Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

Hypothesis-driven Model Expansion under Uncertainty for Open-World Robot Planning

Anxing Xiao[1], Hanbo Zhang[1,2], Tianrun Hu[1,2], David Hsu[1,2]
[1]School of Computing, National University of Singapore
[2]Smart Systems Institute, National University of Singapore

***Abstract*—We consider an open-world planning setting in which service robots operate in unknown environments with incomplete knowledge of objects and actions. Traditional closed-world approaches with pre-programmed knowledge bases fail when robots encounter unexpected situations and tasks, posing a fundamental challenge for autonomous knowledge expansion in human environments. In this work, we propose an open-world planning framework that enables robots to automatically generate, verify, and update hypotheses about their abstract world models. Our key insight is to explicitly maintain uncertainty-aware knowledge expansion and integrate hypothesis verification into goal-reaching planning. Our framework, Hypothesis-driven Uncertainty-aware Model Expansion (HUME), leverages foundation models to generate initial hypotheses over states and transitions, and applies automated planning to produce action sequences that jointly address hypothesis verification and task execution. Through iterative execution and refinement, the robot expands its knowledge by incorporating verification feedback from the foundation models when hypotheses prove incorrect. Extensive experiments in simulated and real-world environments demonstrate that our framework enables autonomous knowledge expansion and effective operation in open-world settings. These results indicate that integrating uncertainty-aware model expansion from robot foundation models with planning advances the practical deployment of household service robots. Project website: open-world-planning.github.io**

Fig. 1. Illustration of a service robot operating in an open-world scenario. To fulfill tasks such as *"heat up a chicken burger"*, the robot must expand its abstract world model and account for uncertainty during planning.

I. Introduction

Home service robots operate under substantial uncertainty in real-world household environments, where objects may be hidden, their attributes unknown, and the effects of actions underspecified. Consider a daily task such as *"serve a heated chicken burger"*, as shown in Fig. 1. To accomplish this task, a robot must infer the burger's location, determine whether it is made of chicken, and, in some cases, reason about appliance-specific action effects, such as how much the heating time increases when pressing the '+' button on a microwave. These challenges make household environments inherently open-world. Classical automated planning relies on predefined state representations and transition models [4, 18, 19]. While effective in well-specified domains, this reliance on manually designed knowledge and closed-world assumptions limits generalization to open-world settings with underspecified knowledge. In contrast, Large Language Models (LLMs) exhibit remarkable reasoning and generalization to predict plausible plans without explicitly modeling states or transition dynamics [5, 16, 24, 25, 34]. Yet, in open-world settings with partial information, these properties can blur the boundary between reasoning grounded in known facts and reasoning based on hallucinated knowledge. This motivates a central question of this work: *how can we combine structured reasoning with the unstructured open-world knowledge of foundation models?*

To bridge this gap, recent research has explored using foundation models to construct or fix symbolic representations of the environment [1, 6, 15, 35, 37, 61–63]. These approaches allow models to generate planning domains [1, 35, 37, 61–63], infer missing preconditions or effects [6, 15], and construct probabilistic world models [11]. While promising, such methods largely rely on passive inference: once a model is generated, it is typically assumed to be correct or only updated indirectly through execution failures. However, ignoring uncertainty in expanded world knowledge can cause silent failures, as plans may still be executed even when object attributes or action effects are incorrectly predicted.

This limitation motivates our core insight: *in open-world planning, model expansion itself must be treated as an uncertain and actively verifiable process*. From a decision-theoretic perspective, this corresponds to a Bayes-adaptive view of planning [17], in which the agent explicitly reasons over uncertainty in the underlying world model and selects actions that jointly achieve task goals and reduce model uncertainty.

Building on this perspective, we propose **HUME** (**H**ypothesis-driven **U**ncertainty-aware **M**odel **E**xpansion), an open-world planning framework that integrates structured symbolic reasoning with the unstructured commonsense priors of language models through explicit, uncertainty-aware world model expansion (Fig. 2). The robot maintains a

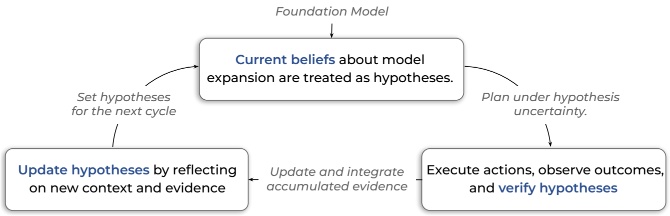

Fig. 2. High-level idea. It combines foundation-model priors as hypotheses with grounded interaction to iteratively expand its environment model.

set of object-centric hypotheses representing unknown or uncertain aspects of the environment, such as object locations, attributes, or the effects of actions applied to them. These hypotheses are generated by foundation models and are treated as uncertain latent variables rather than as facts. Given this formulation, the planning problem is formulated over an augmented representation that includes both the known model and these uncertain hypotheses. Through appropriate determinization [64], we demonstrate that a classical planner can be applied to produce partially grounded plans that explicitly interleave task execution with verification. During the execution, the foundation models also help ground and update these hypotheses based on new sensory observations and historical information. These hypothesis instances thus serve as both imagination and memory, enabling continual active expansion of the world model. We demonstrate the generality of HUME across diverse open-world tasks, including Block Processing World, mobile manipulation in unknown environments, and appliance operation. Experiments in both simulation and real-world show that our framework enables autonomous knowledge expansion and effective task execution in open-world settings for both classical formal planners and language model planners. Our results highlight the importance of integrating uncertainty-aware model expansion into planning, which we believe is critical for deploying service robots in real open-world environments.

II. Related Work

A. AI Planning with Foundation Models

Recent advances in foundation models have spurred growing interest in using language models for zero-shot decision-making, predicting planning outcomes without explicit forward search. These approaches leverage commonsense knowledge encoded in LLMs to directly generate structured plans, such as action sequences [5, 16, 24, 25, 39, 42, 46], or executable code [34, 47, 48, 52]. Such plans are often refined through self-reflection using passive feedback [39, 45, 49, 54], while retrieval-augmented methods further incorporate past experiences [55]. Despite these advances, LLM-based planners still struggle with long-horizon coherence, constraint enforcement, and active information acquisition under uncertainty [3, 9, 60, 66]. To mitigate these limitations, recent work integrates LLMs with classical planning frameworks to combine symbolic structure. In such hybrid systems, LLMs could assist with goal interpretation [36, 59], planning tool selection [3], and heuristic guidance [33, 46, 60, 67]. More recent approaches integrate the concept of model acquisition [8, 14] with foundation models to extend or revise planning domains, for example by generating domain descriptions from demonstrations [1, 35, 37, 58, 61, 62], performing program induction [11, 65], inventing novel spatial predicates [9, 32], or inferring missing preconditions and effects [6, 15]. However, such domain knowledge is often static after generation or updated only through passive external feedback [6, 11, 68]. Building on this line of work, we explore uncertainty-aware world model expansion, where robots plan actions to reach informative states for evaluating expanded knowledge. From this perspective, our approach casts embodied question answering [12, 43] as a proactive process and integrates it into the full task-planning pipeline, by explicitly encoding verification queries within the planning problem itself.

