EXPLORE IT TILL YOU SOLVE IT: EXPLORATION-ONLY SOLUTION FOR ARC-AGI-3

Evgenii Rudakov
Faculty of Educational Sciences
University of Helsinki
Helsinki, Finland
evgenii.rudakov@helsinki.fi

August 11, 2025

ABSTRACT

We present a graph-based exploration-only method for solving interactive reasoning tasks in the ARC-AGI-3 benchmark. Our approach combines intelligent frame processing with systematic state-space exploration to achieve better-than-random and better-than-LLM performance on game environments. The method segments visual frames into meaningful components, prioritizes actions based on visual salience, and maintains a graph representation of explored states. While limited in scope, this approach demonstrates the effectiveness of structured exploration over purely random strategies in novel game environments that test artificial general intelligence capabilities.

1 Introduction

The pursuit of artificial general intelligence (AGI) has led to the development of increasingly sophisticated benchmarks designed to measure human-like reasoning capabilities. The ARC Prize Foundation, founded by Mike Knoop and François Chollet, has established a $1M+ global competition to guide researchers towards AGI through enduring benchmarks. The evolution from static pattern recognition tasks to interactive environments represents a significant paradigm shift in how we evaluate machine intelligence.

ARC-AGI-3 pioneers the Interactive Reasoning Benchmark (IRB) paradigm, leveraging game environments to provide a rich medium for testing experience-driven competence. Unlike previous versions that tested pattern recognition through visual grids, this new approach measures human-like intelligence through skill-acquisition efficiency in novel gaming environments. The benchmark features mini-games set in a grid world where AI agents must figure out the rules and objectives for themselves, learning how to succeed through trial and error.

The fundamental principle underlying ARC-AGI's design is the need to create a fair and meaningful comparison between artificial and human intelligence, focusing on fluid intelligence. As François Chollet articulated in his influential 2019 paper "On the Measure of Intelligence," intelligence should be measured by the efficiency of skill-acquisition on unknown tasks—essentially, how quickly one can learn new skills.

The current state of the field reveals a stark performance gap: while humans achieve 100% success rates on ARC-AGI-3 tasks, frontier AI systems score 0% out of the box. This dramatic difference underscores the challenge of developing truly general intelligence systems that can adapt to novel environments without extensive training.

In this context, we present a pragmatic approach that, while not achieving human-level performance, demonstrates systematic improvement over random exploration strategies and vanilla LLM-agents. Our method helps to set the limits on what can be achieved with exploration only, without learning and intelligent decision-making.

2 Methods

Our approach consists of two primary components: a Frame Processor for intelligent visual analysis and a Level Graph Explorer for systematic state-space exploration. The method is designed to be more intelligent than a purely random agent while remaining computationally tractable for the game environments in ARC-AGI-3.

2.1 Frame Processor

The Frame Processor aims to reduce irrelevant visual variability and focus exploration efforts on actionable regions of the game environment. This component performs several key operations:

Image Segmentation. The processor begins by segmenting each frame into single-color connected components. This segmentation serves as the foundation for identifying distinct visual elements that may represent interactive objects.

Status Bar Detection and Masking. To avoid confusing environmental elements with user interface components, the processor detects and masks likely status bars, specifically step counters. This preprocessing step helps reduce the number of states to explore by the number of steps allowed.

Priority-Based Action Grouping. For click-controlled games, the processor groups visual segments into five priority tiers based on their likelihood of representing interactive buttons or objects. This prioritization is determined by segment size, shape and color salience. The lowest priority tier specifically includes segments identified as likely status bars, ensuring they are explored only when all other options have been exhausted.

State Hashing. Finally, the processor generates a hash of the masked image, which serves as a unique identifier for the current game state. This hash enables efficient state tracking and duplicate detection in the graph exploration phase.

2.2 Level Graph Explorer

The Level Graph Explorer maintains a graph representation of the explored state space, where nodes represent unique game states and edges represent actions that transition between states.

Graph Structure. For each known frame (graph node), the explorer maintains:

- A list of possible actions (clicks for games like `ft09/cv33`, keyboard keys for games like `ls20`)
- For each action: its priority level, testing status, transition result, destination frame, and distance to the nearest unexplored frontier

Action Selection Strategy. The explorer employs a hierarchical action selection strategy:

Algorithm 1 Hierarchical Action Selection

Require: Current game state s, current priority level p
  if untested actions at priority $\leq p$ exist in state s **then**
    Randomly select and execute an untested action with priority $\leq p$ from s
    Update graph with transition result
  else if can reach state with untested actions at priority $\leq p$ from s **then**
    Randomly select and execute an action with the lowest distance to reachable state with untested actions at priority $\leq p$
  else
    Increment priority level: $p \leftarrow p + 1$
    Repeat from current state s with new priority p
  end if

This strategy ensures that the most promising actions (those with higher visual salience) are explored before resorting to lower-priority alternatives.

