Position: Toward a Theory of Agents as Tool-Use Decision-Makers

Anonymous Author(s)
Affiliation
Address
email

Abstract

As Large Language Models (LLMs) evolve into increasingly autonomous agents, fundamental questions about their epistemic foundations remain unresolved: What defines an agent? How should it make decisions? And what objectives should guide its behavior? In this position paper, we argue that true autonomy requires agents to be grounded in a coherent epistemic framework that governs what they know, what they need to know, and how to acquire that knowledge efficiently. We propose a unified theory that treats internal reasoning and external actions as equivalent epistemic tools, enabling agents to systematically coordinate introspection and interaction. Building on this framework, we advocate for aligning an agent's tool use decision-making boundary with its knowledge boundary, thereby minimizing unnecessary tool use and maximizing epistemic efficiency. This perspective shifts the design of agents from mere action executors to knowledge-driven intelligence systems, offering a principled path toward building foundation agents capable of adaptive, efficient, and goal-directed behavior.

1 Introduction

Large Language Models (LLMs) have rapidly evolved beyond text generation into autonomous agents capable of independently planning and executing complex tasks with minimal human oversight [1]. These emerging capabilities have enabled a broad range of real-world applications, including travel planning [2], human-computer interaction [3–5], and scientific research [6–9]. However, as these systems grow increasingly agentic, foundational questions remain unresolved: What is an agent? What constitutes its optimal behavior? And how can such optimality be realized in practice?

From a conceptual standpoint, the dominant view frames agents as LLMs that interleave internal reasoning and external actions to complete tasks. While functionally effective, this pragmatic framing lacks a principled account of how such behaviors should be coordinated or optimized. From an empirical standpoint, existing agentic systems primarily rely on prompting [10, 11] or supervised fine-tuning [12, 13], but seldom investigate how these training paradigms relate to the optimality of agent behavior, leaving opaque the reasons behind agentic success or failure. To address this theoretical and empirical gap, we propose a bottom-up formalization of agency grounded in four key constructs: what is a tool, what is an agent, what constitutes optimal behavior, and how to achieve it. Specifically, we posit that: (1) A tool is any process or interface, whether internal reasoning or external interaction, that contributes to knowledge acquisition towards goal completion. (2) An agent is a decision-maker that dynamically coordinates internal and external tools in pursuit of specific objectives. (3) Optimal behavior occurs when an agent aligns its tool use decisions with its knowledge boundary. (4) This alignment can be operationalized by minimizing the agent's unnecessary tool use. In summary, we argue **an optimal agent is one that adaptively coordinates internal reasoning and external action to acquire only the knowledge it needs, achieving goals efficiently by minimizing unnecessary tool use.**

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

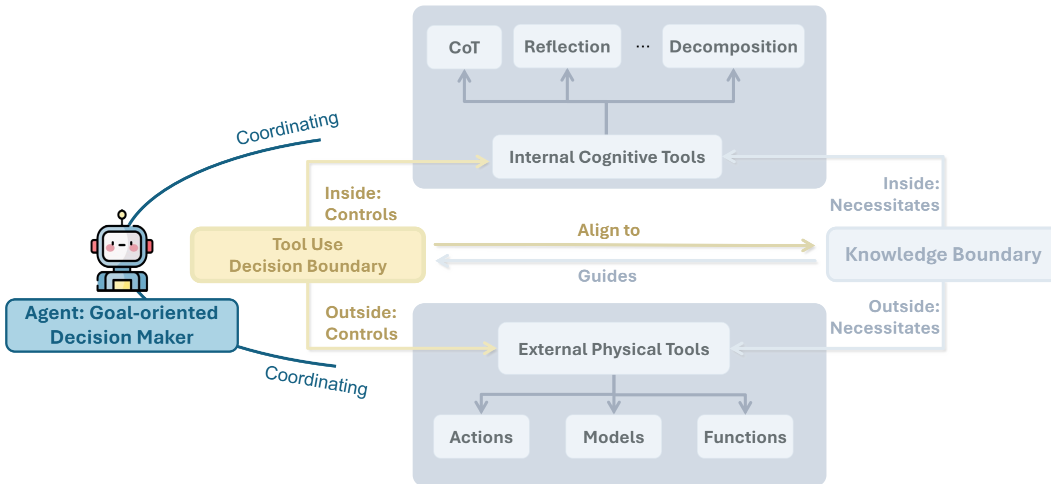


Figure 1: Conceptual framework of agent decision-making based on tool use and knowledge boundaries. The agent is modeled as a goal-directed decision-maker that coordinates internal cognitive tools (e.g., chain-of-thought, reflection) and external physical tools (e.g., actions, models, functions) across a tool use decision boundary. Optimal behavior emerges when tool use decisions align with the knowledge boundary, ensuring the agent invokes only the tools necessary to acquire missing knowledge and efficiently achieve task objectives.

