

# ARC-AGI-3 벤치마크용 에이전트에 대한 연구: 월드 모델

## 위에서의 탐색에 대해서

박성수<sup>0</sup>, 로라 마스카렐, 김선동

GIST

heroictimpani@gm.gist.ac.kr, {lmascarell, sundong}@gist.ac.kr

## Research on Agents for the ARC-AGI-3 Benchmark: On the

## Search on the World Model

Seongsoo Park<sup>0</sup>, Laura Mascarell, Sundong Kim

GIST

heroictimpani@gm.gist.ac.kr, {lmascarell, sundong}@gist.ac.kr

### 요약

인공지능 분야에서, 처음 보는 환경에 적응하는 능력은 AI 에이전트 및 AGI에 대한 중대한 문제이다. ARC-AGI-3 벤치마크는 이렇게 낯선 환경에서 효율적으로 행동하는 능력을 측정하기 위해 개발되었다. ARC-AGI-3에서 해결해야 하는 문제는 환경에 대한 정보가 알려지지 않은 채로 에이전트가 스스로 효율적으로 행동해서 환경의 규칙과 목표를 파악하고 계획을 실행에 옮겨야 하는 어려운 문제이다. 본 연구에서는 미지 환경에 대해서 언어 모델이 관측을 통해 월드 모델을 만들고 이 월드 모델 위에서 state search planner가 실행되는 에이전트 아키텍처를 기반으로, planner의 탐색 효율성을 향상시키기 위한 subgoal 기반 planner를 제안한다. 이번 연구에서는 subgoal 기반 planner의 효율성을 oracle 방식을 통한 시험적 실험을 통해 확인하였다.

### 1. 서론<sup>1</sup>

인공지능 모델이 특정 작업에서 우수한 성능을 내는 것과 대비되어, 새로운 환경에 적응하는 능력은 인공지능 분야에서 앞으로 달성해야 할 중대한 과제이다. ARC-AGI-3 벤치마크 [1]는 ARC Prize Foundation의 새로운 벤치마크로, 상호작용 가능한 환경을 통해서 에이전트 지능(Agentic Intelligence)을 측정하는 것을 목표로 한다. ARC Prize Foundation이 벤치마크를 개발하는 목적은 Chollet가 “On the Measure of Intelligence” [2]에서 제시한 지능 측정에 대한 아이디어를 구현하기 위한 것이다. 글에서 Chollet는 인공 일반 지능(AGI)에는 특정 분야의 스킬을 갖추는 것이 아니라 새로운 스킬을 효율적으로

습득하는 것이 필요하다고 제시하였으며, 이를 구현하기 위한 벤치마크로 새로운 규칙을 추론하는 것을 목표로 하는 ARC-AGI의 최초 버전을 제시하였다.

기존 ARC 버전들과 비교해서 ARC-AGI-3 벤치마크의 특징은 동적 상호작용을 요구한다는 것이다. ARC-AGI-3 벤치마크는 게임 형태의 여러 환경으로 구성되어 있으며, 각 환경은 여러 개의 레벨로 구성되어 있다. ARC-AGI-3 벤치마크에서 측정하고자 하는 에이전트의 역량은 새로운 환경에서 효율적으로 행동하는 능력이다. 에이전트는 규칙이나 목표 등의 정보가 없는 낯선 환경에 대해서 스스로 탐색(Exploration)을 통해서 환경의 규칙을 알아내고, 그렇게 알아낸 규칙과 목표를 통해서 계획을 수립하고 실행하는 등 환경에 대한 활용(Exploitation)을 수행해야 한다. 따라서 새로운 환경에서의 탐색과 활용 과정을 효율적으로 수행하는 학습 효율은 ARC-AGI-3 벤치마크의 핵심으로 작용한다.

본 연구에서 다루는 대상은 다음과 같다. 첫째, 미지

\* 본 연구는 IITP 디지털혁신기술 국제공동연구(RS-2024-00445087), 과학기술원 InnoCORE 사업(26-InnoCORE-01)에 의해 수행되었습니다.

