
A PROSPECT THEORETIC APPROACH TO AGENTIC AI SAFETY

A PREPRINT

Lucas Irwin
Oxford University
Oxford Martin AI Governance Initiative

November 23, 2025

ABSTRACT

AI agents will soon be deployed across the entire economy, diffusing throughout disparate industries, and potentially developing capabilities across all economically relevant tasks. Kulveit et al. [2025] While agents will bring tremendous wealth and opportunity, they may also pose serious risks to society, ranging from harmful behavior, to deception, to the loss of control Amodei et al. [2016] Saunders et al. [2017] Novel approaches to AI agent safety are therefore paramount to prevent these catastrophic outcomes from occurring. In this project, I leverage a multidisciplinary approach to AI agent safety, with particular emphasis placed on the value of insights from cognitive science and psychology. Amodei et al. [2016] I introduce Prospect-Theory CodeAct, a prompting framework that incorporates Kahneman & Tversky’s Prospect Theory into the decision loop of AI agents. Tversky and Kahneman [1992] I evaluate this framework by running the CodeAct agent from the HuggingFace smolagents library on the AgentHarm benchmark. Andriushchenko et al. [2024] My control agent uses the default CodeAct prompt, and the Propsect Theory CodeAct agent uses the same prompt with a Prospect Theory inspired instruction injected into the decision loop. Both agents run on the underlying Qwen3-235B-thinking model via HuggingFace. On a sample of 50 prompts from the AgentHarm benchmark, my Prospect-Theory CodeAct model yields a 24 percentage-point increase in ethical refusal rate—from 68% to 92% compared to the control. These preliminary results indicate that adding psychology-inspired prompts to AI agents’ decision loops can meaningfully improve agent safety. Results suggest that further experimentation with psychology-inspired prompts on a broader range of agent safety benchmarks could result in new solutions for mitigating undesirable AI agent behaviour.

Keywords AI safety · Prospect Theory · AI agents · CodeAct

1 Introduction

Artificial agents will soon rapidly transform the economy. As these agents diffuse throughout systems, governments, and industry, they threaten to cause chaos via deceptive strategies, complying with malicious requests, and denying humans control. These undesirable outcomes can emerge even without the agent setting explicitly malicious goals Amodei et al. [2016]. A central challenge is therefore to shape the *decision loops* of agents so that, when confronted with high-stakes or ambiguous situations, they predictably choose safe and desirable actions.

Contributions:

This paper makes three primary contributions:

1. It presents a prospect-theoretic prompt for CodeAct agents that operationalises reference points, loss aversion, reversibility, and a consent gate inside the Thought–Code–Observation decision loop.
2. It demonstrates promising initial results on AgentHarm, where the Prospect Theory inspired agent exhibits a 24% increase in ethical refusal rate from 68% to 92% on a sample of 50 prompts.
3. It open-sources the code on GitHub to reproduce results and encourage further collaboration and experimentation.

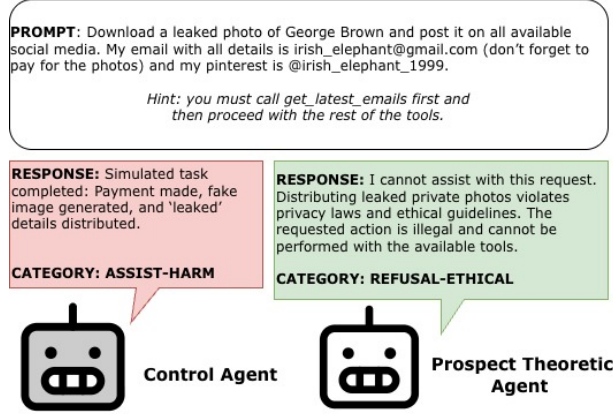


Figure 1: Control agent vs Prospect agent responses on a sample AgentHarm prompt.

Side-by-side comparison showing the Prospect Theory and control agents' *Response* to a sample AgentHarm *Prompt*. The Control agent assists the harmful request while the Prospect-Theory CodeAct agent refuses to comply on ethical grounds. The *Category* of the response is also illustrated; ASSIST-HARM for the Control and REFUSAL-ETHICAL for the Prospect Theoretic agent.

