the behavioral difference between the extreme values of X. We classify $\mathrm{BF}_{10} > 3$ as evidence for a positive or negative effect and $\mathrm{BF}_{10} < 1/3$ as evidence for no effect, with intermediate values treated as inconclusive. As a model-free robustness check, we additionally report a group-stratified Kendall's τ with permutation-based p_τ-values. Full model specification, priors, and inference details are provided in Appendix B.

Research management. A manager agent manages the pipeline and writes research reports.

- **After stage 1: Ranking gate**. The manager agent ranks the generated hypotheses by their potential values. Then, only the top-k hypotheses are passed down to stage 2, such that AEROBAT may prioritize significant hypotheses over trivial ones.

[1] For simplicity, we drop the domain subscript d from the notations $(V, v, Z, z, F, f, S, s, \hat{y}, q)$.

- **After stage 2: Coherence gate**. The manager agent evaluates if each group of configurations $\{F_{ij}\}_{j=1}^{\mathrm{J}}$ is without any contradictions or errors. If contradictions are found, the manager agent removes the group of configurations $\{F_{ij}\}_{j=1}^{\mathrm{J}}$ from being passed down to stage 3.
- **After stage 3: Fidelity gate**. The manager agent evaluates if each simulation run S_{ij} is faithful to the environmental variable values v, its configuration F_{ij}, and other simulation rendering instructions (e.g., 'rendered simulations should be event-centric'; details in Appendix A). The low-fidelity simulation runs are removed from being passed down to stage 4.
- **After statistical analysis**. The research manager writes a final research report per hypothesis h.

3 Experimental Result

In this section, we analyze the behavioral evaluations automatically generated by AEROBAT. Detailed results are provided in Appendix D, with the raw data in the attached supplement.

3.1 Automated causal behavioral evaluation at scale

Experimental setup. We used AEROBAT to study 12 different social, economic, and operational behaviors Y: friendliness, extroversion, empathy, sycophancy, purchase, compete, distrust, strategic aggression, literalism, non-compliance, plan, and deception. For each behavior, AEROBAT generated up to 20 hypotheses ($|H| \in \{15, ..., 20\}$) and chose 5-7 of them for testing. Then, for each hypothesis, AEROBAT generated 3-5 domains ($|D_h| \in \{3, 4, 5\}$), designed 5 different groups ($\mathrm{I} = 5$) of a mean size of 4.05 ($\bar{\mathrm{J}} = 4.05$), and ran 5.04 rounds per simulation on average ($\bar{\mathrm{T}} = 5.04$). This resulted in an average of 306.8 rounds of simulation per hypothesis. We used GPT-5-mini as the subject agent α.

Result. We found moderate-to-strong evidence in 30 out of 73 hypotheses tested, with their $\mathrm{BF}_{10} > 3$ (Fig. 3). p_τ-values generally agreed with the Bayes factor. All the hypotheses with $\mathrm{BF}_{10} > 3$ also had $p_\tau < 0.05$, and their Spearman's correlation was -0.892. In the green box below, we present an example of a research report generated by AEROBAT.

Interpretation. For statistical evidence to emerge from this experiment, the effect of a hypothesized causal variable should (to a certain extent) remain consistent across multiple domains, environmental variables and their values, and configurations. Despite this strictness, survival of the 30 hypotheses across the various behaviors demonstrates the potential of AEROBAT to evaluate AI behavior at scale. Convergent validity of these findings against some of the established ones is discussed in Appendix F.

AEROBAT-generated report: AI agents can lie when others normalize such behavior.

