Boldly Propose, Carefully Verify: LLMs in Agents Should Be Used Only For Abduction

Eneko Sabaté
Universitat Politècnica de Catalunya
Barcelona, Spain
eneko.sabate@estudiantat.upc.edu

Victor Gimenez-Abalos
Barcelona Supercomputing Center
Barcelona, Spain
victor.gimenez@bsc.es

Adrian Tormos*
Barcelona Supercomputing Center
Barcelona, Spain
adrian.tormos@bsc.es

Oriol Miro-Lopez-Feliu
Barcelona Supercomputing Center
Barcelona, Spain
oriol.miro@bsc.es

Sergio Alvarez-Napagao
Universitat Politècnica de Catalunya
Barcelona Supercomputing Center
Barcelona, Spain
sergio.alvarez@bsc.es

ABSTRACT

The appearance of Large Language Models (LLMs) has transformed the field of agents with new architectures and inference mechanisms. These systems, however, struggle with capabilities such as rationality or coherent behaviour. Stand-alone LLM-based architectures are not capable of reliably performing planning to produce rational behaviour. On the other hand, traditional agents have long-standing knowledge to tackle these capabilities, but struggle with tasks requiring autonomous abstractions and knowledge abduction, making some domains unattainable: these agents do not have full reasoning capabilities due to failing at abduction. While abduction is computationally difficult, Large Language Models (LLMs) excel at producing probable hypotheses without many requirements. Moreover, verifying these hypotheses via deductive knowledge and interaction with the surroundings is within the capacity of traditional agents, and the feedback can establish a virtuous loop with abduction mechanisms. In this paper, we propose to work toward hybrid systems integrating two stand-alone LLM-based abductive modules: an Abductive Reasoner Module to generate hypotheses based on detected discrepancies between agent beliefs and environment behaviour, and an Experiment Designer Module to generate goals testing these hypotheses. These modules could be integrated with cognitive architectures using means-ends reasoning to fulfil goals (tests and system-goals). We illustrate this perspective through a practical case study in *Baba Is You*, a complex rule-learning environment, empirically showing that abductive-deductive separation is viable even with minimal agent design.

KEYWORDS

Abduction, Inference to the best explanation, LLM-based agents, Scientific discovery, Learning, Agentic AI, Knowledge representation, Knowledge Transfer, Cognitive Architecture

1 INTRODUCTION

Agentic Artificial Intelligence (AI) has recently appeared, producing endless literature about agents with Large Language Models (LLMs) as components [20, 45, 47]. These agentic architectures purportedly show more versatile and interactive capabilities for autonomy than agents not using LLMs, solving complex tasks without human intervention [33, 34].

Planning (or means-ends reasoning) is a key intelligent capability for problem-solving. Although a long-standing focus of the agents community, it has recently become one of the current topics of interest in the agentic community. LLMs are stochastic pattern learning models that use massive statistics of language usage to output text that, while resembling abductive reasoning, cannot be said to reliably follow any rules of inference under any logic [8, 28]. When the action space of a task is known, planning is a purely deductive task, for which LLMs lack formal guarantees of correctness, and for which they have been shown to be unreliable and low-performing [24, 28], sometimes even confabulating and making up facts that support their "reasoning" [21, 36].

Real-world planning also requires learning the action space, often extrapolating from sparse experience, or tangentially similar cases that are hard to generalise. This was already remarked upon in the 80s with famous critiques to purely deductive systems [22]. While abductive reasoning (*i.e.* generating plausible hypotheses from incomplete information) was discussed as a promising solution, abduction theories were piecemeal and unmechanisable [9, 22]. Today, with models that work precisely under the notion of finding the most probable output, deducto-nomological theories [17] appear more reachable: use abduction to propose modifications to *means*, and take proposal validation as *ends* to test the correctness of abduction via *means-ends* reasoning. The merging of deductive, classical systems with abductive, agentic systems could produce *rational*, efficient agents that are more flexible and capable of navigating more complex environments. In environments where testing entails no risk, leveraging LLMs for this purpose is enticing.

In this paper, we do an initial investigation on the potential of LLM-based abduction and testing in an environment where the main difficulty is precisely hypothesising the dynamics of actions: *Baba Is You*. Our approach is inspired by the cycle of Conjecture and Refutation [30]. Upon experiencing unexpected behaviour, the agent generates *bold hypotheses* regarding the causes, tests them through active experimentation (using BFS search), and loops in this behaviour until the expectations match reality. The results hint at the feasibility of coupling this with classical means-ends

*Corresponding author

Proc.@ of the Adaptive and Learning Agents Workshop (ALA 2026), Aydeniz, Delgrange, Mohammedalamen, Yang (eds.), May 25 – 26, 2026, Paphos, Cyprus, https://alaworkshop2026.github.io/. 2026.

reasoning to solve complex problems with unknown action spaces with formal guarantees.