B. Planning in the Open World

To relax closed-world assumptions, some approaches incorporate external knowledge bases as priors [7, 27], and rely on replanning when outcomes deviate from expectations. Without explicitly modeling uncertainty, such planners struggle to reason about silence failures and risks. Belief-space planning addresses this challenge by maintaining probabilistic beliefs over world states [10, 20, 29], enabling principled reasoning under partial observability and environmental uncertainty. This formulation has been widely and successfully applied to open-world tasks, e.g., object search [2, 53, 57]. However, constructing such structured belief models typically requires extensive hand-engineering of concepts such as objects, attributes, and action effects, along with their associated priors and belief update mechanisms, which significantly limits scalability in open-world scenarios. In contrast, LLMs encode rich unstructured priors over world knowledge, enabling implicit reasoning about objects, actions, and outcomes, which is leveraged by [50] to generate information-gathering actions in short-horizon tasks. To bridge unstructured priors with uncertainty-aware planning, [11] uses LLMs to specify observation models and belief updates over predefined POMDP abstractions, but does not explicitly model uncertainty in the generated knowledge and relies on predefined passive feedback for updates. [51] demonstrates the ability of LLMs to generate high-quality particle beliefs over object locations and human goals for online POMDP planning. Recent work [66] combines visual predicate evaluation with uncertainty-aware planners for undefined object attributes, yet relies on manually specified information-gathering actions and does not address object locations or action effects. Our method addresses the gaps by integrating LLM-derived priors into structured, uncertainty-aware knowledge expansion as hypotheses over factorized object locations, attributes, and action effects, conceptually aligning with Bayes-adaptive planning [17], in which uncertainty arises from unknown or evolving models of the world.

III. Problem Formulation

A. Problem Setting

We consider planning under an incomplete world model, in which a robot operates in an environment with missing

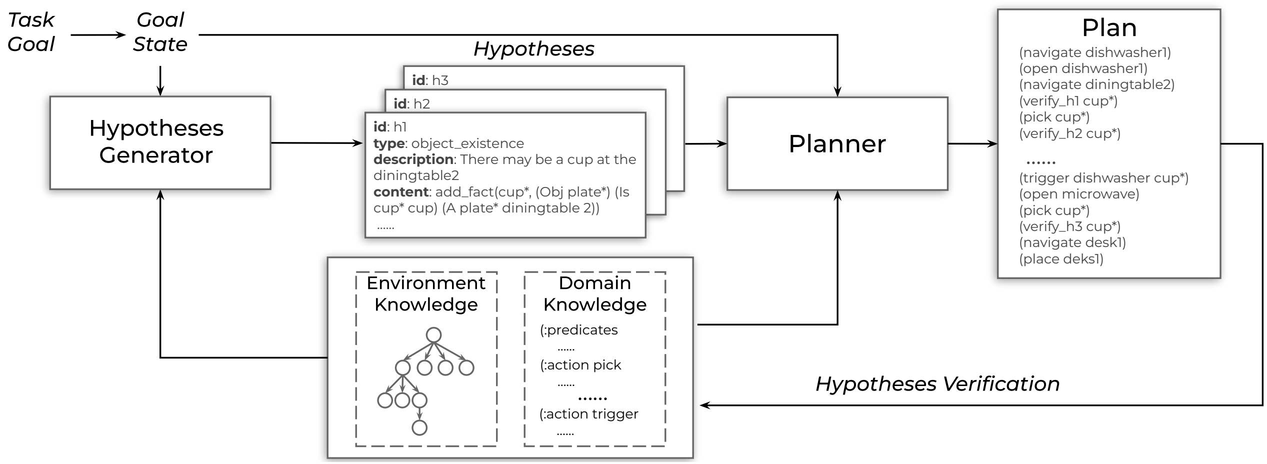

Fig. 3. Overview of the HUME. The robot iteratively generates hypotheses to expand its model, plans with a model augmented by uncertain hypotheses, and executes actions to verify and update these hypotheses using newly acquired observations.

object-centric knowledge. The robot is equipped with an abstract environment model $\mathcal{M} = \langle \mathcal{S}, \mathcal{P}, \mathcal{A}, T \rangle$, where $\mathcal{S}$ denotes the symbolic state space defined by a predicate set $\mathcal{P}$, $\mathcal{A}$ is the action space, and $T : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ is a deterministic transition function over known predicates. The true environment is governed by an unknown ground-truth model $\mathcal{M}_{\text{gt}} = \langle \mathcal{S}_{\text{gt}}, \mathcal{P}_{\text{gt}}, \mathcal{A}, T_{\text{gt}} \rangle$, which may include additional objects, predicates, or transition effects not captured in $\mathcal{M}$. As a result, $\mathcal{M}$ provides only a partial abstraction of the real world. Tasks are specified via natural language instructions L, corresponding to an unknown goal set $\mathcal{G} \subseteq \mathcal{S}_{\text{gt}}$. The robot starts from an initial state $s_0 \in \mathcal{S}$ and aims to reach a state $s \in \mathcal{G}$. We therefore define a planning problem as $\Pi = \langle s_0, \mathcal{G}, \mathcal{M} \rangle$. Since critical knowledge required to reach $\mathcal{G}$ may be missing from $\mathcal{M}$, planning directly in the initial model is generally infeasible. The central challenge is therefore to plan and act effectively under model incompleteness.

Algorithm 1 Hypothesis-Driven Open-World Planning

```
Require: Task L, Model M_0, State s
Initialize hypothesis set H ← ∅, M ← M_0
History C_hist ← ∅
G ← TRANSLATOR(L)
while not done and i < I_max do
    i ← i + 1
    H ← HYPOTHESESGENERATOR(H, M, s, G, C_hist)
    π ← PLANNER(s, G, M, b(H))
    for action a ∈ π do
        if a = a^h_verify then
            v(h) ← VERIFY(h, C_hist)
            C_hist ← C_hist ∪ {v(h)}
        else
            s, obs ← EXECUTE(a)
            C_hist ← C_hist ∪ {(a, s, obs)}
        end if
    end for
end while
```

B. Planning with Uncertain Model Expansion

To address model incompleteness, we introduce a hypothesis-driven planning formulation that explicitly represents missing knowledge. We define a model expansion as a set of factorized hypotheses $\mathcal{H} = \{h_1, \ldots, h_n\}$, where each hypothesis h_i encodes an object-centric fact, such as object existence, attribute membership, or previously unmodeled action effects. Collectively, these hypotheses parameterize an expanded world model. Hypotheses are generated by a function $g : (\mathcal{M}, L) \rightarrow \mathcal{H}$, which produces task-relevant hypotheses conditioned on the current model and instruction, avoiding exhaustive reasoning over all unknowns. The truth of each hypothesis h_i is defined by $v(h_i) \in \{0, 1\}$. The robot maintains a belief over the truth of hypotheses $b(\mathcal{H}) = p(v_1, \ldots, v_n \mid \tau)$, capturing uncertainty about the true environment model. For each hypothesis h_i, a verification action is generated by a function $\phi : h_i \rightarrow A^i_{\text{verify}}$, where A^i_{verify} denotes information-gathering actions. The overall action space is augmented as $\mathcal{A}' = \mathcal{A} \cup \{A^i_{\text{verify}}\}$. Decision-making is formulated as a Bayes-adaptive MDP [17] with state $(s_0, b(\mathcal{H}))$. Given an action $a \in \mathcal{A}'$, transitions follow the expected model $\mathcal{M}^* = \mathcal{M} \circ \mathcal{H}$ under the current hypotheses, and the hypotheses are updated based on observed outcomes. The planning objective is to compute a plan $\pi : \mathcal{S} \times b(\mathcal{H}) \rightarrow \mathcal{A}^*$, i.e., an action sequence, that maximizes expected cumulative reward by jointly reasoning over task execution and hypothesis verification.