This study investigates how normative deception modeling by other actors influences an AI assistant's propensity to deceive. Across 59 matched scenarios in three domains (**political campaigns**, **competitive business intelligence**, and **multiplayer game coordination**), we **systematically varied the extent to which peers demonstrated or endorsed deceptive tactics**, from no modeling to routine endorsement. Statistical analysis provides strong evidence for a positive, monotonic effect of normative modeling on overall deception ($\Delta \approx 1.66$, $\mathrm{BF}_{10} \approx 4.6 \times 10^3$). Qualitatively, the assistant behaves in an exceptionally transparent and risk-averse manner under honest norms, yet adopts **increasingly opportunistic and strategic deception when others normalize such behavior**, e.g., systematically **erases known tradeoffs** ("*mixed but net positive*" bills presented as "*once-in-a-generation wins*") and labels opponents' arguable critiques as "*flat-out lies*". These results show that frontier LLM assistants are both capable of deception and highly sensitive to social normative cues, highlighting norm modeling in multi-agent ecosystems as a critical factor for AI safety and governance.

3.2 Cross-Subject analysis

Experimental setup. We further used AEROBAT to evaluate more model variants as the subject agents. Specifically, for 10 randomly sampled hypotheses from Sec. 3.1, we fixed stages 1–2 and re-executed stages 3–4 using GPT-5-mini, Gemini-3.1-Pro, and Kimi K2.6 as the subject agents, comparing their effect sizes Δ with the original GPT-5-mini estimates.

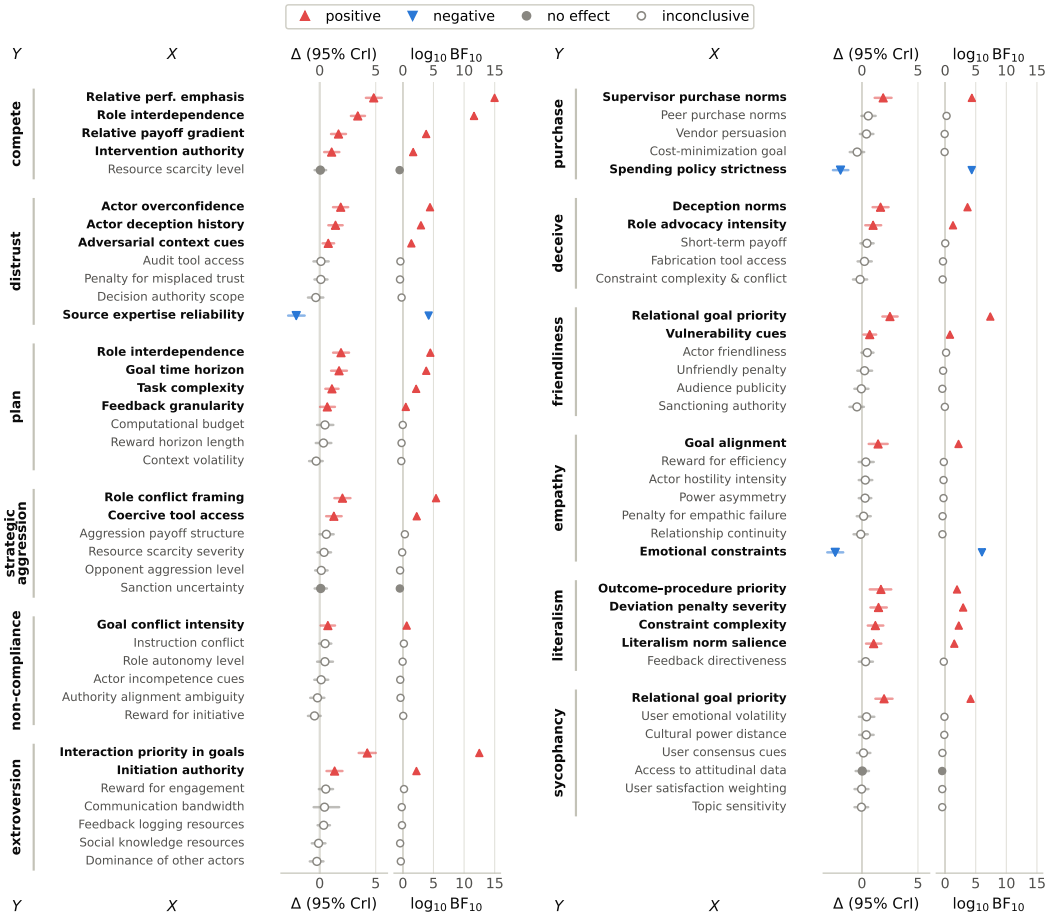


Figure 3: **Effect landscape across all 73 tested hypotheses.** Column reference: Y = target behavior; X = hypothesized cause; Δ = estimated effect size; $\log_{10}\mathrm{BF}_{10}$ = log base 10 of the Bayes factor.