환경의 규칙을 언어 모델을 통해 코드 형태의 월드 모델로 추상화하여 지식 전이를 구현하는 아키텍처를 제안한다. 둘째, 월드 모델 위에서의 탐색 과정의 병목을 해결하기 위해 월드 모델에서 subgoal을 추출하여 탐색 공간을 압축하는 planner 구조를 제안한다. 셋째, 이러한 planner의 구조를 시험하기 위해 oracle 방식으로 subgoal을 직접 설정한 실험을 통해 subgoal의 유용성을 입증한다.

## 2. 관련 연구

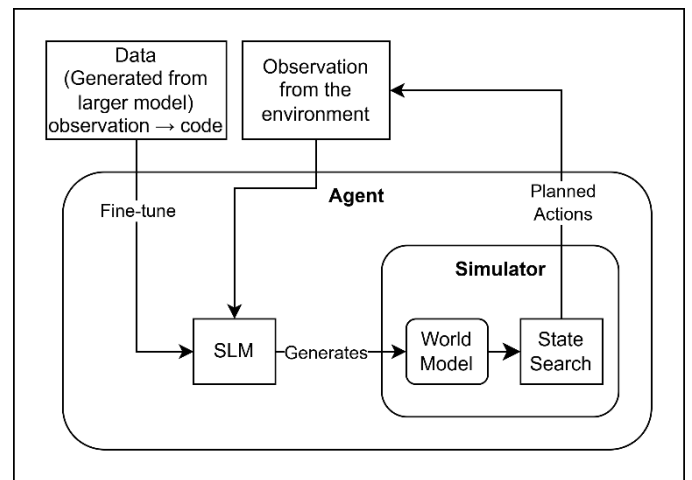
본 연구에서 제시하는 ARC-AGI-3 벤치마크용 에이전트의 구조는 TheoryCoder 프레임워크 [3]의 구조를 참고하여 제안되었다. TheoryCoder는 전체 구조에서 계층적 계획이 들어가고, 그 중 low-level에서 LLM, 월드 모델, BFS의 순환이 돌아가며, 게임의 목표가 high-level에서 미리 정의되어 있다. 반면 본 연구에서는 언어 모델, 월드 모델과 state search의 피드백 루프가 에이전트의 메인이 되어서 미지 환경에 대해 직접적인 추론을 하게 하고, 계층적 계획을 state search에 통합하여 병목 현상을 완화하고자 하였다. 또한 본 연구는 2026년 4월 기준 ARC Prize 2026 리더보드에서 상위를 차지한 FORGE [4]에서도 영감을 받았으며, 본 연구는 environment file을 기반으로 BFS를 수행하는 FORGE의 단순한 구조와 달리, 계층적 planner를 도입하여 단순 레벨별 BFS와 비교하였다.

## 3. 전반적인 에이전트의 구조

ARC-AGI-3 대회인 ARC Prize 2026 [5]에서는 에이전트가 인터넷 연결 없이 제한 시간 내로 문제를 해결해야 한다. 또한 ARC-AGI-3 벤치마크에서는 환경 규칙이 알려지지 않으므로, 강화 학습과 같은 일반적인 인공지능 학습 방식으로는 문제를 해결하기에 난점이 있다. 다양한 환경에서의 일반화 능력을 시험하는 ARC-AGI-3을 위한 에이전트의 구조에서는 지식 전이(Knowledge Transfer)가 핵심 요소이다. 본 연구에서는 지식 전이의 구현을 위해서 관측 데이터를 명시적인 코드 형태의 월드 모델로 변환하는 에이전트 프레임워크를 제안한다. 이를 통해서 환경의 규칙에 대한 symbolic grounding을 구현하며, 미지 환경의 규칙에 대해 관측과 추론을 통해서 접근할 수 있다.

본 연구에서는 ARC Prize 2026 제출을 염두에 두고 그림 1과 같은 구조의 에이전트를 제안한다. 에이전트는 소규모 언어 모델(SLM)과 이 모델에 의해 동적으로 생성되는 시뮬레이터로 구성된다. SLM은 직접적으로 action을 계획하고 실행하는 대신, ARC 환경에서 얻은 관측으로부터 시뮬레이터 환경을 정의하는 코드 형태의