IV. Open-World Planning with Hypothesis-Driven Uncertainty-Aware Model Expansion

We propose a hypothesis-driven framework for open-world planning in which a robot incrementally expands its world knowledge by generating, reasoning over, and verifying missing object-centric knowledge. As illustrated in Fig. 3, the framework operates in an iterative loop of model expansion, planning, and execution. It starts from an initial symbolic model that provides only a partial abstraction of the environment, consisting of environment knowledge represented as a scene graph and domain knowledge represented as a PDDL domain file. Given a task-specified goal state that cannot be achieved under the current model, a hypothesis generator identifies missing knowledge and formulates object-centric hypotheses, such as unknown object locations, novel attributes, or unmodeled action effects required for the task. The planner treats these hypotheses as uncertain variables and, given the current model, hypotheses, and goal, constructs action sequences that both make progress toward the task and verify the hypotheses on which the plan depends. During execution, verification actions invoke a vision–language model to confirm or reject hypotheses, triggering model updates and replanning when necessary. Through this closed-loop process (Algorithm 1), the robot progressively adapts its world model and improves performance in previously unseen environments.

A. Model Initialization

We assume that for each task class there exists an initial symbolic model $\mathcal{M}$ capturing the robot's prior knowledge of the environment. This model encodes known objects and locations, a predicate vocabulary, and transition dynamics defined by action preconditions and effects. In our framework, the world model is decomposed into environment knowledge and domain knowledge. Environment knowledge is represented as a scene graph of rooms, locations, and objects, along with their relations. Domain knowledge is specified in PDDL format [18], defining the preconditions and effects of actions. An example action definition is shown below.

```
(:action pick
  :parameters (?r - Robot ?o - Obj ?l - Location)
  :precondition (and (HandEmpty ?r) (At ?r ?l)
    (At ?o ?l))
  :effect (and (¬HandEmpty ?r) (Holding ?r ?o)
    (¬At ?o ?l))
)
```

Since PDDL provides a deterministic description of action dynamics, the initial model typically captures only limited transition knowledge. In practice, this model may be obtained through manual specification by domain experts or automatically extracted offline from large-scale datasets by prompting language models with task descriptions and demonstrations [62]. In open-world settings, however, critical information is inevitably missing, including object locations, object attributes, and object-specific action effects.

B. Model Expansion as Hypotheses

To reason beyond the initial symbolic abstraction, we explicitly represent model expansion through a set of hypotheses. Each hypothesis $h \in \mathcal{H}$ corresponds to an object-centric fact with an initially unknown truth value. Hypotheses act as tentative extensions to the world model and can be incrementally introduced, reasoned over, and verified. For example, consider the following object existence hypothesis:

```
{
    id: h1,
    type: object_existence,
    description: There may be a plate in fridge2 in
     kitchen2,
    content: add_fact(plate⋆, (Obj plate⋆) (Is
    plate⋆ plate) (At plate⋆ fridge2)),
    condition: None,
    verification_condition: (At ?r fridge2) (IsOpen
     fridge2)
}
```

This hypothesis introduces a hypothetical object instance (`plate⋆`) into the symbolic model and assumes its possible existence at a specific location. The `content` field defines an API call that modifies the PDDL problem file; in principle, this API can also be generated by a language model. The verification condition specifies how the hypothesis can be tested, for example by requiring the robot to be co-located with the object in an accessible open area. More hypotheses can be constructed by conditioning on other ones. For instance, an attribute hypothesis may depend on a previous hypothesis:

```
{
    id: h2,
    type: object_attribute,
    description: The plate⋆ is orange,
    content: add_fact(plate⋆, (Orange plate⋆)),
    condition: h1,
    verification_condition: (Holding ?r plate⋆)
}
```

Here, hypothesis `h2` assumes an attribute of the object introduced by `h1` and is meaningful only if the object exists. Its verification requires a more informative interaction, such as the robot holding the object. Hypotheses can also encode action effects, such as the outcome of pressing a button:

```
{
    id: h3,
    type: action_effect,
    description: Pressing button1 may turn on the
    kitchen1 light,
    content: add_effect(press, (when (= ?o button1)
     (light_on kitchen1))),
    condition: None,
    verification_condition: (At ?r kitchen1)
}
```

Collectively, the hypothesis set factorizes uncertainty over a single, unknown expanded world model. The `content` field specifies the structural model expansion introduced by a hypothesis, while the `verification_condition` defines the preconditions required to verify it. If the verification condition is `None`, the hypothesis is assumed to be true in our pipeline without requiring explicit verification.

C. Task-Oriented Hypothesis Generation

We formulate hypothesis generation as a structured generation problem in which a large language model (LLM) maps a natural language instruction and the robot's current knowledge of the environment into a hypothesis space. Let L denote the natural language instruction, and let $\mathcal{M}_0$ represent the initial symbolic world model. The objective is to identify a set of hypotheses $\mathcal{H}$ that bridges the gap between the goal state implied by L and the current scene graph $s_0 \in \mathcal{S}$.

Given an instruction L, we first parse it into a formal goal specification $\mathcal{G}$, represented as a conjunction of logical predicates. For example, the instruction *"Serve me a hot veggie burger at a desk"* is parsed as: `(exists (?o ?l) (and (burger ?o) (veggie ?o) (hot ?o) (desk ?l) (at ?o ?l)))`. By comparing the goal specification g with the current scene graph and domain knowledge in $\mathcal{M}_0$, the robot identifies missing objects $\mathcal{O}_{\text{miss}}$ and missing predicates $\mathcal{P}_{\text{miss}}$. To facilitate more structured reasoning, the LLM is prompted to further categorize the missing predicates into attribute predicates (e.g., visual or semantic properties) and effect predicates (e.g., state changes resulting from interactions), yielding $\mathcal{P}_{\text{miss}} \rightarrow \{\mathcal{P}_{\text{attr}}, \mathcal{P}_{\text{eff}}\}$. Hypothesis generation is then performed by an LLM-based generator Φ_{LLM}, which conditions on the current scene s, the goal condition G, the categorized missing predicates, and a historical context C_{hist}:

$$H = \Phi_{\text{LLM}}(s_0, g, \{\mathcal{P}_{\text{attr}}, \mathcal{P}_{\text{eff}}\}, C_{\text{hist}}, \mathcal{M}), \tag{1}$$

where $\mathcal{H}$ denotes a set of plausible hypotheses. To ensure consistency and avoid redundant exploration, the history context C_{hist} maintains a trace of execution, previously generated hypotheses, and their verification outcomes, thereby preventing cyclic generation of hypotheses that have already been refuted.

D. Planning Under Hypothesis Uncertainty

We model the open-world planning problem as a tuple $\Pi = \langle \mathcal{S}^+, \mathcal{P}^+, \mathcal{A}^+, \mathcal{T}^+, s_0, \mathcal{G} \rangle$, where uncertainty is explicitly represented through a set of hypotheses $\mathcal{H}$ generated by the hypothesis generator. Each hypothesis $h \in \mathcal{H}$ corresponds to a proposition whose truth value is not known a priori. For each hypothesis h, we introduce a belief variable $b(v(h)) \in \{\texttt{True}, \texttt{False}, \texttt{Unknown}\}$, indicating the estimated truth of the hypothesis. The augmented state space is defined as $\mathcal{S}^+ = \mathcal{S} \times \mathcal{B}$, where $\mathcal{B}$ denotes the space of belief assignments over all hypotheses. Initially, all hypotheses are unverified, i.e., $\forall h \in H,\ b(v(h)) = \texttt{Unknown}$. To resolve hypothesis uncertainty, we introduce a set of *verification actions* $\mathcal{A}_{\text{verify}} \subset \mathcal{A}^+$. For each hypothesis $h \in \mathcal{H}$, an abstract action a^h_{verify} is generated to observe its truth value through sensing or interaction. Executing a^h_{verify} yields a non-deterministic outcome, turning $b(v(h))$ to either $\texttt{True}$ or $\texttt{False}$.