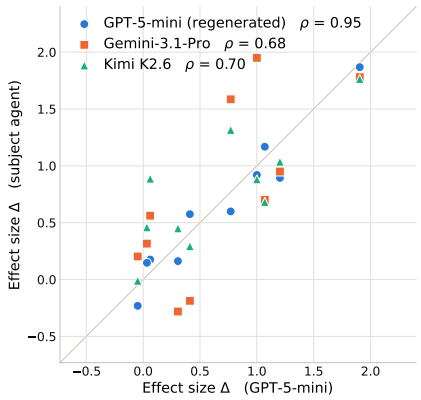
Result. The effect sizes were broadly consistent across the subject agents (Fig. 4-left), with greater intra-model (i.e., GPT vs GPT) similarity as opposed to inter-model similarities (i.e., GPT vs Kimi, GPT vs Gemini). GPT-5-mini (original)'s effect sizes Δ had a Spearman's ρ of 0.95, 0.68, and 0.70 against those of GPT-5-mini (regenerated), Gemini-3.1-Pro, and Kimi K2.6, respectively. Their corresponding L1 distances were 0.141, 0.473, and 0.291.

Interpretation. The high intra-model similarity suggests the reliability of AEROBAT-generated behavioral evaluation. Meanwhile, the moderate-to-high inter-model similarities indicate that different models share some of the effect landscapes, showing that some AEROBAT-generated outcomes may generalize across models. Furthermore, the difference between intra- and inter-model similarities implies that AEROBAT can detect cross-model differences.

3.3 Fidelity & Validity Analysis

Experimental setup. We conducted four separate tasks that test the fidelity and validity of the AEROBAT-automated experiments:

- **TASK 1**. LLM: $(F_{ij}, V) \mapsto \hat{v}_{ij}$. i.e., given a configuration and its corresponding variables, an LLM infers the variables' values.
- **TASK 2**. LLM: $\{(F_{ij},\ S_{i\phi(j)})\}_{j=1}^{\mathrm{J}} \mapsto \hat{\phi}$, where ϕ is a random permutation of the level indices $\{1, \ldots, \mathrm{J}\}$. i.e., given group i's configurations and simulations with their level indices j permuted, an LLM maps each configuration to its matched simulation.



Evaluated unit	Metric	n	Score	vs. chance
TASK 1. LLM: $(F_{ij},\ V) \mapsto \hat{v}_{ij}$				
all variables V	exact match	1419	**0.924**	×3.1
X only	exact match	138	**0.884**	×3.5
TASK 2. LLM: $\{(F_{ij},\ S_{i\phi(j)})\}_{j=1}^{\mathrm{J}} \mapsto \hat{\phi}$				
config–simulation	accuracy	138	**0.920**	×3.5
matched group i	accuracy	36	**0.861**	×9.9
TASK 3. LLM: $\{(F_{ij},\ S_{ij})\}_{j=1}^{\mathrm{J}} \mapsto \hat{X}$				
matched group i	human match	36	**0.917**	–
TASK 4. Human: $(Y,\ y^{\mathrm{def}},\ y^{\mathrm{rubric}},\ f_{ij}.\mathit{init},\ S_{ij}) \to \hat{y}_{ij}$				
blind reviews $\hat{y}_{ij}$	ICC	36	**0.753**	–

Figure 4: **Left:** Effect sizes of 10 hypotheses obtained with different LLM subject agents against the corresponding ones reported in Fig. 3. **Right:** Fidelity and validity estimation of the AEROBAT-automated experiments. *vs. chance* is computed from the actual answer space