월드 모델을 생성한다. 여기서 생성되는 월드 모델의 형태는 ARC-AGI Toolkit [6]을 이용한 Python 코드로서, 이것은 ARC-AGI Toolkit을 통해서 ARC-AGI-3 벤치마크를 시험할 때 생성되는 environment file과 같은 형태이다. 또한, 이렇게 생성된 환경 위에서 state search와 action sequence 계획이 수행된다. 한편 SLM은 프런티어 LLM을 통해 얻은 데이터를 사용하여 미세 조정을 통해 훈련되는데, 이것은 SLM이 추론 자체를 학습하여 지식 전이 기능을 학습하도록 하는 방법이다. 이러한 구조는 SLM이 직접 새로운 규칙 자체를 추론하게 함으로써 환경에 유연하게 적응하게 하며, 이는 ARC의 목표인 새로운 스킬을 효율적으로 습득하는 것을 가능하게 한다.



[그림 1] 전반적인 에이전트의 구조 제안.

## 4. 월드 모델 위에서의 탐색

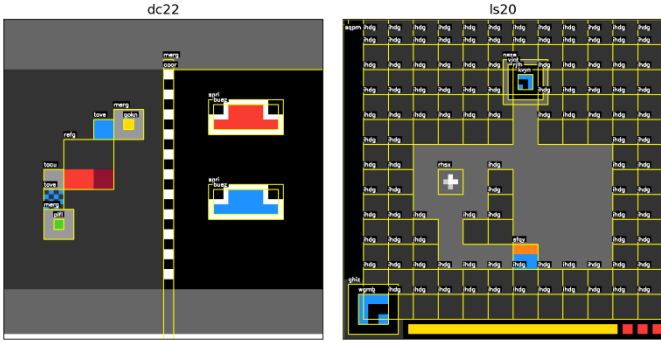
그림 1의 에이전트 구조에서 시뮬레이터는 코드 형태의 월드 모델과 action sequence를 생성하는 state search planner로 이루어져 있다. 여기서 state search에 BFS를 사용하는 것은 간단하면서 다양한 환경에 범용적이라는 장점이 있으나, 단점은 search depth가 증가함에 따라서 실행 시간이 지수적으로 팽창한다는 것이다. 이에 따라 BFS를 이용한 탐색은 에이전트 아키텍처의 병목 지점으로 작용한다.

본 연구에서 제시하는 대안은 state search에서 월드 모델이 가지고 있는 정보를 최대한 활용하는 것으로서, 제안하는 탐색 방식은 다음과 같다. 코드 형태의 월드 모델에서는 객체 간의 다양한 상호작용들의 정보가 들어가 있는데, 이 상호작용의 목록에서 subgoal들을 추출한다. Subgoal은 특정 객체들의 상호작용이나 객체를 특정 위치로 옮기는 것 등의 가시적인 요소들로 적용된다. 그렇게 추출한 subgoal들을 planner에서 엮어서 순서를 가진 하나의 subgoal sequence를 만들고, 각각의 subgoal마다 BFS 탐색을 한다. 이것은

subgoal을 추출하고 그 subgoal들을 순회하는 형태의 계층적 planner이다. 시간 복잡도를 비교하면, 각각의 subgoal에 대한 탐색 깊이를  $d_k$ 라고 하면 전체 레벨에 대한 BFS가  $O(b^d) = O(\prod_k b^{d_k})$ 인 것에 비해 subgoal 분할 시  $O(\sum_k b^{d_k})$ 로 실행 시간을 크게 줄일 수 있다.

이 연구의 실험에서는 이러한 subgoal 방식의 state search가 효율적인지 시험하기 위해서, oracle 방식으로 subgoal을 수동으로 설정하여 실행 시간을 비교하였다. ARC-AGI Toolkit이 생성하는 각 환경의 environment file로부터 월드 모델의 객체에 해당하는 sprite를 추출한 뒤, 이렇게 추출한 sprite의 정보를 이용하여 각 환경의 초반 몇 개의 레벨에 대해서 subgoal sequence를 수동으로 설정하였다.

다음 그림 2는 실험 대상인 dc22와 ls20 환경에 대해서 sprite를 추출한 것을 시각화한 것을 보여준다.



[그림 2] dc22와 ls20 환경에 대한 sprite 추출 시각화.