2 BACKGROUND

Autonomy is a central concept to the study of intelligent agency, where agents are defined by their capacity to act independently [32]. While a precise definition of intelligence is still debated, one argument is that intelligence is related to the ability to achieve goals in a wide range of environments [26, 31]. Another emerging perspective within the community emphasises that intelligence should be measured by efficiency in skill acquisition, focusing on abstraction, reasoning and generalisation from limited data [5]. This last posture has been put into practice through the Abstraction and Reasoning Corpus (ARC) for Artificial General Intelligence (AGI) benchmark [10], composed of challenges designed to resist memorisation such that solutions require generalisation capabilities.

Abduction is a foundation of human learning and discovery, and an apparent requirement for the previous definition. However, its implementation in AI has historically been constrained by the immense difficulty of modelling the real world's complexity. Abductive Logic Programming (ALP) and neuro-symbolic architectures have been used to develop abductive systems to some success on small, restricted tasks [18, 19, 43].

Prescinding from abduction, Hayes's *Naive Physics Manifesto* [16], attempted to codify a comprehensive world model in purely deductive terms. This line showed initial success, but later showed that the explicitness of deductive models precludes scalability and is unfeasible for real-world domains [22]. More so, the abductive solutions to these same problems remains unoperationalisable, and formalising abduction as pure logic has been shown an oversight [9, 22].

However, recent inference models show promise in acquiring abduction capabilities without a rigorous framework. LLMs show *apparent* reasoning, generating convincing (if not entirely reliable) plans, decisions, or complex code. This utility is expanded through embodiment [6, 40], converting LLMs into agents that interact with environments that provide feedback. To bridge the gap between LLM outputs and agency, frameworks such as ReAct [44], Voyager [41], Generative Agents [29], and CRADLE [37] have emerged, enabling LLMs to function as the cognitive cores of agents capable of executing tasks autonomously.

Some of these works (*e.g. Voyager*, *CRADLE*) have tackled autonomous curriculum planning and skill acquisition, although relying on complete, LLM-provided domain knowledge, or even access to *Wikipedia* pages and the internet. This reliance limits their robustness and generalizability, underexploits the agent's capacity to learn by interacting with the environment, and offers no guarantee of alignment between the environment and the external knowledge.

A large portion of the field also questions the limitations of LLMs, as they lack semantic understanding [2] and formal guarantees of correctness. There is no proof that deploying these agents to effect change in the real world would not result in disaster, but industry is pushing adoption of purely LLM-based technology.

Other fields of research like Model-Based Reinforcement Learning (MBRL) approach the task of learning through the development of combining reinforcement learning and the development of explicit models of the dynamics of the environment. Some approaches include latent models [13, 14] and video generation models [4, 42]. While generally able to learn multiple complex tasks, these models are not easily inspectable.

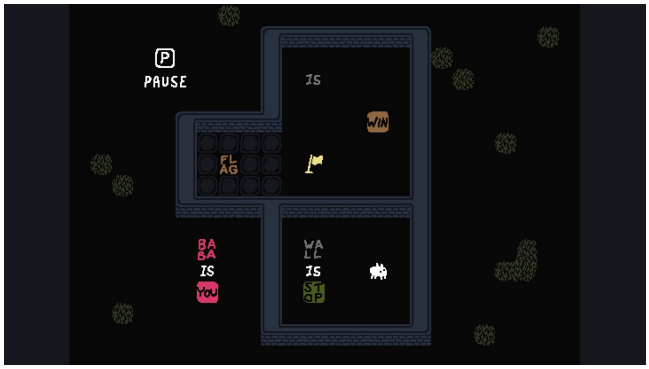


Figure 1: Level 01 of *Baba Is You*. Two rules are active: Player input controls Baba (the white rabbit-like sprite), and walls are impassable objects. To complete the level, the player must form *X IS WIN* ($X \in \{$`FLAG`, `WALL`, `BABA`$\}$) and reach *X*. Our agent tries to form *FLAG IS WIN*, but is stopped by the *WALL*. The agent abducts *WALL IS STOP* is the cause, proposes to test it (by trying another wall, or breaking the rule before trying again). Then it breaks the rule *WALL IS STOP*, making the rest of the map reachable, and finishes the level.

In parallel, work at the interface of agentic LLMs and classical planning uses Planning Domain Definition Language (PDDL) as an intermediate representation. LLM models are used either to generate generalised planning programs or heuristics that are executed by sound planners [35], or to construct and iteratively repair PDDL domain models that are then handed to off-the-shelf planners [12, 27]. Recent survey work frames this pattern as a division of labour, with LLMs serving as model and goal formalisers and symbolic planners providing reliable means-end reasoning [38]. Our proposal adopts the same separation, but applies the LLMs component to abduction and experiment design rather than directly to planning.