To arrive at these positions, we systematically introduce the *Theory of Agent*, inspired by the cognitive concept of *Theory of Mind*, which characterizes an agent's capacity to model not only external environments but also its own internal knowledge state. This theory is grounded in the core insight that reasoning and acting are not distinct behaviors but rather *epistemically equivalent tools* for acquiring task-relevant knowledge. This allows us to conceptualize agents as *knowledge-driven tool-use decision-makers* that adaptively choose between internal introspection and external interaction (As shown in Figure 1). Our theoretical framework is developed in two stages: (1) a unified behavioral framework (Section 2) that models reasoning and acting under a shared decision-making logic, treating them as internal and external tools for retrieving different sources of knowledge. This unification enables us to frame all agent interactions as tool use decisions; (2) three core principles of knowledge (Section 3) that define the structure and dynamics of an agent's epistemic state and decision-making process, providing theoretical lemmata for what constitutes optimal behavior.

Building on this theoretical foundation, we first articulate the importance of aligning an agent's knowledge boundary with its decision boundary, and explore how this alignment is manifested through training and inference (Section 4.1). Furthermore, we identify four distinct behavior modes of agents, and argue that an autonomous agent should learn to accomplish its predefined goals with the minimal number of external tool calls (Section 4.2): an objective that aligns closely with expert's vision for autonomous machine intelligence [14], where "a truly autonomous machine intelligence is designed to minimize the number of actions a system needs to take in the real world to learn a task." In our framework, the world model is embodied by the LLM itself, as proposed in recent studies [15, 16]. Finally, we outline a general roadmap toward building foundation agents that realize these properties and principles in practice (Section 4.3).

2 Foundations

2.1 The Unification of Reasoning and Acting

It is widely recognized that reasoning and acting constitute the two fundamental capabilities of intelligent agent behavior [10, 17]. Reasoning enables an agent to plan, infer, reflect, and monitor its internal cognitive state, while acting allows it to engage with the external environment to gather new information or carry out tasks. Rather than viewing these modalities as distinct or sequential processes, we propose that:

Position: A Unified View of Reasoning and Acting

Reasoning and acting should be treated as equivalent epistemic tools within a unified framework, where reasoning entails an internal cognitive tool for manipulating information within the agent's parametric knowledge space, while acting entails an external physical tool for acquiring information beyond the agent's internal capabilities.

This unified perspective aligns with the affordance theory [18], which suggests that actions arise from the interplay between perception and interaction. From this perspective, reasoning and acting are not hierarchically ordered or merely sequential but are co-equal capabilities of decision-making. Each plays a complementary role in enabling agents to resolve uncertainty and make progress toward task completion. Embracing this integrated view encourages the development of intelligent systems that can seamlessly coordinate their internal cognitive mechanisms and external interactive capabilities, based on their current knowledge state and the epistemic demands of the task at hand.

Internal cognitive tools. Cognitive tools refer to internal cognitive mechanisms that support systematic or investigative thinking to solve problems [19, 20]. In the context of intelligent agents, various reasoning modules [11, 21], such as Chain-of-Thought [22], reflection, decomposition, and alternative-thinking, function as cognitive processes that enable the retrieval and manipulation of internal knowledge to guide problem-solving. For instance, Reasoning via Planning (RAP) [15] conceptualizes the language model as both a world model and a reasoning engine, incrementally accumulating knowledge through iterative reasoning steps. Similarly, Self-Discover [11] constructs abstract reasoning structures and then instantiates them to address complex tasks, mirroring the approach of tool-based agents that first generate plans for tool use and then execute them sequentially [23, 24]. Beyond these, other cognitive tools appear in diverse applications, such as conversational strategies in dialogue systems [25] and psychologically inspired mechanisms designed to model uncertainty, emotion, or user intent [26]. Despite their varied forms, these tools share a common function: *they serve as triggers for internal knowledge retrieval, allowing the model to reason and act based on its embedded understanding of the world.*

External physical tools. External physical tools refer to modules or interfaces outside the model that are invoked through specific triggers, such as rules, actions, or special tokens, whose outputs are then incorporated into the model's context to inform subsequent reasoning [27, 24]. These tools function as vital interfaces between the agent and its environment, enabling the acquisition of task-relevant knowledge that lies beyond the agent's internal parameters. Importantly, external tools span a wide spectrum of interactions, capturing how agents, like humans, leverage their surroundings to reduce uncertainty or complete tasks. Examples include querying a search engine, calling an API, processing sensor input, or performing physical actions [27, 28]. For instance, clicking a button in a user interface may be represented as an external tool call, where the input parameter is the button's location and the resulting webpage serves as the observation. Similarly, in embodied settings, actions such as "MoveTo(Room A)" can be interpreted as tool invocations, with "Room A" as the parameter and the resulting sensory output as the feedback. This perspective enables a unified treatment of diverse forms of interaction as structured tool use: *they serve as interfaces for external knowledge acquisition, allowing the model to access and interact with knowledge beyond its epistemic capacity.*