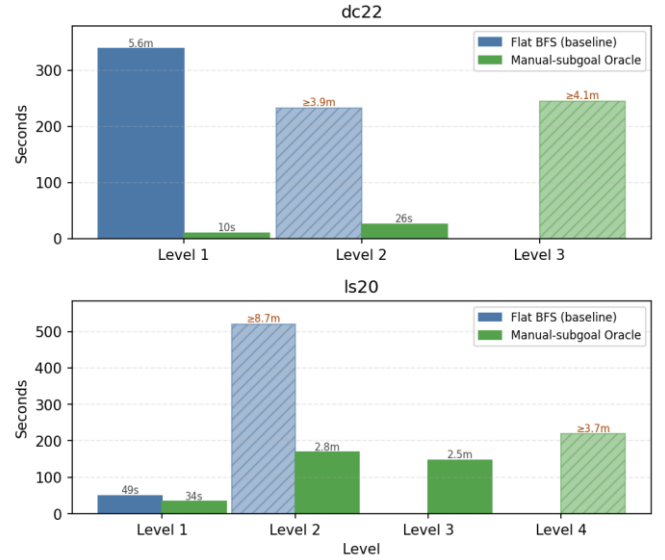
실험은 단순 레벨별 BFS 방식과 subgoal을 설정한 방식의 레벨 해결 시간을 비교하였다. 두 방식에서 월드 모델로는 모두 ARC-AGI Toolkit에서 제공하는 각 환경의 environment file이 사용되었으며, 이것은 월드 모델의 ground truth로 작용한다. 환경 중 dc22는 레벨 2까지, ls20는 레벨 3까지에 대해서 수동으로 subgoal sequence를 설정하였다.

실험 결과는 두 방식의 각 레벨 실행 시간 차이를 그래프(그림 3)로 나타내었다. 실험 결과, dc22의 레벨 1에서는 기존 Flat BFS가 5분 이상 소요했던 탐색 시간이 subgoal 방식에서 10초로 대폭 단축되었고, ls20에서는 실행 시간 초과로 해결하지 못했던 레벨 2를 3분 이내로 해결하는 등, subgoal을 설정한 planner가 더욱 효율적임을 확인할 수 있다.

현재의 실험은 제시하는 planner 구조의 잠재적 효용성을 확인하기 위해, oracle 방식으로 subgoal을 수동으로 설정함을 통해서 subgoal의 효용성에 대한 사전 실험을 진행하였다. 향후 연구의 과제는 코드 형태의 월드 모델에서 자동으로 subgoal을 추출하는 것과, 그렇게 추출한 subgoal들을 하나의 subgoal

sequence로 연결하는 planner를 개발하는 것이다. 그리고 이렇게 개선된 planner와 함께, SLM, 월드 모델, state search planner로 이루어진 피드백 루프 형태의 전반적인 에이전트의 구조를 구현하는 것이 향후 연구의 최종적인 목적이다.

Time per level — solid = cleared, hatched = uncleared (lower bound) · budget 10 min/run



[그림 3] 단순 BFS 방식과 manual subgoal 형태의 oracle 방식에 대한 두 가지 환경에서의 실행 시간 비교. 각 환경과 에이전트의 조합에 대한 실험은 10분 동안 실행되었으며, 막대에 빗금이 있는 것은 시간 내로 해당 레벨을 해결하지 못했음을 의미한다.

### 참고문헌

- [1] ARC Prize Foundation, "ARC-AGI-3: A New Challenge for Frontier Agentic Intelligence," arXiv preprint arXiv:2603.24621, 2026.
- [2] F. Chollet, "On the Measure of Intelligence," arXiv preprint arXiv:1911.01547, 2019.
- [3] Z. Ahmed, J. B. Tenenbaum, C. J. Bates, and S. J. Gershman, "Synthesizing world models for bilevel planning," arXiv preprint arXiv:2503.20124, 2025.
- [4] CHRONOS, "FORGE ARC-AGI-3 Agent," Kaggle, 2026. <https://www.kaggle.com/code/projectforty2/forge-arc-agi-3-agent>
- [5] Abstraction and Reasoning Corpus, "ARC Prize 2026 - ARC-AGI-3," Kaggle, 2026. <https://www.kaggle.com/c/arc-prize-2026-arc-agi-3>
- [6] ARC Prize Foundation, "ARC-AGI," GitHub, 2026. <https://github.com/arcprize/ARC-AGI>