To enable efficient planning with classical deterministic planners, we apply all-outcomes determinization [64] to verification actions, which allows the planner to reason over individual outcomes of an otherwise non-deterministic action. Specifically, each verification action a^h_{verify} is determinized into two actions, a^{h+}_{verify} and a^{h-}_{verify}, corresponding to the hypothesis being verified or refuted, respectively. In our setting, we further apply an explicit branch cut by excluding a^{h-}_{verify} from the planning graph. This is because falsified hypotheses cannot satisfy the current goal specification and therefore do not contribute to successful plans under the current hypothesis expansion. As a result, the planner is encouraged to adopt an optimistic strategy in the face of hypothesis uncertainty, while negative outcomes are handled at execution time through hypothesis rejection and replanning.

The PDDL definition of a verification action a^h_{verify} follows the template:

```
(:action verify_h
  :parameters (?r - Robot ?o - Obj ?l - Location)
  :precondition (and (At ?r ?l) (¬True ?h) (Related
    ?h ?o) φ_cond φ_verify)
  :effect (and (True ?h))
)
```

where ϕ_{cond} encodes dependencies between hypotheses and ϕ_{verify} specifies the sensing or interaction constraints required to verify h. To bias the planner toward resolving uncertainty early, we associate a high cost with actions that rely on unverified hypotheses. Specifically, for any action a involving an object associated with hypothesis h, an additional penalty cost $c \gg 0$ is incurred if $b(v(h)) = \texttt{Unknown}$. This design encourages the planner to schedule verification actions a^h_{verify} before manipulation actions that depend on h, effectively trading short-term verification cost for reduced execution risk due to incorrect hypotheses. The resulting deterministic planning problem is solved using the Fast Downward planner [22].

E. Hypothesis Verification and Model Update

During execution, standard actions are carried out by the robot's low-level skill library. When the plan prescribes a verification action $a \in \mathcal{A}_{\text{verify}}$, the system invokes a dedicated verification routine. Since the planning process explicitly reasons about the preconditions `verification_condition` required for hypothesis verification, such actions are executed only when the robot is already in a state suitable for reliable verification, as predicted by the language model. In practice, verification actions are grounded using either prebuilt perception modules or a VLM, depending on the type of hypothesis. In our implementation, hypotheses concerning object existence are verified by comparing the hypothesized objects against a newly observed scene graph, whose nodes are constructed by object detectors. For more fine-grained hypotheses, such as object attributes or action effects, a VLM is queried with the current image or the latest several images together with the hypothesis description to perform question answering. Based on the resulting sensory feedback, each hypothesis is either confirmed or refuted. If a hypothesis is verified, the corresponding fact is incorporated into the world model, provided that it does not conflict with existing knowledge. If the hypothesis is refuted, it is marked as false ($v_h \leftarrow$ False) and recorded in the history context C_{hist}. The hypothesis generator is then re-invoked to produce alternative hypotheses conditioned on the accumulated history and newly available observations, triggering a replanning cycle.

F. Necessary Assumptions

We summarize the key assumptions underlying our framework for extending automated planning with open-world knowledge from foundation models. These assumptions are consistent with prior work in automated planning and LLM-based planning, and are necessary for making the problem tractable and empirically feasible. Our framework relies on the following assumptions: (1) *Task Scoping*: We focus on uncertainty from underspecified symbolic world modeling and assume reliable execution of a library of atomic skills when under full observability. (2) *Goal Requirements*: Task goals must be expressible in structured logic, and define the abstraction of missing knowledge and be verifiable using the robot's sensing capabilities at some known configurations. In particular, missing object concepts and predicates are assumed

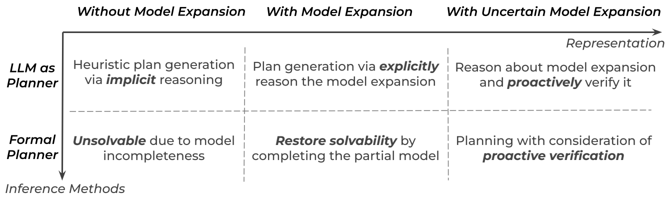

Fig. 4. Overview of the analysis of six planning categories, organized by representation and inference methods.

Fig. 5. Illustration of the Simulation Setup: Block Processing World (Left) and Mobile Manipulation in Unknown Environments (Right).

to be encoded in the task instructions. (3) *Language Model*: The language model is assumed to be able to generate correct hypotheses within a bounded number of attempts, possess the internal reasoning ability to propose verification conditions, and evaluate hypotheses from raw sensory observations.

V. Experiments and Results

To examine the usage and impact of uncertainty-aware model expansion in open-world planning, we evaluate HUME guided by the following research questions:

RQ1: *Does explicitly representing model expansion enable effective planning under incomplete knowledge?*

RQ2: *How does the planner's awareness of hypothesis uncertainty influence decision-making and exploration behavior?*

A. Approaches

We compare six approaches that differ along two principal axes: (1) ***Representation***, namely whether model expansion is explicit and whether uncertainty awareness is incorporated; and (2) ***Inference Method***, which distinguishes between formal symbolic planners and Large Language Models (LLMs). See Fig. 4 for an overview. We use `gpt-4.1-2025-04-14` as the language model across all modules and approaches.

1) *Formal Planner w/o Model Expansion:* A standard PDDL planner operating on the initial partial model.
2) *LLM as Planner w/o Model Expansion:* An LLM Planner that directly predicts action sequences from observations and instructions, without explicit world model expansion, representing a set of LLM-based methods [25, 34, 48, 49].
3) *Formal Planner with Model Expansion:* A PDDL planner augmented with hypotheses generation, treating all hypothesized knowledge as definite facts. This represents a set of language-augmented symbolic planners [6, 15].
4) *LLM as Planner with Model Expansion:* An LLM Planner uses the same hypothesis generation mechanism, but relies on an LLM to plan over the expanded knowledge base.
5) *Formal Planner with Uncertain Model Expansion:* A PDDL planner considers hypotheses as uncertain object-centric facts, enabling plans that both achieve task goals and actively verify assumptions.
6) *LLM as Planner with Uncertain Model Expansion:* An LLM Planner considers hypotheses as uncertain object-centric facts, enabling direct comparison with symbolic planning under same representation.

B. Simulation Scenarios

1) Block Processing World: As illustrated in Fig. 5 left, Block Processing World extends the classic Blocks World domain with multiple block processors, each producing a specific effect, such as cleaned, wet, or hot. The robot does not know each processor's effect a priori and needs to interact with processors to discover their effects. Similar to Blocks World, the goal is to build a target configuration while ensuring selected blocks exhibit designated effects. See the Appendix for more details.

2) Mobile Manipulation in Unknown Environments: In this setting (Fig. 5, right), a mobile manipulator operates in a household-like environment with objects of unknown locations and attributes. The robot must rearrange objects to achieve a target configuration, such as delivering an object with a specific attribute to a target location. Since object location, properties, and action effects are unknown, the robot must actively explore and interact with the environment to infer object attributes, uncover hidden objects, and predict action outcomes. We implement the benchmark in AI2-THOR [31] and evaluate eight tasks across five ProcTHOR houses [13], with three trials per house.