3 AN ABDUCTIVE DOMAIN: *BABA IS YOU*

Baba Is You is a 2D gridworld environment (Figure 1) where the game mechanics exist as manipulable physical objects within the environment. These mechanics constitute rules describing the behaviour and properties of entities. Rules are constituted by words (a kind of entity), which can be pushed by the player, rewriting the governing logic of the simulation. For instance, *WALL IS STOP* makes it so wall objects block passage through. Disassembling the sequence by pushing one of the words (*i.e.* breaking the rule) allows objects to pass through walls. A fundamental (and common) rule is the eponymous *Baba Is You*, which determines that entities of kind Baba are player-controlled. Substituting the word "Baba" by others will change the kind of entity that the player controls. Rule configuration changes action effects.

This dynamic makes the game remarkably demanding for AI [3, 11], especially given that the player must learn rule effects via interaction. Although the action space is known (4 directions of movement), its consequences given a rule configuration are not.

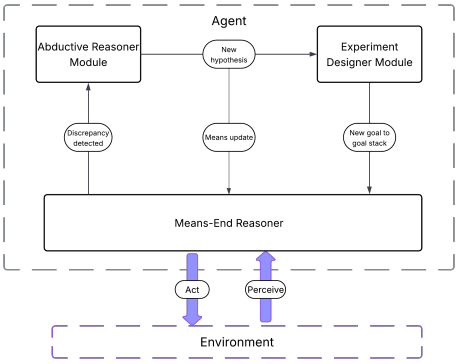


Figure 2: Overview of the proposed Agent Architecture

Traditional search methods are unsuitable without hard-coding the mechanics of the game [25], and planning solutions at action level rather than abstracting to higher-level behaviours (such as composing and breaking rules) supposes a problem due to the high combinatorial complexity of the game [1, 7]. *Baba Is You* also specially undermines Reinforcement Learning (RL) application: rewards can only be confidently given on beating the level, as there is no feasible way to detect when the agent is getting closer to the solution [15].

4 ABDUCTION-BASED AGENTS

The objective of this work is to preliminarily illustrate (in practice) the feasibility of using an LLM-based module to learn the dynamics of an environment. This is done in a two-fold way. Firstly, checking if the hypothesiser is capable of providing sound explanations as to why actions resulted in unexpected facts: in *Baba Is You*, this should always translate to learning the effects of a rule. Secondly, providing a goal or set of goals that, if achieved, validate the hypothesised rule, building a curriculum of actions that make further goals (like winning) easier to attain.

In the proposed framework, an agent is instantiated with a base goal of *winning*, in a *means-ends reasoner* (in this preliminary exploration, a *BFS*). Then, as the agent acts, it finds unexpected situations due to the active rules, hypothesises about them and generates experiments for them. Experiments are manifested in new goals. Centered on the Popperian maxim of the *Bold Hypothesis*, we do not assume or require that hypotheses are correct from the beginning: testing filters many results out. Neither do we require that the experiments cover everything: instead, much like a human player, a misconception may still be useful to beat one level, and later levels may require a refinement (*i.e.* re-hypothesising over a tested rule).

4.1 Agent Architecture

The proposed architecture (Figure 2) is divided into two modules, in addition to a means-end reasoner.

Abductive Reasoner Module (ARM). This module receives a discrepancy between the expected state according to the means-ends reasoner and the actual observed state. The discrepancy, along with a context (*i.e.* performed action, rules currently in action and beliefs about what those rules do), are used to generate a hypothesis to fill the gap in the current knowledge. For example, as shown in Figure 3, if the agent attempts to go through an object that is *STOP* before learning the pertinent rule, the module will receive the context that "the agent was next to the object and expected to move inside it, and instead remained in the same position", and may hypothesise that objects that are *STOP* will not permit passage through them.

Experiment Designer Module (EDM). This module takes the output of the *ARM* and produces some objectives which test the hypothesis. For example, if the hypothesis is that rule *WALL IS STOP* results in walls impeding movement into the object, then the module may suggest trying to go into a different wall, or breaking the rule and trying again. This module is less critical for efficacy (as wrong hypotheses are still corrected via experience) but is there for efficiency: it is faster to test and reject hypotheses than to try potentially long plans repeatedly failing. Proactively refining the knowledge base quickly corrects knowledge in few prompts and interactions with the environment.

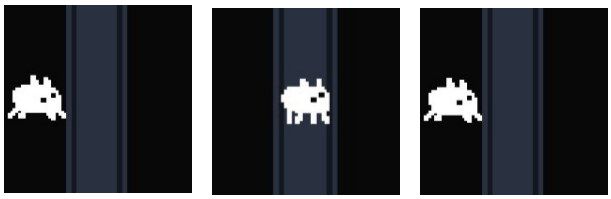
Means-ends reasoner + actuator. This module takes hypotheses on rules, as well as some designer-provided game knowledge (such as the four movement actions and the possibility of pushing text to make and break rules), to form a world model (*i.e.* the *means*, a predictor of how the game state changes when an action is executed). This world model is used together with a goal description to output a plan, consisting in sequences of actions that (according to the world model) fulfil the goal description. It is also in charge of executing the plan and, on detection of a discrepancy with expected results, engaging the Abductive Reasoner Module.