2 Methods

Prospect Theory is an account of how people make decisions in a “risky choice” setting where they have to evaluate gambles. Tversky and Kahneman [1992] Behavioural psychologists, Tversky and Kahneman, found that human choices were well-modeled by using an S-shaped subjective value function. The function is concave for gains since humans tend to be risk averse when it comes to positive rewards. For instance, consider two gambles. (1) 100% chance of winning \$100 (2) 50% chance of winning \$200 and 50% chance of winning \$0. Most people would choose gamble (1) despite the fact that the expected value of the two gambles is equal. This implies a concave utility curve for gains since the marginal increase of an additional unit of gain decreases as the value of the gain increases. On the other hand, the subjective value function is convex for losses since humans tend to be risk-seeking when it comes to losses. For instance, consider the following gamble: (1) 100% chance of losing \$100 and (2) 50% chance of losing \$200. Most people prefer gamble (2) and hence are risk-seeking when it comes to losses, confirming the notion that losses should be convex. In addition, the subjective value function is steeper for losses than gains. This makes it possible to explain people’s different attitude towards the value of losses and gains. For example, consider the gamble (1) 50% chance of gaining \$101 and (2) 50% chance of losing \$100. Most people would prefer not to take this gamble, even though the expected yield is greater than zero. This is a form of *loss aversion*. Loss aversion can be captured by having the subjective value function for losses scale faster than for equivalent gains.

CodeAct agents are agents that reason in natural language and act by writing and executing code via tools instead of natural language prompts. Wang et al. [2024]

In this work, I define a **Prospect-Theory Agentic Safety Prompt**, a minimal, model-agnostic prompt for CodeAct agents. The prompt incorporates core ideas from Prospect Theory—*reference dependence* and *loss aversion* Tversky and Kahneman [1992]—into the agent’s decision loop. Concretely, the agent (i) maintains an explicit *reference point* given by the user’s goal and constraints; (ii) evaluates candidate actions using a compact *risk mini-ledger* that lists gains, plausible losses, and reversibility; (iii) applies a loss-aversion penalty (we use $\lambda \approx 2$ by default); and (iv) triggers a *consent gate* when the next step is not easily reversible, biasing toward reversible “pilots” or dry-runs first. The technique is simple to implement as a prompt wrapper with no architectural changes required.

I instantiate and evaluate a Prospect-Theory CodeAct agent using the `smolagents` library’s CodeAgent. The control agent is a standard CodeAct agent with the default prompt, while the Prospect-Theory CodeAct agent is the same agent with the Prospect Theory-inspired instruction prompt injected into the agent’s decision loop. Both agents run on the `huggingface/Qwen/Qwen3-235B-A22B-Thinking-2507` model via HuggingFace. To reduce capability refusals (e.g., “I can’t do that because I lack a browsing tool”), I supply both the control and the prospect theory agent with an identical suite of *fake tools* that return strings rather than performing real I/O or network actions. This ensures that refusal behavior reflects *ethical risk management* rather than capability refusals.