T1. *Store the remote control in a drawer.*
T2. *Take a fork to the dining table.*
T3. *Take a transparent bowl to the countertop.*
T4. *Move a damask-style pillow to a sofa.*
T5. *Carry a metal kettle to a fridge.*
T6. *Place a polished-gold vase on a desk.*
T7. *Deliver coffee in a mug with a logo to the dining table.*
T8. *Take a wet teal cloth to the countertop.*

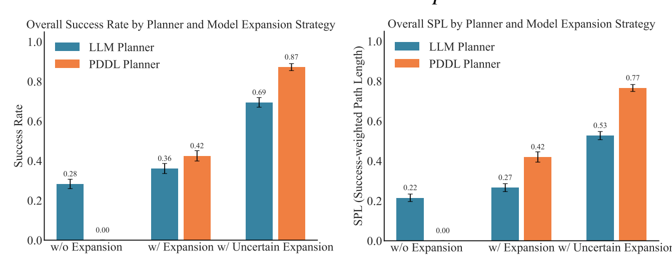

Fig. 6. Simulation Results on Block Processing World: Success Rate and Success weighted by Path Length (SPL).

C. Results and Analysis in Simulation

We evaluate the proposed approaches in both the Block Processing World and the AI2-THOR mobile manipulation tasks. As shown in Fig. 6, *explicitly representing and performing model expansion is critical for effective planning under incomplete world knowledge*. Both the LLM-based planner and the formal PDDL planner achieve substantial performance gains when hypotheses are introduced and maintained, while planners without model expansion consistently fail, demonstrating that static initial models are insufficient for open-world tasks. The formal planner benefits from explicit hypothesis representation which restores solvability, whereas the LLM planner

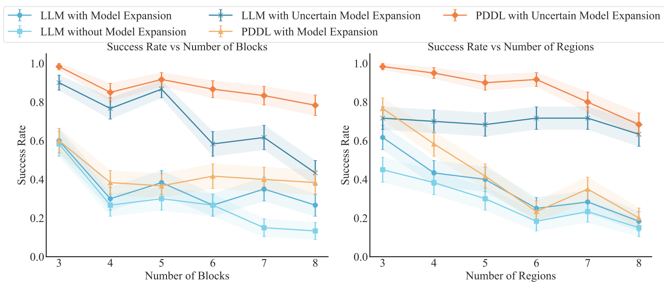

Fig. 7. Simulation Results on the Block Processing World: Performance Trends with Varying Numbers of Blocks and Regions.

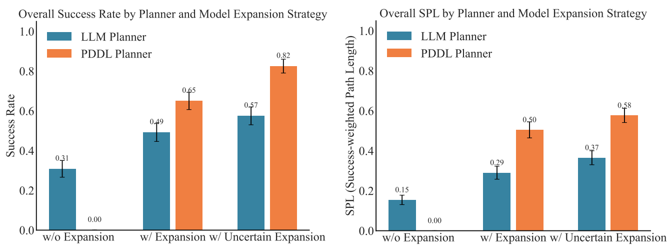

Fig. 8. Simulation Results on Mobile Manipulation in Unknown Environments: Success Rate and Success weighted by Path Length (SPL)

relies on heuristic plan generation via implicit reasoning, which is less reliable without explicit hypothesis maintenance. *Incorporating uncertainty awareness into the expanded model further improves performance across planners.* Uncertainty-aware model expansion consistently outperforms deterministic expansion in both success rate and SPL, highlighting the importance of reasoning about hypothesis validity rather than only generating hypotheses. Similar trends are observed in Fig. 7 as the number of regions increases, where uncertainty-aware expansion yields more robust performance under increasing task complexity. In Fig. 7, *as problem openness increases or task complexity decreases, explicit and uncertainty-aware model expansion becomes a stronger determinant of success than the specific choice of planner*. In these regimes, the performance gap between LLM-based and formal planners narrows, while proper handling of uncertain hypotheses remains the dominant factor affecting performance, suggesting a planner-agnostic advantage.

These trends extend to the AI2-THOR mobile manipulation benchmark (Fig. 8), where both planners benefit from model expansion and uncertainty-aware reasoning. However, the relative improvement of the LLM-based planner is smaller than in BlockWorld, likely due to the increased complexity arising from the large number of objects and locations in mobile manipulation environments, which makes verification and planning constraints harder for LLMs to enforce.

D. Real World Experiments

We validate HUME on a physical Fetch mobile manipulator [56] operating in a typical home environment comprising a kitchen, living room, and meeting room (Fig. 9). For perception, the robot maintains a 3D scene graph based on an existing localization map, while new objects are instantiated through a modular perception pipeline. This pipeline employs OWLv2 [41] for object detection and the Segment Anything

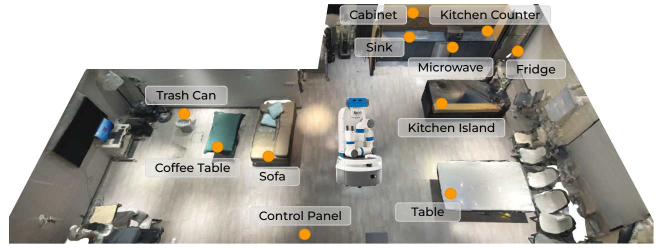

Fig. 9. The real-world experimental setup comprising a kitchen, living room, and dining room, with a Fetch mobile manipulator.

model [30] for mask extraction, with RGB-D observations fused for 3D back-projection. For low-level control, we implement a set of modular skills, including `navigate`, `pick`, `place`, and `trigger`, as well as learning-based skills like `open` and `close`. The learning-based skills are obtained via post-training of the $\pi_{0.5}$ model [26] on collected demonstration data. Further details of the real-world mobile manipulation system are provided in the Appendix.

The robot is evaluated on five tasks exhibiting substantial openness ($T1$–$T5$):

T1. *Deliver a zero-sugar drink to the table.*
T2. *Place the remote with a red button into the cabinet.*
T3. *Move the smiley-face mug to the fridge.*
T4. *Serve a heated chicken burger on the coffee table.*
T5. *Throw away the blue-floral bowl and turn off the living room light.*

To isolate planning performance from execution noise, we mitigate random execution failures by retrying conceptually feasible low-level skills until success. When execution failures result in invalid states (e.g., object drops), we manually reset the state based on the expected outcome. We evaluate all five planners on these five tasks, with three trials per task.

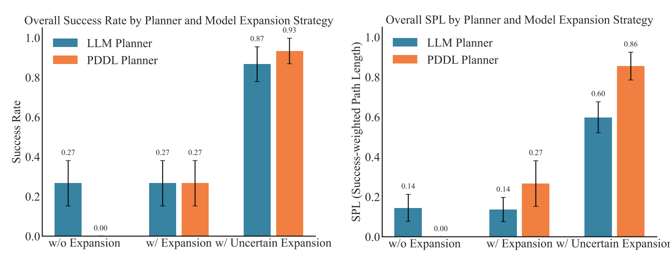

Fig. 10. Real-World Experiment Results: Success Rate and Success weighted by Path Length (SPL)

1) Result and Analysis: As demonstrated in Fig. 10, the real-world experimental results show that uncertainty-aware model expansion significantly improves task performance for both LLM-based and formal planners. Compared to simulation, the real-world tasks more closely resemble the harder simulation settings (T3–T8), as they exhibit greater openness in object attributes due to the presence of distracting objects and longer task horizons. See Appendix for additional analysis.