Frontier Management. The explorer maintains paths to frontier frames, states with untested actions, and tracks the distance from each explored state to the nearest frontier. This information guides the search process toward unexplored regions of the state space.

3 Discussion

3.1 Strengths and Effectiveness

Our graph-based exploration method represents a structured approach to the challenge of interactive reasoning in novel game environments. The method's primary strength lies in its ability to be systematically more efficient than random exploration while remaining computationally tractable.

The hierarchical action selection strategy based on visual salience provides a reasonable heuristic for prioritizing exploration in environments where visual cues correlate with interactive elements. By focusing first on visually prominent components that are likely to be buttons or interactive objects, the method can more efficiently discover the core mechanics of each game.

The graph representation of explored states prevents redundant exploration and enables systematic coverage of the reachable state space. This approach is particularly valuable in deterministic environments where the same action from the same state always produces the same result.

3.2 Limitations and Failure Modes

Despite its structured approach, the method has several significant limitations that can impact its effectiveness:

Status Bar Sensitivity. The method's reliance on status bar detection makes it vulnerable to games where the user interface significantly differs from public examples. If status bars are integrated into the scene rather than positioned at the edges, the state space implicitly includes many status bar variants, effectively degrading the method toward random exploration.

Scalability Constraints. Large state spaces, such as those encountered in levels 3-4 of `ft09`, can make the brute-force exploration approach intractable. The method's exhaustive nature means that computational requirements grow exponentially with the size of the reachable state space.

Non-Determinism and Partial Observability. The method assumes deterministic environments with full observability. Non-deterministic transitions or partially observable states can cause significant issues, as the graph representation becomes less reliable and the systematic exploration strategy may fail to capture the true nature of state transitions.

3.3 Future Directions

A natural extension would be to incorporate simple world models that predict the next frame given the current frame and action. Such models could improve sample efficiency by approximately the average number of actions per frame, reducing the computational burden of exhaustive exploration.

However, a fundamental challenge remains in determining how to prioritize exploration of "interesting" states. While uncertainty-based or surprise-driven exploration strategies are commonly employed in reinforcement learning, their applicability to these reasoning games is questionable. It is not immediately clear why states with higher model uncertainty or prediction error should correlate with progress toward solving the game. For instance, in the ft09 environment, there is no principled reason to expect that the correct pattern should be more surprising to a learned model than an incorrect one.

4 Results Template

4.1 Experimental Setup

We evaluated our graph-based exploration method on three games from the ARC-AGI-3 benchmark: ft09 (10 levels), ls20 (8 levels), and vc33 (9 levels). Each level was attempted multiple times to compute average performance metrics. The method was run until either the level was solved or a maximum action limit was reached, at which point the level was considered unsolved.

4.2 Performance Results

Table 1 presents the average number of actions required to solve each level across the three games. Unsolved levels are marked as "Not solved" in the table.

Table 1: Average number of actions to solve each level by game

Level	ft09	ls20	vc33
1	$<$500	$<$200	$<$10
2	$<$200	$<$5,000	$<$10
3	$<$10,000	Not solved	$<$100
4	$<$100,000	Not solved	$<$500
5	$<$5,000	Not solved	$<$1,000
6	Not solved	Not solved	$<$50,000
7	Not solved	Not solved	$<$5,000
8	Not solved	Not solved	$<$500
9	Not solved	-	Not solved
10	Not solved	-	-

4.3 Analysis

The results demonstrate varying levels of success across the three games. The vc33 game showed the most consistent solvability, with 8 out of 9 levels solved, though with significant variation in the number of actions required (ranging from $<$10 to $<$50,000 actions). The ft09 game proved more challenging, with only the first 5 levels solved, including one level requiring up to 100,000 actions. The ls20 game was the most difficult, with only the first 2 levels solved.

The exponential growth in action requirements for certain levels (e.g., ft09 level 4 requiring $<$100,000 actions) highlights the scalability limitations of the brute-force exploration approach. Interestingly, some higher-numbered levels required fewer actions than lower-numbered ones (e.g., vc33 levels 7-8), suggesting that level numbering does not always correlate with difficulty for this exploration method.

The method's performance varied significantly between games, indicating that game-specific characteristics such as state space size, visual complexity, and interaction patterns strongly influence the effectiveness of the exploration strategy.

5 Conclusion

We have presented a graph-based exploration method for ARC-AGI-3 interactive reasoning benchmarks that demonstrates systematic improvement over random exploration strategies. While the approach has clear limitations in terms of scalability and robustness to interface variations, it provides a foundation for more sophisticated reasoning systems.

The method's ability to outperform purely random agents represents a step toward systems capable of structured reasoning in novel interactive environments. Future work should focus on incorporating learned world models and uncertainty-based exploration to achieve the sample efficiency necessary for more complex reasoning tasks.

The stark performance gap between human and AI performance on ARC-AGI-3 (100% vs. 0%) underscores the fundamental challenges that remain in developing truly general intelligence systems. Our approach, while limited, contributes to the broader research effort aimed at closing this gap through systematic and structured exploration strategies.