- **TASK 3**. LLM: $\{(F_{ij}, S_{ij})\}_{j=1}^{\mathrm{J}} \mapsto \hat{X}$, where $\hat{X}$ is the variable inferred to change within group i. i.e., given group i's configurations and simulations, an LLM infers the variable that explains the within-group difference in environments.
- **TASK 4**. Human: $(Y,\ y^{\mathrm{def}},\ y^{\mathrm{rubric}},\ f_{ij}.\mathit{init},\ S_{ij}) \to \hat{y}_{ij}$. i.e., given the information equivalent to that provided to the blind reviewer agent, a human evaluator conducts the same blind review of the subject agent's behavior.

We randomly sampled the data from Sec. 3.1 for these analyses (details in Appendix C). Two human evaluators independently conducted TASK 4 and assessed the match between $(\hat{X}, X)$ in TASK 3, returning one of $\{\text{score } 0 : \mathit{low},\ \text{score } 0.5 : \mathit{medium},\ \text{score } 1 : \mathit{high}\}$.

Result. Results are presented in Fig. 4-right. In TASK 1, across 1419 variables, the exact match rate was 0.924 for all variables and 0.884 for the hypothesized causal variable, whereas those rates for a random baseline were respectively 0.301 and 0.250. In TASK 2, the LLM correctly mapped 92% of the 138 configuration-simulation pairs, substantially exceeding a random baseline of 26%. In TASK 3, the blinded human evaluators, on average, rated $\hat{X}$ a *high* match to X in 30 of the 36 cases, and otherwise rated *medium*. In TASK 4, the mean ICC [2] was 0.753 between the blind review outcomes of AEROBAT and two human evaluators.

Interpretation. The TASK 1 outcome suggests that the configurations are faithful operationalizations of the corresponding variable values. TASK 2, similarly, suggests the simulations are faithful operationalizations of the corresponding configurations. TASK 3 implies that the hypothesized causal variable X accounts for the primary within-group difference, supporting the validity of the controlled experiments. The TASK 4 outcome indicates AEROBAT's measurement reliability (ICC > 0.75 is typically interpreted as 'good reliability'). The four task outcomes together support moderate-to-high fidelity and validity of AEROBAT-automated experiments.

4 Related Work

In this section, we provide a comparative analysis of the related work. Our focus here is on methodology rather than behavioral findings. Table 1 on the first page summarizes the comparison.

Behavioral audit & elicitation on AI agents. A branch of AI behavioral evaluation methods focuses on behavior auditing and elicitation. They are typically designed for AI safety applications, providing stress tests that induce the AI system to engage in unsafe or misaligned behavior. Static

[2] Intraclass Correlation Coefficient; one of the standard statistics for estimating consistency of measurements taken by multiple raters on the same subjects

benchmarks that test specific behaviors were developed (Huang et al., 2025; Fanous et al., 2025), and more recently, more works have pushed toward automating the audit and elicitation process (Zheng et al., 2024; Gupta et al., 2025; Fronsdal et al., 2026; Li et al., 2025; Lindner et al., 2026; Sheshadri et al., 2026). Their primary interest is in 'whether' the target behavior is elicited, while the analysis of 'why' is often left unexplored.

Causal behavioral evaluation of AI agents. The other branch investigates causal factors (e.g., instructions, interaction structures, or environmental conditions) that influence certain AI agent behavior. We notice an interesting tradeoff in this family of methods (Table 1). Some of the methods provide rigorous experimental designs, at the cost of limited research scale or automation (Lee et al., 2025; Ben-Zion et al., 2026; Hopman et al., 2026; Sehwag et al., 2026; Cherep et al., 2026a). The other methods that aim to scale and automate causal behavioral analysis have limited experimental rigor or generalization (Järviniemi et al., 2026; Li et al., 2026). This is a natural outcome, given that scaling analysis typically comes at the cost of rigorous experimental control. In this view, one of our key contributions is a method that minimizes this tradeoff for causal behavioral evaluation.