Implementation details. The Abductive Reasoner Module's prompt includes some instructions on the order of reasoning, and the list of tiles where the content expected by the means-ends reasoner differs from the factual content returned. For each of them, it lists the objects contained within the tile (*e.g.* BABA, or WALL, BABA). It also contains the set of *active* rules (provided by the game itself). Finally, it contains the set of (active) beliefs described in JSON format: each active rule in the level that is known (or has an associated hypothesis) and the description of how it is expected to work. The output is a new JSON with the rule and the description of what it does (hypothesised). See Figure 3 for an example of a formatted discrepancy and a formatted hypothesis, respectively.

The Experiment Designer Module receives the stack of current goals and the current state, similarly formatted, as well as a number of experiments to provide. Both of these modules show good results with even small LLM (about 12B parameters).

For the *means-ends reasoning*, we started by implementing the simplest example possible, a *BFS* algorithm using a Python *step* function which returns the next state given current state and an action. For this exploration, this step function is created with a more powerful LLM, which takes the hypotheses on rule knowledge to generate code implementing the step function, which is then called by the search algorithm. As environment complexity increased, the *BFS* scaled too poorly, and we resorted to A^*, and finally to providing human feedback to planning, affecting only in efficiency. However, we strongly believe that the same can be achieved with a much better formalism such as *PDDL* [23] that benefit from having more efficient solvers.

ACTIVE RULES: *BABA IS YOU*, *FLAG IS WIN*, *WALL IS STOP*
KNOWN RULES: *BABA IS YOU*, *FLAG IS WIN*



Previous step | Predicted | Observed

(a)

```
Action Performed:
RIGHT

Map Discrepancies:
Differences at X=16,Y=13:
- pre-action              : BABA
- post-action (real)      : BABA
- post-action (simulated) : <EMPTY>
Differences at X=17,Y=13:
- pre-action              : WALL
- post-action (real)      : WALL
- post-action (simulated) : WALL,BABA

Active Rules:
- BABA IS YOU
- WALL IS STOP

Current Beliefs:
{
  "BABA IS YOU": {
    "description": ...
  }
}
```

(b)

```
{
  "WALL IS STOP": {
     "description": "Objects with the property 'STOP' prevent movement through them.",
     "reasoning": ...
  }
}
```

(c)

Figure 3: An example of the process of abduction after a discrepancy. (a) In a situation in which *WALL IS STOP* is unknown for the agent, the actuator module is running a plan in which BABA traverses a WALL, predicting that the former will go through the latter. When it observes that the observed state differs from its predcitions (BABA does not go through the wall), it triggers the pipeline to construct a discrepancy. (b) The discrepancy is described and included in the prompt that the Abductive Reasoner Module receives. (c) The Abductive Reasoner Module hypothesises that the rule *WALL IS STOP* disallows entities moving through them.

5 FIRST STEPS FOR AN ABDUCTIVE AGENT

We established a framework for conducting the experiments on the proposed architecture within *Baba Is You*, spanning *Overworld* levels 00-05, in which we used different inference engines depending on the needs of each module[1]. Specifically, the Abductive Reasoner Module and Experiment Designer Module have used an open source LLM *GLM-4.5-Air-IQ4_XS* [46][2] (12B active parameters) as their inference engine. For the *means-ends reasoner*, *Gemini-3 Pro* [39] has been used to update the *step* function of the search algorithm. The levels tested present an increasing progression of complexity[3]:

- **Level 00** introduces four rule properties: *YOU* (designates the controllable entity), *WIN* (defines the victory condition), *PUSH* (allows objects to be moved), and *STOP* (prevents movement through the object). It requires no rule manipulation to win.
- **Level 01** presents rule manipulation as a core mechanic, requiring the construction, modification, and destruction of rules to complete the level (breaking *WALL IS YOU*, forming *X IS WIN*).
- **Level 02** is the first level where the player's identity is reassigned (from *BABA* to *WALL*). Entity changes aside, it is the same as 01 but with one less entity present (the *FLAG*).
- **Level 03** adds the *DEFEAT* property: any player-controlled object is destroyed upon contact with a *DEFEAT* object.
- **Level 04** introduces the *SINK* property, which destroys all objects overlapping with the *SINK* object (including itself) when they occupy the same tile.