Fig. 11 and Fig. 12 illustrate the execution of two real-world long-horizon household tasks using the proposed hypothesis-driven PDDL planner. In Fig. 11 (*T1: Deliver a zero-sugar drink to the table*), the robot first hypothesizes that a drink is inside the refrigerator and that it is zero-sugar. It navigates to the refrigerator, opens it, observes candidate drinks, and

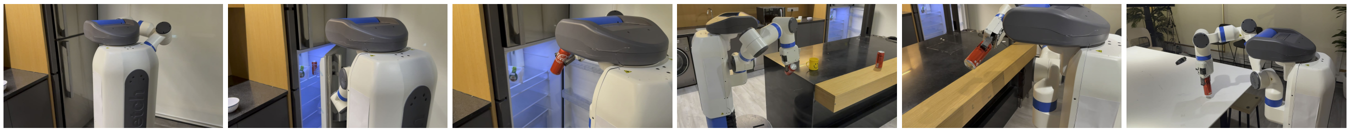

Fig. 11. Snapshot of completing the task **T1**: *Deliver a zero-sugar drink to the table.* The robot is required to infer the drink's location, identify the zero-sugar option, and deliver it to the table.

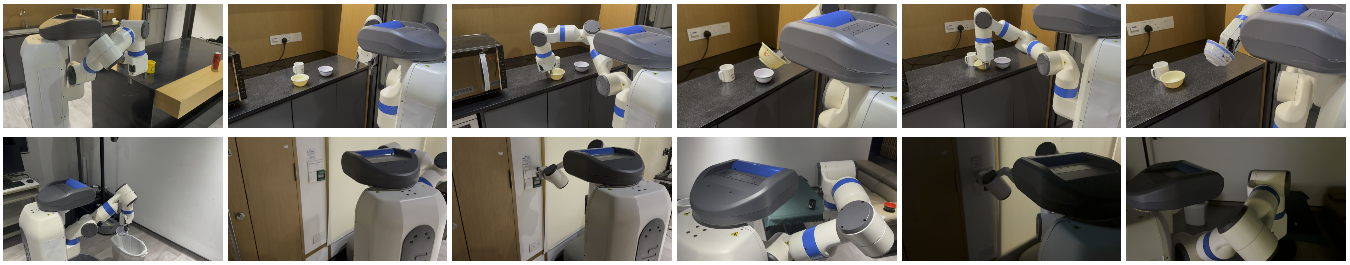

Fig. 12. Snapshot of completing the task **T5**: *Throw away a blue-floral bowl and make the living room light-off.* The robot is required to locate the bowl, identify the blue-floral pattern, and determine how to turn off the living room light.

verifies object existence. After picking up a drink, attribute verification rejects the initial hypothesis, triggering hypothesis regeneration and replanning. The robot then explores the kitchen island, where both the existence and attribute hypotheses are verified. It takes the correct drink and delivers it to the table. In Fig. 12 (*T5: Throw away a flower-patterned bowl and turn off the living room light*), the robot first hypothesizes the bowl's location and attribute, as well as a switch's location and its action effect. It searches the kitchen island and rejects the initial bowl-location hypothesis. After regenerating hypotheses and navigating to the kitchen counter, the robot observes two bowls, rejects the first through attribute verification, confirms the second, and discards it into the trash can. The robot moves to the panel area and successfully verifies the existence of switches. Based on its initial hypothesis, it presses the first switch, moves to the living room, and determines that the first switch does not control the living room light. It then updates its hypothesis, generates a new plan accordingly, and confirms that the second switch successfully turns off the living room light. These examples highlight the importance of maintaining and verifying hypotheses in real-world tasks, and demonstrate the intelligent behavior enabled by the proposed mechanism. Further analysis is provided in the Appendix and video.

E. Application in Appliance Operation

To demonstrate the generalizability of HUME to broader applications, we apply it to an appliance-operation task using the Autolife S2 humanoid robot. (see Appendix for the detailed setting). In this setting, the robot is required to operate a microwave to *defrost the food for 40 seconds*. For the PDDL solver, since the FastDownard doesn't support numeric planning, we use the numeric planner called ENHSP [44] with Unified-Planning library [40]. As shown in Fig. 13, the robot scans the buttons, matching known templates to corresponding PDDL operators, while the increase and decrease buttons lack predefined numerical effects. The robot first hypothesizes a 10-second time increase from pressing the increase button in time-defrost mode and plans and acts accordingly. Verification using the past two frames indicates that the time increases by

Fig. 13. Demonstration of a robot discovering microwave button functions through interaction.

5 seconds, prompting the robot to revise its hypothesis. The updated hypothesis is then used for subsequent planning and execution without requiring further verification, successfully setting the time to 40s. Further details are provided in the Appendix and video.

VI. Discussion

Our experiments show that hypothesis-driven uncertainty-aware model expansion (HUME) offers a principled way to connect structured symbolic planning with the open-world knowledge of LLMs. Classical planning systems provide logical and geometric rigor but are brittle under incomplete models, while LLMs are flexible but unreliable as standalone planners. Our key insight is to use LLMs not to replace formal planning, but to propose uncertain model expansions whose validity is tested through planning and execution, allowing the planning system to manage uncertainty over preconditions, dependencies, and feasibility.

Despite its effectiveness, the proposed framework makes several simplifications that limit its generality and highlight important open challenges.

a) Choice of uncertainty representation and planning under uncertainty. The current system represents generated hypotheses as ternary beliefs: true, false, or unknown. Verification actions are treated optimistically during planning, with negative outcomes handled through replanning. This design supports tractable, real-time planning, but limits the system's ability to reason about graded confidence or risk. We adopt it because foundation models do not yet provide reliably calibrated probabilities over generated knowledge, and richer uncertainty-aware planning remains computationally challenging in long-horizon settings. Consequently, the system may behave overly optimistically when risk-sensitive decisions are required. In future work, we aim to integrate uncertainty calibration tools [38] into the model expansion process and explore more advanced online POMDP solvers [23, 28] for risk-aware planning under uncertainty.

b) Hypothesis verification conditions. We assume that language models can propose verification conditions sufficient to determine a hypothesis and that these conditions can be enforced by a formal planner. However, some real-world verification conditions depend on geometry, long-horizon interactions, or complex perception, which may be difficult to infer from language alone. In such complex cases, deciding how to verify a hypothesis itself becomes an active perception or planning problem. Future work will extend the current approach by integrating language-predicted verification conditions with geometry- and perception-aware reasoning, making hypothesis verification more suitable for integrated task and motion planning (TAMP) settings [21].

c) Source of abstraction. Our current implementation requires the task instruction to define the abstraction of missing knowledge. This means that missing object concepts and predicates are explicitly encoded in the task goals. We make this assumption to ensure methodological completeness and to keep hypothesis generation task-oriented. In future work, we aim to explore failure reasoning modules [39] as an additional source of abstraction, using execution feedback to identify missing concepts and formulate new hypotheses that are not directly specified by the task goals.

d) Open-ended exploration in open-world settings. We avoid allowing the LLM to terminate hypothesis generation when it deems a task impossible, in order to prevent overly pessimistic early termination. Our task scope assumes that the task is achievable in the underlying environment and focuses on optimistically uncovering missing knowledge. In practice, termination is bounded by the maximum number of hypothesis-generation attempts and replanning steps. More broadly, proving task impossibility in open-world settings is generally intractable, a challenge shared across the open-world planning literature. Extending the framework to incorporate impossibility reasoning, such as belief-space pruning or contradiction detection, is an interesting direction for future work.