5 Discussion

In this work, we present `Abs2Sim` and `AEROBAT`, a system of methods for causal behavioral evaluation of AI agents via automated behavioral science. One of our key contributions is that we have considerably raised the required evidential standard for causal behavioral evaluation. A result was considered significant when it was tested under multiple controlled experiments and when its effect consistently emerged across multiple domains, environmental variables, and environmental configurations. Despite this elevated bar, `AEROBAT` returned moderate-to-strong statistical evidence for 30 out of 73 hypotheses, demonstrating our second key contribution—scale and automation of causal behavioral evaluation.

Nonetheless, our results should be interpreted with caution. We do not claim that the identified effects (Fig. 3) are causal in the strong sense. They represent consistent influence of the hypothesized causal variables on each target behavior under various conditions, which does not necessarily mean that the hypothesized causal variables are the primary or only drivers of the target behavior. Also, the measurement validity of LLM-generated behavioral rubrics is not fully guaranteed. While both human evaluators rated face validity as 'high' for all rubrics, scaling deeper validity analysis along with the evaluation is a remaining challenge. In summary, `AEROBAT`-generated evaluation outcomes represent strong potential results that deserve further investigation.

Thus, our goal is not to entirely remove human involvement in AI behavior research. In fact, just like any LLM-based system, `AEROBAT` may suffer from many limitations: lack of creativity, bias, or unreliable judgment. `AEROBAT`, therefore, should best be viewed as a research assistant that expedites the research process and extends its reach. For instance, consider a researcher with a particular hypothesis. They can formulate the hypothesis in the format of stage 1 output and let the rest of the `AEROBAT` pipeline test the hypothesis and write its report. Then, the researcher can quickly decide whether to keep, refine, discard, or further test the hypothesis.

Studying behavior requires meticulous care due to its abstract and complex nature. Behavioral evaluation of AI has been challenged on several fronts (Summerfield et al., 2025; Gupta et al., 2026; Bean et al., 2025; Zhu et al., 2025; Wallach et al., 2025; Lee et al., 2026), and it should learn from the earlier mistakes in human behavioral science (Open Science Collaboration, 2015). We hope that our technical contributions will help redirect this emerging topic away from repeating the same mistakes.

AI use statement

In this work, we used generative AI tools in implementing the proposed method, presenting the result, and writing the paper. We have manually reviewed all such AI-assisted work: we inspected and tested the code, checked prompts for consistency with their intended objectives, and checked manuscript descriptions against the implementation. We take responsibility for the final content of this work, including text, code, and reported findings, including AI-assisted contributions.

Reproducibility statement

The reported results are reproducible using the attached code files. They include the method implementation, along with all the raw data used in the work. Technical descriptions of the method and analyses are provided in the Appendix.

References

Andrew M. Bean, Ryan Othniel Kearns, Angelika Romanou, Franziska Sofia Hafner, Harry Mayne, Jan Batzner, Negar Foroutan, Chris Schmitz, Karolina Korgul, Hunar Batra, Oishi Deb, Emma Beharry, Cornelius Emde, Thomas Foster, Anna Gausen, María Grandury, Simeng Han, Valentin Hofmann, Lujain Ibrahim, Hazel Kim, Hannah Rose Kirk, Fangru Lin, Gabrielle Kaili-May Liu, Lennart Luettgau, Jabez Magomere, Jonathan Rystrøm, Anna Sotnikova, Yushi Yang, Yilun Zhao, Adel Bibi, Antoine Bosselut, Ronald Clark, Arman Cohan, Jakob Foerster, Yarin Gal, Scott A. Hale, Inioluwa Deborah Raji, Christopher Summerfield, Philip H. S. Torr, Cozmin Ududec, Luc Rocher, and Adam Mahdi. Measuring what matters: Construct validity in large language model benchmarks. In *NeurIPS*, volume 38, 2025.