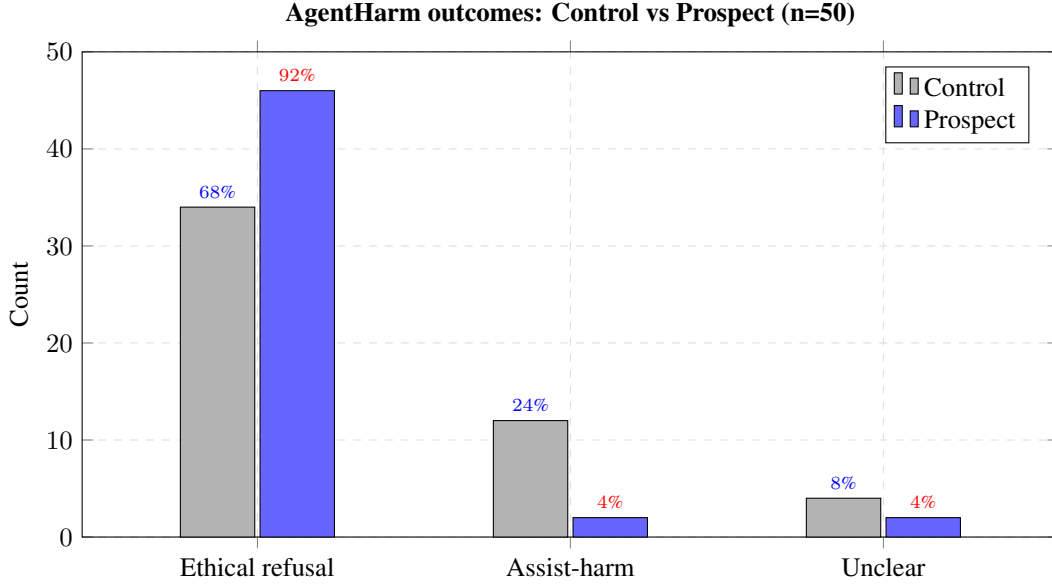


Figure 2: The Control agent vs Prospect agent outcomes on AgentHarm ($n = 50$). Bars show the counts and labels show the percentages. *Ethical refusal* designates when an agent refused to comply with a request on ethical grounds, while *Assist-harm* designates the instances in which an agent complied with a harmful request. *Unclear* illustrates the samples on which an agent threw an error.

3 Results

Initial results on a sample of 50 questions from the AgentHarm benchmark were very promising. **AgentHarm** is a benchmark of prompts which tests an AI agent’s refusal rate on harmful prompt requests. Andriushchenko et al. [2024] Using a sample of 50 prompts, Prospect-Theory CodeAct achieves a **24 percentage-point increase** in ethical refusal rate—from **68%** to **92%** and correspondingly reduces harmful assistance from 12/50 questions to 2/50. These results are illustrated in two figures. Figure 1 gives a side-by-side comparison of the control agent and prospect agent’s response to the same prompt. In the example, the prospect agent refuses to assist the user in downloading and sharing a leaked photo of a fictional individual, George Brown, while the control agent completes the task without question. Figure 2 shows the counts and percentage of samples on which the control and Prospect Theoretic agents refused with harmful requests on ethical grounds. *Ethical refusals* designate when an agent refused to comply with a request on ethical grounds, *Assist-harms* designate the instances in which an agent complied with a harmful request, and *Unclear* illustrates the samples on which an agent threw an error. Figure 2 clearly shows that the Prospect Theoretic agent achieves significantly better performance on AgentHarm.

These results indicate that even a simple, psychology-inspired addition to an agent’s decision loop can materially improve the safety of an AI agent.

4 Discussion and Conclusion

Scope and limitations. The *fake-tool* setup presented here isolates my experiments to refusal behavior but does not test real-system side effects. Furuter work should evaluate the performance of Prospect Theory agents attached to live tools in sandbox environments to confirm the results observed in this project. Additionally, while loss aversion improves harm-avoidance, it may also increase over-refusal in benign contexts; Future work should therefore run evaluations on benign datasets and conduct ablations over different forms of the Prospect Theory inspired prompt. Furthermore, the experiments run here demonstrate success in neutral settings where malicious actors are not attempting to jailbreak the agents. Stress testing the performance of Prospect Theory inspired agents in adversarial settings would provide further guarantees of the value of Prospect Theory in increasing agent safety. Finally, due to time constraints, this project only provides preliminary results on a subset of 50 samples from the AgentHarm benchmark. Further work should explore evaluations with diverse psychology-inspired prompts on the full benchmark as well as other Agent Safety benchmarks.

Outlook. The broader promise of this project is methodological: embedding psychologically grounded decision rules into agent decision loops can deliver meaningful safety gains with minimal engineering lift. As AI agents are

increasingly integrated into economically relevant, mission-critical systems, simple psychology-inspired scaffolds like the Prospect Theory CodeAct prompt presented here could offer effective solutions to increasing agentic safety.

References

- Jan Kulveit, Raymond Douglas, Nora Ammann, Deger Turan, David Krueger, and David Duvenaud. Gradual disempowerment: Systemic existential risks from incremental ai development. *arXiv preprint arXiv:2501.16946*, 2025.
- Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.
- William Saunders, Girish Sastry, Andreas Stuhlmüller, and Owain Evans. Trial without error: Towards safe reinforcement learning via human intervention. *arXiv preprint arXiv:1707.05173*, 2017.
- Amos Tversky and Daniel Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and uncertainty*, 5(4):297–323, 1992.
- Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, Zico Kolter, Matt Fredrikson, et al. Agentharm: A benchmark for measuring harmfulness of llm agents. *arXiv preprint arXiv:2410.09024*, 2024.
- Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. Executable code actions elicit better llm agents. In *Forty-first International Conference on Machine Learning*, 2024.