[1]Code, experiments, and prompts available at https://github.com/levyy7/BabaIsBot
[2]A quantisation downloaded from https://huggingface.co/bartowski/zai-org_GLM-4.5-Air-GGUF
[3]Level layouts and solutions can be found in: https://babaiswiki.fandom.com/wiki/Map

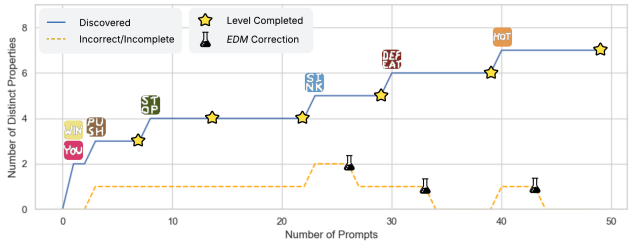


Figure 4: Agent property discovery and correction progress.

- **Level 05** introduces *HOT* and *MELT*: *MELT* objects are destroyed when they overlap *HOT* objects.

We evaluated our system playing *Baba Is You* on levels 00-05. With no initial knowledge on rules, the agent is run sequentially from level 0 to level 5. This way, rules are incrementally learnt from the environment. Past Level 01, the search space became too large for a simple BFS, and past Level 03 it was too much for A^*. For the purpose of this paper, the planning for the latter 2 levels was done by a human strictly adhering to the knowledge base of the agent. Finally, the results are evaluated regarding: 1) overall level-solving ability, 2) effectiveness in discovering new properties, and 3) efficiency of rectifying incorrect or incomplete rules.

Figure 4 illustrates the cumulative number of distinct properties discovered as a function of the total number of prompts issued to an LLMs. A second line shows the number of incorrect/incomplete properties within the system, with flasks denote belief correction via an experiment of the Experiment Designer Module. Jumps in

Discrep. ID	Active rules	Correct beliefs	Action	Before	Predicted	Real
1	BABA IS YOU WALL IS STOP	BABA IS YOU	Right	(16, 13): BABA (17, 13): WALL	(17, 13): BABA, WALL	(16, 13): BABA (17, 13): WALL
2	*(1)* FLAG IS WIN ROCK IS PUSH	BABA IS YOU FLAG IS WIN	Right	(16, 11): BABA (17, 11): ROCK	(17, 11): BABA, ROCK	(17, 11): BABA (18, 11): ROCK
3	*(2)* WATER IS SINK	BABA IS YOU FLAG IS WIN	Down	(10, 7): BABA (10, 8): WATER	(10, 8): BABA, WATER	*(empty)*
4	*(2)* SKULL IS DEFEAT	BABA IS YOU FLAG IS WIN	Right	(9, 13): BABA (10, 13): SKULL	(10, 13): BABA, SKULL	(10, 13): SKULL
5	*(2)* LAVA IS HOT FLAT IS MELT	BABA IS YOU FLAG IS WIN	Right	(17, 8): BABA (18, 8): LAVA	(18, 8): BABA, LAVA	(18, 8): LAVA

Table 1: List of discrepancies that formed part of the user study. A number between parenthesis in the *Active rules* column indicates that all active rules from that particular discrepancy were also active (*e.g.* , *(1)* in row 2 indicates that *BABA IS YOU* and *WALL IS STOP* were also active in discrepancy 2).

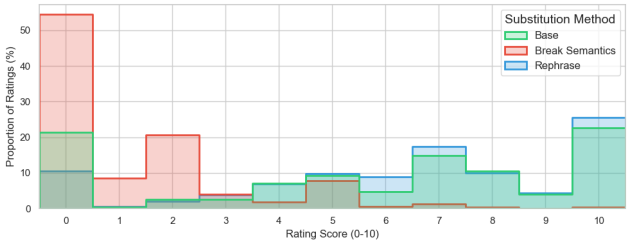


Figure 5: Impact of Isomorphic Substitutions on Abduction

the plot align with transitions to new levels. In terms of abduction errors, the results show that three rule descriptions were incorrect. Two are rectified immediately, but the Experiment Designer Module fails to correct *PUSH* initially. Still, an incorrect hypothesis about *PUSH* is enough to solve many levels before it requires correction.

To test whether the abduction relies on prior knowledge of the game (*i.e.* contamination from LLM training), and the relevance of the words to solving the game, we perform another experiment. An isomorphic substitution is applied to the rules and game state: 1) preserving semantics and syntax (*e.g. WALL IS STOP*), 2) rephrasing the state by using synonyms and altering phrase structure (*e.g. BARRIER IS IMMOVABLE* replaces *WALL IS STOP*), and 3) breaking state semantics by substituting words with semantically unrelated terms (*e.g. PENTAGON IS RED* replaces *WALL IS STOP*). These permutations are applied to various scenarios. The Abductive Reasoner Module runs 25 inferences for every $\{substitution, scenario\}$ pair, with scenarios collected from discrepancies[4] found within the previous experiment (see Table 1). The resulting hypotheses are then evaluated in a user study, in which 3 experts rate the veracity and relevance of the generated output, shown in Figure 5.