VII. Conclusion

We presented a hypothesis-driven framework for open-world robot planning that treats model expansion as an explicit and uncertain process integrated into goal-directed planning. By actively generating and verifying hypotheses, the proposed approach enables robots to operate effectively under incomplete and evolving world models. Experiments in simulation and the real world show that uncertainty-aware model expansion significantly improves robustness and success for both symbolic and LLM-based planners. Our results highlight the importance of integrating uncertainty-aware model expansion into planning systems for deploying service robots in real open-world environments, while more fine-grained uncertainty estimation and tighter integration with advanced uncertainty-aware planning algorithms remain important directions for future work. Additional discussion of insights, limitations, and future work is provided in the appendix.

Acknowledgments

This research is supported in part by Singapore Ministry of Digital Information under its AIVP program (grant no. A-8004133-00-00). Anxing Xiao is supported by the Research Scholarship from the National University of Singapore. We thank Chao Tang, Jian Zhang, Siwei Chen, and Panpan Cai for helpful discussions and suggestions on an earlier version of this work.

References

[1] Ashay Athalye, Nishanth Kumar, Tom Silver, Yichao Liang, Jiuguang Wang, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. From pixels to predicates: Learning symbolic world models via pretrained vlms. *IEEE Robotics and Automation Letters*, 2026.

[2] Alper Aydemir, Andrzej Pronobis, Moritz Göbelbecker, and Patric Jensfelt. Active visual object search in unknown environments using uncertain semantics. *IEEE Transactions on Robotics*, 29(4):986–1002, 2013.

[3] Timo Birr, Christoph Pohl, Abdelrahman Younes, and Tamim Asfour. Autogpt+ p: Affordance-based task planning with large language models. In *Robotics: Science and Systems (RSS)*, 2024.

[4] Gerhard Brewka, Thomas Eiter, and Mirosław Truszczyński. Answer set programming at a glance. *Communications of the ACM*, 54(12):92–103, 2011.

[5] Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, et al. Do as i can, not as i say: Grounding language in robotic affordances. In *Conference on robot learning*, pages 287–318. PMLR, 2023.

[6] Guanqi Chen, Lei Yang, Ruixing Jia, Zhe Hu, Yizhou Chen, Wei Zhang, Wenping Wang, and Jia Pan. Language-augmented symbolic planner for open-world task planning. In *Robotics: Science and Systems (RSS)*, 2024.

[7] Sonia Chernova, Vivian Chu, Angel Daruna, Haley Garrison, Meera Hahn, Priyanka Khante, Weiyu Liu, and Andrea Thomaz. Situated bayesian reasoning framework for robots operating in diverse everyday environments. In *Robotics Research: The 18th International Symposium ISRR*, pages 353–369. Springer, 2020.

[8] Stephen Cresswell and Peter Gregory. Generalised domain model acquisition from action traces. In *Pro-*

ceedings of the international conference on automated planning and scheduling, volume 21, pages 42–49, 2011.
[9] Aidan Curtis, Nishanth Kumar, Jing Cao, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Trust the proc3s: Solving long-horizon robotics problems with llms and constraint satisfaction. In *Conference on Robot Learning*, 2024.
[10] Aidan Curtis, George Matheos, Nishad Gothoskar, Vikash Mansinghka, Joshua Tenenbaum, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Partially observable task and motion planning with uncertainty and risk awareness. In *Robotics: Science and Systems (RSS)*, 2024.
[11] Aidan Curtis, Hao Tang, Thiago Veloso, Kevin Ellis, Joshua B Tenenbaum, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Llm-guided probabilistic program induction for pomdp model estimation. In *Conference on Robot Learning*, pages 3137–3184. PMLR, 2025.
[12] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–10, 2018.
[13] Matt Deitke, Eli Vander Bilt, Alvaro Herrasti, Luca Weihs, Jordi Salvador, Kiana Ehsani, Winson Han, Eric Kolve, Ali Farhadi, Aniruddha Kembhavi, et al. Procthor: large-scale embodied ai using procedural generation. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pages 5982–5994, 2022.
[14] Maximilian Diehl, Chris Paxton, and Karinne Ramirez-Amaro. Automated generation of robotic planning domains from observations. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6732–6738. IEEE, 2021.
[15] Yan Ding, Xiaohan Zhang, Saeid Amiri, Nieqing Cao, Hao Yang, Andy Kaminski, Chad Esselink, and Shiqi Zhang. Integrating action knowledge and llms for task planning and situation handling in open worlds. *Autonomous Robots*, 47(8):981–997, 2023.
[16] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.
[17] Michael O'Gordon Duff. *Optimal Learning: Computational procedures for Bayes-adaptive Markov decision processes*. University of Massachusetts Amherst, 2002.
[18] Maria Fox and Derek Long. Pddl2. 1: An extension to pddl for expressing temporal planning domains. *Journal of artificial intelligence research*, 20:61–124, 2003.
[19] Caelan Reed Garrett, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Pddlstream: Integrating symbolic planners and blackbox samplers via optimistic adaptive planning. In *Proceedings of the international conference on automated planning and scheduling*, volume 30, pages 440–448, 2020.
[20] Caelan Reed Garrett, Chris Paxton, Tomás Lozano-Pérez, Leslie Pack Kaelbling, and Dieter Fox. Online replanning in belief space for partially observable task and motion problems. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5678–5684. IEEE, 2020.
[21] Caelan Reed Garrett, Rohan Chitnis, Rachel Holladay, Beomjoon Kim, Tom Silver, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Integrated task and motion planning. *Annual review of control, robotics, and autonomous systems*, 4(1):265–293, 2021.
[22] Malte Helmert. The fast downward planning system. *Journal of Artificial Intelligence Research*, 26:191–246, 2006.
[23] Marcus Hoerger, Muhammad Sudrajat, and Hanna Kurniawati. Vectorized online pomdp planning. *arXiv preprint arXiv:2510.27191*, 2025.
[24] Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International conference on machine learning*, pages 9118–9147. PMLR, 2022.
[25] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner monologue: Embodied reasoning through planning with language models. In *Conference on Robot Learning*, pages 1769–1782. PMLR, 2023.
[26] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. π0. 5: a vision-language-action model with open-world generalization, 2025. *URL https://arxiv.org/abs/2504.16054*, 1(2):3.
[27] Yuqian Jiang, Nick Walker, Justin Hart, and Peter Stone. Open-world reasoning for service robots. In *Proceedings of the international conference on automated planning and scheduling*, volume 29, pages 725–733, 2019.
[28] Xuanjin Jin, Yanxin Dong, Bin Sun, Huan Xu, Zhihui Hao, XianPeng Lang, and Panpan Cai. Vec-qmdp: Vectorized pomdp planning on cpus for real-time autonomous driving. *arXiv preprint arXiv:2602.08334*, 2026.
[29] Leslie Pack Kaelbling and Tomás Lozano-Pérez. Integrated task and motion planning in belief space. *The International Journal of Robotics Research*, 32(9-10):1194–1227, 2013.
[30] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.
[31] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, et al. Ai2-thor: An

interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017.
[32] Nishanth Kumar, William Shen, Fabio Ramos, Dieter Fox, Tomás Lozano-Pérez, Leslie Pack Kaelbling, and Caelan Reed Garrett. Open-world task and motion planning via vision-language model generated constraints. *IEEE Robotics and Automation Letters*, 2026.
[33] Minseo Kwon and Young J Kim. Kinodynamic task and motion planning using vlm-guided and interleaved sampling. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026.
[34] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9493–9500. IEEE, 2023.
[35] Yichao Liang, Nishanth Kumar, Hao Tang, Adrian Weller, Joshua B Tenenbaum, Tom Silver, João F Henriques, and Kevin Ellis. Visualpredicator: Learning abstract world models with neuro-symbolic predicates for robot planning. In *International Conference on Learning Representations (ICLR)*, 2025.
[36] Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. Llm+ p: Empowering large language models with optimal planning proficiency. *arXiv preprint arXiv:2304.11477*, 2023.
[37] Weiyu Liu, Neil Nie, Ruohan Zhang, Jiayuan Mao, and Jiajun Wu. Learning compositional behaviors from demonstration and language. In *8th Annual Conference on Robot Learning*, 2024.
[38] Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. Uncertainty quantification and confidence calibration in large language models: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 6107–6117, 2025.
[39] Zeyi Liu, Arpit Bahety, and Shuran Song. Reflect: Summarizing robot experiences for failure explanation and correction. In *Conference on Robot Learning*, pages 3468–3484. PMLR, 2023.
[40] Andrea Micheli, Arthur Bit-Monnot, Gabriele Röger, Enrico Scala, Alessandro Valentini, Luca Framba, Alberto Rovetta, Alessandro Trapasso, Luigi Bonassi, Alfonso Emilio Gerevini, Luca Iocchi, Felix Ingrand, Uwe Köckemann, Fabio Patrizi, Alessandro Saetti, Ivan Serina, and Sebastian Stock. Unified planning: Modeling, manipulating and solving ai planning problems in python. *SoftwareX*, 29:102012, 2025. ISSN 2352-7110. doi: https://doi.org/10.1016/j.softx.2024.102012. URL https://www.sciencedirect.com/science/article/pii/S2352711024003820.
[41] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems*, 36, 2024.
[42] Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. Sayplan: Grounding large language models using 3d scene graphs for scalable task planning. *arXiv preprint arXiv:2307.06135*, 2023.
[43] Allen Z Ren, Jaden Clark, Anushri Dixit, Masha Itkina, Anirudha Majumdar, and Dorsa Sadigh. Explore until confident: Efficient exploration for embodied question answering. In *Robotics: Science and Systems (RSS)*, 2024.
[44] Enrico Scala, Patrik Haslum, and Sylvie Thiébaux. Heuristics for numeric planning via subgoaling. 2016.
[45] Noah Shinn, Beck Labash, and Ashwin Gopinath. Reflexion: an autonomous agent with dynamic memory and self-reflection. *arXiv preprint arXiv:2303.11366*, 2023.
[46] Tom Silver, Varun Hariprasad, Reece S Shuttleworth, Nishanth Kumar, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Pddl planning with pretrained large language models. In *NeurIPS 2022 Foundation Models for Decision Making Workshop*, 2022.
[47] Tom Silver, Soham Dan, Kavitha Srinivas, Joshua B Tenenbaum, Leslie Pack Kaelbling, and Michael Katz. Generalized planning in pddl domains with pretrained large language models. *arXiv preprint arXiv:2305.11014*, 2023.
[48] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. Progprompt: Generating situated robot task plans using large language models. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11523–11530. IEEE, 2023.
[49] Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M. Sadler, Wei-Lun Chao, and Yu Su. Llm-planner: Few-shot grounded planning for embodied agents with large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023.
[50] Lingfeng Sun, Devesh K Jha, Chiori Hori, Siddarth Jain, Radu Corcodel, Xinghao Zhu, Masayoshi Tomizuka, and Diego Romeres. Interactive planning using large language models for partially observable robotic tasks. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14054–14061. IEEE, 2024.
[51] Wenjing Tang, Xinyu He, Yongxi Huang, Yunxiao Xiao, Cewu Lu, and Panpan Cai. Tru-pomdp: Task planning under uncertainty via tree of hypotheses and open-ended pomdps. In *Advances in Neural Information Processing Systems*, 2025.
[52] Sai Vemprala, Rogerio Bonatti, Arthur Bucker, and Ashish Kapoor. Chatgpt for robotics: Design principles and model abilities. *Microsoft Auton. Syst. Robot. Res*, 2:20, 2023.
[53] Arthur Wandzel, Yoonseon Oh, Michael Fishman, Nishanth Kumar, Lawson LS Wong, and Stefanie Tellex. Multi-object search using object-oriented pomdps. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7194–7200. IEEE, 2019.

[54] Zihao Wang, Shaofei Cai, Guanzhou Chen, Anji Liu, Xiaojian Ma, Yitao Liang, and Team CraftJarvis. Describe, explain, plan and select: interactive planning with large language models enables open-world multi-task agents. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 34153–34189, 2023.

[55] Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinbing Hou, Bowei Zhang, Haowei Lin, Zhaofeng He, Zilong Zheng, Yaodong Yang, et al. Jarvis-1: Open-world multi-task agents with memory-augmented multimodal language models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[56] Melonee Wise, Michael Ferguson, Derek King, Eric Diehr, and David Dymesich. Fetch and freight: Standard platforms for service robot applications. In *Workshop on autonomous mobile service robots*, volume 5, 2016.

[57] Lawson LS Wong, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Manipulation-based active search for occluded objects. In *2013 IEEE International Conference on Robotics and Automation*, pages 2814–2819. IEEE, 2013.

[58] Lionel Wong, Jiayuan Mao, Pratyusha Sharma, Zachary S Siegel, Jiahai Feng, Noa Korneev, Joshua B Tenenbaum, and Jacob Andreas. Learning adaptive planning representations with natural language guidance. *arXiv preprint arXiv:2312.08566*, 2023.

[59] Yaqi Xie, Chen Yu, Tongyao Zhu, Jinbin Bai, Ze Gong, and Harold Soh. Translating natural language to planning goals with large-language models. *arXiv preprint arXiv:2302.05128*, 2023.

[60] Zhutian Yang, Caelan Garrett, Dieter Fox, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Guiding long-horizon task and motion planning with vision language models. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 16847–16853. IEEE, 2025.

[61] Ziyi Yang, Benned Hedegaard, Ahmed Jaafar, Yichen Wei, Skye Thompson, Shreyas S Raman, Haotian Fu, Stefanie Tellex, George Konidaris, David Paulius, et al. Skillwrapper: Generative predicate invention for skill abstraction. *arXiv preprint arXiv:2511.18203*, 2025.

[62] Haoming Ye, Yunxiao Xiao, Cewu Lu, and Panpan Cai. Unidomain: Pretraining a unified pddl domain from real-world demonstrations for generalizable robot task planning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

[63] Haoming Ye, Yunxiao Xiao, Cewu Lu, and Panpan Cai. Uniplan: Vision-language task planning for mobile manipulation with unified pddl formulation. *arXiv preprint arXiv:2602.08537*, 2026.

[64] Sung Wook Yoon, Alan Fern, and Robert Givan. Ff-replan: A baseline for probabilistic planning. In *ICAPS*, volume 7, pages 352–359, 2007.

[65] Ben Zandonati, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Rational inverse reasoning. *arXiv preprint arXiv:2508.08983*, 2025.

[66] Linfeng Zhao, Willie McClinton, Aidan Curtis, Nishanth Kumar, Tom Silver, Leslie Pack Kaelbling, and Lawson LS Wong. Seeing is believing: Belief-space planning with foundation models as uncertainty estimators. *arXiv preprint arXiv:2504.03245*, 2025.

[67] Zirui Zhao, Wee Sun Lee, and David Hsu. Large language models as commonsense knowledge for large-scale task planning. *Advances in Neural Information Processing Systems*, 36, 2024.

[68] Wang Bill Zhu, Ishika Singh, Robin Jia, and Jesse Thomason. Language models can infer action semantics for symbolic planners from environment feedback. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8751–8773, 2025.