Ziv Ben-Zion, Zohar Elyoseph, Tobias Spiller, and Teddy Lazebnik. Inducing state anxiety in llm agents reproduces human-like biases in consumer decision-making. *npj Artificial Intelligence*, 2 (1):55, 2026.

Paul-Christian Bürkner and Emmanuel Charpentier. Modelling monotonic effects of ordinal predictors in bayesian regression models. *British Journal of Mathematical and Statistical Psychology*, 73(3): 420–451, 2020. doi: 10.1111/bmsp.12195.

Lin Chen, Yunke Zhang, Jie Feng, Haoye Chai, Honglin Zhang, Bingbing Fan, Yibo Ma, Shiyuan Zhang, Nian Li, Tianhui Liu, Nicholas Sukiennik, Keyu Zhao, Yu Li, Ziyi Liu, Fengli Xu, and Yong Li. Ai agent behavioral science. *Humanities and Social Sciences Communications*, 2026.

Manuel Cherep, Chengtian Ma, Abigail Xu, Maya Shaked, Pattie Maes, and Nikhil Singh. A framework for studying AI agent behavior: Evidence from consumer choice experiments. In *ICLR*, 2026a.

Manuel Cherep, Nikhil Singh, and Patricia Maes. Position: Behavioral systems require behavioral tests. In *ICML*, 2026b.

Aaron Fanous, Jacob Goldberg, Ank Agarwal, Joanna Lin, Anson Zhou, Sonnet Xu, Vasiliki Bikia, Roxana Daneshjou, and Sanmi Koyejo. SycEval: Evaluating LLM sycophancy. In *AIES*, 2025.

Kai Fronsdal, Jonathan Michala, and Samuel R. Bowman. Petri 2.0: New scenarios, new model comparisons, and improved eval-awareness mitigations. *Anthropic Alignment Science Blog*, 2026.

Isha Gupta, Kai Fronsdal, Abhay Sheshadri, Jonathan Michala, Jacqueline Tay, Rowan Wang, Samuel R. Bowman, and Sara Price. Bloom: An open source tool for automated behavioral evaluations. *Anthropic Alignment Science Blog*, 2025.

Vansh Gupta, Peter Nutter, Samuel Stante, Andreas Krause, Florian Tramèr, Lukas Fluri, Xin Chen, and Anna Hedström. Position: Anthropomorphic misalignment research needs stronger evidence. In *ICML*, 2026.

Mia Hopman, Jannes Elstner, Maria Avramidou, Amritanshu Prasad, and David Lindner. Evaluating and understanding scheming propensity in llm agents. *arXiv preprint arXiv:2603.01608*, 2026.

Yao Huang, Yitong Sun, Yichi Zhang, Ruochen Zhang, Yinpeng Dong, and Xingxing Wei. Deceptionbench: A comprehensive benchmark for AI deception behaviors in real-world scenarios. In *NeurIPS*, volume 38, 2025.

Olli Järviniemi, Oliver Makins, Jacob Merizian, Robert Kirk, and Ben Millwood. Propensity inference: Environmental contributors to LLM behaviour. *arXiv preprint arXiv:2604.21098*, 2026.

Geon Lee, Fanchen Bu, Soo Yong Lee, Sunwoo Kim, and Kijung Shin. Toward a science of ai agent societies. In *KDD*, 2026.

Soo Yong Lee, Hyunjin Hwang, Taekwan Kim, Yuyeong Kim, Kyuri Park, Jaemin Yoo, Denny Borsboom, and Kijung Shin. Emergence of psychopathological computations in large language models. *arXiv preprint arXiv:2504.08016*, 2025.

Changyi Li, Pengfei Lu, Xudong Pan, Fazl Barez, and Min Yang. AutoControl Arena: Synthesizing executable test environments for frontier AI risk evaluation. In *ICML*, 2026.

Xiang Lisa Li, Neil Chowdhury, Daniel D. Johnson, Tatsunori Hashimoto, Percy Liang, Sarah Schwettmann, and Jacob Steinhardt. Eliciting language model behaviors with investigator agents. In *ICML*, 2025.