A similar distribution is observed for the baseline and rephrase methods overall, although the baseline is more likely to generate useless hypotheses than the rephrase method, possibly due to the less awkward syntax of the latter. Breaking semantic knowledge results in obvious performance degradation, indicating that the LLM takes advantage of the meaning of the words, but does not necessarily benefit from knowledge of the game in its training data.

[4]Note that some of the discrepancies included in the user study have been modified: some learnt knowledge that would have been passed to the Abductive Reasoner Module in the original experiment has been omitted to give less context to the module, thus creating more challenging versions of the original discrepancies.

6 DISCUSSION

Within the first 6 levels of *Baba Is You*, we found that hypothesis and verification via LLM can reasonably enable *means-ends* reasoning to solve problems where action spaces are hard. We believe that working in this direction will allow the construction of strictly rational agents that leverage capabilities only available to LLM-based agents right now. Furthermore, if experimental goals are separate from other goals, the agent could feasibly test its abductions in test environments (avoiding experimental behaviour causing risks in a real environment). This is especially important for agents working in domains where rational behaviour is a requirement.

Our results show impressive abductive performance even for a low-parameter model, and when abductions were wrong, the workflow was quick to correct it in most cases. Converting abduction hypotheses directly into *means* in a *means-ends* reasoner formalism with efficient solvers should be the next step to validate the feasibility of abducting agents.

The substitution experiments show that performance is not too brittlely related to LLM knowledge of the game and that, while the discrepancies provided can be enough to formulate correct hypotheses despite having no semantic knowledge, LLMs (just like players) benefit from the semantic hints in the wording of rules.

In agentic AI, a large part of the community is advocating for end-to-end LLM-based planning despite their strong limitations and lack of guarantees. Hybrid architectures can add the guarantees of deductive inferences and interface with powerful abduction mechanisms of these new models, even without fine-tuning or large parameter counts. The applicability of abduction to new problems is a very interesting paradigm that could benefit systems using Belief-Desire-Intention (BDI) or Answer Set Programming (ASP).

ACKNOWLEDGEMENTS

This work has been partially supported by the AIXPERT (Grant agreement ID: 101214389) Horizon Europe project and V. Gimenez-Abalos' and A. Tormos' fellowships within the "Generación D" initiative, Red.es, MTDFP, for talent attraction (C005/24-ED CV1). Funded by the European Union NextGenerationEU funds, through PRTR.

REFERENCES

[1] Zergham Ahmed, Joshua B. Tenenbaum, Christopher J. Bates, and Samuel J. Gershman. 2025. Synthesizing world models for bilevel planning. https://doi.org/10.48550/arXiv.2503.20124 arXiv:2503.20124 [cs].

[2] Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. ACM, Virtual Event Canada, 610–623. https://doi.org/10.1145/3442188.3445922

[3] M. Charity and Julian Togelius. 2022. Keke AI Competition: Solving puzzle levels in a dynamically changing mechanic space. https://doi.org/10.48550/arXiv.2209.04911 arXiv:2209.04911 [cs].

[4] Jie Cheng, Ruixi Qiao, YINGWEI MA, Binhua Li, Gang Xiong, Qinghai Miao, Yongbin Li, and Yisheng Lv. 2025. Scaling Offline Model-Based RL via Jointly-Optimized World-Action Model Pretraining. In *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=T1OvCSFaum

[5] François Chollet. 2019. On the Measure of Intelligence. https://doi.org/10.48550/arXiv.1911.01547 arXiv:1911.01547 [cs].

[6] Andy Clark. 2001. *Being there: putting brain, body, and world together again* (1. paperback ed ed.). MIT Press, Cambridge, Mass.

[7] Nathan Cloos, Meagan Jens, Michelangelo Naim, Yen-Ling Kuo, Ignacio Cases, Andrei Barbu, and Christopher J. Cueva. 2024. Baba Is AI: Break the Rules to Beat the Benchmark. https://doi.org/10.48550/arXiv.2407.13729 arXiv:2407.13729 [cs].

[8] Luciano Floridi, Jessica Morley, Claudio Novelli, and David Watson. 2025. What Kind of Reasoning (if any) is an LLM actually doing? On the Stochastic Nature and Abductive Appearance of Large Language Models. arXiv:2512.10080 (Dec. 2025). https://doi.org/10.48550/arXiv.2512.10080 arXiv:2512.10080 [cs].

[9] Jerry A. Fodor. 1983. *The Modularity of Mind*. The MIT Press. https://doi.org/10.7551/mitpress/4737.001.0001

[10] ARC Foundation. 2026. ARC-AGI-3: A New Challenge for Frontier Agentic Intelligence. *arXiv preprint arXiv:2603.24621* (2026).

[11] Jonathan Geller. 2022. Baba is You is Undecidable. *IEEE Conference on Computatonal Intelligence and Games, CIG* (4 2022). https://doi.org/10.1109/CoG57401.2023.10333219

[12] Lin Guan, Karthik Valmeekam, Sarath Sreedharan, and Subbarao Kambhampati. 2023. Leveraging pre-trained large language models to construct and utilize world models for model-based task planning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems* (New Orleans, LA, USA) *(NIPS '23)*. Curran Associates Inc., Red Hook, NY, USA, Article 3459, 14 pages.

[13] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. 2024. Mastering Diverse Domains through World Models. https://arxiv.org/abs/2301.04104

[14] Nicklas Hansen, Hao Su, and Xiaolong Wang. 2024. TD-MPC2: Scalable, Robust World Models for Continuous Control. In *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=Oxh5CstDJU

[15] Haoyu Liu, Fan-Yun Sun, Frieda Rong, Kumiko Nakajima, Nicholas Haber, and Shima Salehi. 2023. Characterizing Learning Progress of Problem-Solvers Using Puzzle-Solving Log Data. (July 2023). https://doi.org/10.5281/ZENODO.8115768 Publisher: Zenodo.

[16] Patrick J. Hayes. 1979. The Naive Physics Manifesto. *Expert Systems in the Micro-Electronic Age* (1979). https://cir.nii.ac.jp/crid/1570009749139377408 Publisher: Edinburgh University Press.

[17] Carl G. Hempel and Paul Oppenheim. 1948. Studies in the Logic of Explanation. *Philosophy of Science* 15, 2 (April 1948), 135–175. https://doi.org/10.1086/286983

[18] Wen-Chao Hu, Wang-Zhou Dai, Yuan Jiang, and Zhi-Hua Zhou. 2025. Efficient rectification of neuro-symbolic reasoning inconsistencies by abductive reflection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 17333–17341.

[19] Yu-Xuan Huang, Wang-Zhou Dai, Yuan Jiang, and Zhi-Hua Zhou. 2023. Enabling knowledge refinement upon new concepts in abductive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 7928–7935.

[20] Naveen Krishnan. 2025. AI Agents: Evolution, Architecture, and Real-World Applications. https://doi.org/10.48550/arXiv.2503.12687 arXiv:2503.12687 [cs].

[21] Kaiqu Liang, Haimin Hu, Xuandong Zhao, Dawn Song, Thomas L. Griffiths, and Jaime Fernández Fisac. 2025. Machine Bullshit: Characterizing the Emergent Disregard for Truth in Large Language Models. arXiv:2507.07484 (2025). https://doi.org/10.48550/arXiv.2507.07484 arXiv:2507.07484 [cs].

[22] Drew McDermott. 1987. A critique of pure reason. *Computational Intelligence* 3, 1 (1987), 151–160. https://doi.org/10.1111/j.1467-8640.1987.tb00183.x _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8640.1987.tb00183.x.

[23] D. McDermott, M. Ghallab, A. Howe, Craig A. Knoblock, A. Ram, M. Veloso, Daniel S. Weld, and D. Wilkins. 1998. PDDL-the planning domain definition language.

[24] Oriol Miró I López Feliu. 2026. *Evaluating planning capabilities in agentic systems*. Master Thesis. UPC, Facultat d'Informàtica de Barcelona, Departament de Ciències de la Computació. https://hdl.handle.net/2117/460615

[25] Christopher Olson, Lars Wagner, and Alexander Dockhorn. 2023. Evolutionary Optimization of Baba Is You Agents. In *2023 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, Chicago, IL, USA, 1–8. https://doi.org/10.1109/CEC53210.2023.10253977

[26] OpenAI, Josh Achiam, et al. 2024. GPT-4 Technical Report. https://doi.org/10.48550/arXiv.2303.08774 arXiv:2303.08774 [cs].

[27] James Oswald, Kavitha Srinivas, Harsha Kokel, Junkyu Lee, Michael Katz, and Shirin Sohrabi. 2024. Large Language Models as Planning Domain Generators. *Proceedings of the International Conference on Automated Planning and Scheduling* 34, 1 (May 2024), 423–431. https://doi.org/10.1609/icaps.v34i1.31502

[28] Melissa Z Pan, Negar Arabzadeh, Riccardo Cogo, Yuxuan Zhu, Alexander Xiong, Lakshya A Agrawal, Huanzhi Mao, Emma Shen, Sid Pallerla, Liana Patel, et al. 2025. Measuring Agents in Production. *arXiv preprint arXiv:2512.04123* (2025).