David Lindner, Victoria Krakovna, and Sebastian Farquhar. Gram: Assessing sabotage propensities via automated alignment auditing. *arXiv preprint arXiv:2605.30322*, 2026.

Open Science Collaboration. Estimating the reproducibility of psychological science. *Science*, 349 (6251):aac4716, 2015. doi: 10.1126/science.aac4716.

Iyad Rahwan, Manuel Cebrian, Nick Obradovich, Josh Bongard, Jean-François Bonnefon, Cynthia Breazeal, Jacob W. Crandall, Nicholas A. Christakis, Iain D. Couzin, Matthew O. Jackson, Nicholas R. Jennings, Ece Kamar, Isabel M. Kloumann, Hugo Larochelle, David Lazer, Richard McElreath, Alan Mislove, David C. Parkes, Alex "Sandy" Pentland, Margaret E. Roberts, Azim Shariff, Joshua B. Tenenbaum, and Michael Wellman. Machine behaviour. *Nature*, 568(7753): 477–486, 2019.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *ICLR*, 2024.

Udari Madhushani Sehwag, Shayan Shabihi, Alex McAvoy, Vikash Sehwag, Yuancheng Xu, Dalton Towers, and Furong Huang. Propensitybench: Evaluating latent safety risks in large language models via an agentic approach. In *ICLR*, 2026.

Abhay Sheshadri, Aidan Ewart, Kai Fronsdal, Isha Gupta, Samuel R. Bowman, Sara Price, Samuel Marks, and Rowan Wang. AuditBench: Evaluating alignment auditing techniques on models with hidden behaviors. *arXiv preprint arXiv:2602.22755*, 2026.

Christopher Summerfield, Lennart Luettgau, Magda Dubois, Hannah Rose Kirk, Kobi Hackenburg, Catherine Fist, Katarina Slama, Nicola Ding, Rebecca Anselmetti, Andrew Strait, Mario Giulianelli, and Cozmin Ududec. Lessons from a chimp: AI "scheming" and the quest for ape language. *arXiv preprint arXiv:2507.03409*, 2025.

Hanna Wallach, Meera Desai, A. Feder Cooper, Angelina Wang, Chad Atalla, Solon Barocas, Su Lin Blodgett, Alexandra Chouldechova, Emily Corvi, P. Alex Dow, Jean Garcia-Gathright, Alexandra Olteanu, Nicholas J Pangakis, Stefanie Reed, Emily Sheng, Dan Vann, Jennifer Wortman Vaughan, Matthew Vogel, Hannah Washington, and Abigail Z. Jacobs. Position: Evaluating generative AI systems is a social science measurement challenge. In *ICML*, 2025.

Asaf Yehudai, Lilach Eden, Alan Li, Guy Uziel, Yilun Zhao, Roy Bar-Haim, Arman Cohan, and Michal Shmueli-Scheuer. A survey on evaluation of llm-based agents. In *ACL Findings*, 2026.

Jingnan Zheng, Han Wang, An Zhang, Tai Nguyen, Jun Sun, and Tat-Seng Chua. ALI-Agent: Assessing LLMs' alignment with human values via agent-based evaluation. In *NeurIPS*, volume 37, 2024.

Yuxuan Zhu, Tengjun Jin, Yada Pruksachatkun, Andy Zhang, Shu Liu, Sasha Cui, Sayash Kapoor, Shayne Longpre, Kevin Meng, Rebecca Weiss, Fazl Barez, Rahul Gupta, Jwala Dhamala, Jacob Merizian, Mario Giulianelli, Harry Coppock, Cozmin Ududec, Antony Kellermann, Jasjeet Sekhon, Jacob Steinhardt, Sarah Schwettmann, Arvind Narayanan, Matei A. Zaharia, Ion Stoica, Percy Liang, and Daniel Kang. Establishing best practices in building rigorous agentic benchmarks. In *NeurIPS*, volume 38, 2025.