[29] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative Agents: Interactive Simulacra of Human Behavior. *UIST 2023 - Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (4 2023). https://doi.org/10.1145/3586183.3606763

[30] Karl Popper. 1962. *Conjectures and Refutations: The Growth of Scientific Knowledge*. Routledge.

[31] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. 2022. A Generalist Agent. https://doi.org/10.48550/arXiv.2205.06175 arXiv:2205.06175 [cs].

[32] Stuart J. Russell and Peter Norvig. 2016. *Artificial intelligence: a modern approach* (third edition, global edition ed.). Pearson, Boston Columbus Indianapolis.

[33] Ranjan Sapkota, Konstantinos I. Roumeliotis, and Manoj Karkee. 2026. AI Agents vs. Agentic AI: A Conceptual taxonomy, applications and challenges. *Information Fusion* 126 (Feb. 2026), 103599. https://doi.org/10.1016/j.inffus.2025.103599

[34] Johannes Schneider. 2025. Generative to Agentic AI: Survey, Conceptualization, and Challenges. https://doi.org/10.48550/arXiv.2504.18875 arXiv:2504.18875 [cs].

[35] Tom Silver, Soham Dan, Kavitha Srinivas, Joshua B. Tenenbaum, Leslie Kaelbling, and Michael Katz. 2024. Generalized Planning in PDDL Domains with Pretrained Large Language Models. *Proceedings of the AAAI Conference on Artificial Intelligence* 38, 18 (Mar. 2024), 20256–20264. https://doi.org/10.1609/aaai.v38i18.30006

[36] Mirac Suzgun, Tayfun Gur, Federico Bianchi, Daniel E. Ho, Thomas Icard, Dan Jurafsky, and James Zou. 2025. Language models cannot reliably distinguish belief from knowledge and fact. *Nature Machine Intelligence* 7, 11 (Nov. 2025), 1780–1790. https://doi.org/10.1038/s42256-025-01113-8

[37] Weihao Tan, Wentao Zhang, Xinrun Xu, Haochong Xia, Ziluo Ding, Boyu Li, Bohan Zhou, Junpeng Yue, Jiechuan Jiang, Yewen Li, Ruyi An, Molei Qin, Chuqiao Zong, Longtao Zheng, Yujie Wu, Xiaoqiang Chai, Yifei Bi, Tianbao Xie, Pengjie Gu, Xiyun Li, Ceyao Zhang, Long Tian, Chaojie Wang, Xinrun Wang, Börje F. Karlsson, Bo An, Shuicheng Yan, and Zongqing Lu. 2024. Cradle: Empowering Foundation Agents Towards General Computer Control. https://doi.org/10.48550/arXiv.2403.03186 arXiv:2403.03186 [cs].

[38] Marcus Tantakoun, Christian Muise, and Xiaodan Zhu. 2025. LLMs as Planning Formalizers: A Survey for Leveraging Large Language Models to Construct Automated Planning Models. In *Findings of the Association for Computational Linguistics: ACL 2025*, Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (Eds.). Association for Computational Linguistics, Vienna, Austria, 25167–25188. https://doi.org/10.18653/v1/2025.findings-acl.1291

[39] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805* (2023).

[40] Francisco J. Varela, Evan Thompson, and Eleanor Rosch. 1991. *The Embodied Mind: Cognitive Science and Human Experience*. MIT Press.

[41] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Anima Anandkumar, Ut Austin, and Uw Madison. 2023. Voyager: An Open-Ended Embodied Agent with Large Language Models. (5 2023). https://arxiv.org/abs/2305.16291v2

[42] Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. 2024. Driving into the future: Multiview visual forecasting and planning with

world model for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14749–14759.

[43] Wen-Da Wei, Xiao-Wen Yang, Jie-Jing Shao, and Lan-Zhe Guo. 2025. Curriculum abductive learning for mitigating reasoning shortcuts. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*. 6533–6541.

[44] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. ReAct: Synergizing Reasoning and Acting in Language Models. *11th International Conference on Learning Representations, ICLR 2023* (10 2022). https://arxiv.org/abs/2210.03629v3

[45] Asaf Yehudai, Lilach Eden, Alan Li, Guy Uziel, Yilun Zhao, Roy Bar-Haim, Arman Cohan, and Michal Shmueli-Scheuer. 2025. Survey on Evaluation of LLM-based Agents. https://doi.org/10.48550/arXiv.2503.16416 arXiv:2503.16416 [cs].

[46] Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, et al. 2025. Glm-4.5: Agentic, reasoning, and coding (arc) foundation models. *arXiv preprint arXiv:2508.06471* (2025).

[47] Bingxi Zhao, Lin Geng Foo, Ping Hu, Christian Theobalt, Hossein Rahmani, and Jun Liu. 2025. LLM-based Agentic Reasoning Frameworks: A Survey from Methods to Scenarios. https://doi.org/10.48550/arXiv.2508.17692 arXiv:2508.17692 [cs].