2.2 Tool-Integrated Agents

Building on the unification of reasoning and acting, we further propose a redefinition of the agent grounded in this integrated perspective:

Position: Definition of Agent

An agent is an entity that coordinates internal cognitive tools (e.g., reflection) and external physical tools (e.g., function callings) to acquire knowledge in order to achieve a specific goal.

From this viewpoint, an agent is fundamentally a knowledge-driven decision-maker that navigates a task by alternating between internal reasoning and external interaction. Formally, a tool-integrated agent trajectory can be described as $\tau = (t_1, k_1, t_2, k_2, ..., t_n, k_n)$, where each t_i is either an internal or external tool invocation, and each k_i represents the corresponding knowledge retrieved. Here, "knowledge" is broadly defined as *any information that advances the agent's problem-solving state*. At each step, the agent must choose the most epistemically valuable tool based on its current state, aiming to progressively bridge the knowledge gap toward a complete solution. The process concludes when the agent has accumulated sufficient knowledge to achieve the pre-defined goal.

This unified framework offers several key advantages: (1) It generalizes prior approaches such as ReAct [10], which can be viewed as special cases where internal tool steps (e.g., reasoning) are treated as monolithic thought units r_i, leveraging the model's pre-trained cognitive abilities without requiring explicit tool separation. (2) It aligns with findings from large reasoning models (LRMs), which show that outcome-based reinforcement learning (RL) can effectively train agents to discover and utilize internal cognitive tools [29]. The same principle applies to external physical tools, as shown in recent studies on tool-augmented agents [30]. Thus, the framework provides a coherent foundation for agentic learning across both domains. (3) Most importantly, this perspective leads to a new learning paradigm: *next tool prediction*. Just as next-token prediction enables LLMs to learn a compressed representation of the world from text, next-tool prediction allows agents to learn procedural knowledge through interaction. By learning to choose the right tool, agents can dynamically update their internal representations and evolve through experience, mimicking human-like adaptation and learning.

3 Principles of Knowledge and Decision in Model

As established earlier, an agent functions as a knowledge-driven decision-maker regarding the use of internal or external tools. This implies the existence of a knowledge space defined by the agent's own knowledge boundary, which informs its decisions, and a corresponding decision boundary that determines whether internal or external tools are employed. Understanding these boundaries is crucial for analyzing agent behavior and serves as a basis for optimizing it. In this section, we formalize the concepts of knowledge and decision boundaries (Section 3.1), and, based on their definitions, we introduce three key principles that highlight optimal agent behavior (Section 3.2 to Section 3.4).

3.1 The Definition of Knowledge and Decision Boundaries

Knowledge Boundary. At any time step t, let $\mathcal{W}$ represent the complete set of world knowledge. We define the model m's internal and external knowledge as:

$$\mathcal{K}_{\text{int}}(m,t) \subseteq \mathcal{W} \quad \text{and} \quad \mathcal{K}_{\text{ext}}(m,t) = \mathcal{W} \setminus \mathcal{K}_{\text{int}}(m,t)$$

where $\mathcal{K}_{\text{int}}(m,t)$ denotes the internal knowledge embedded in m, and $\mathcal{K}_{\text{ext}}(m,t)$ represents the external knowledge accessible from the world. The *knowledge boundary* is defined as the frontier between the two:

$$\partial\mathcal{K}(m,t) = \partial\mathcal{K}_{\text{int}}(m,t) = \partial\mathcal{K}_{\text{ext}}(m,t)$$

This boundary marks the epistemic limit of the model's internal knowledge. We assume all internal or external knowledge is accurate, leaving discussion of overlap or conflict to Appendix B.

Decision Boundary. Given a time step t, let $\mathcal{T}_{int} = \{t_{int}^1, ..., t_{int}^n\}$ be the set of internal cognitive tools and $\mathcal{T}_{ext} = \{t_{ext}^1, ..., t_{ext}^m\}$ the set of external physical tools. The *decision boundary* $\partial\mathcal{D}(m,t)$ is the point at which the model decides whether to use internal or external tools to acquire additional task-relevant knowledge:

$$\partial\mathcal{D}(m,t) = \partial\mathcal{T}_{\text{int}}(m,t) = \partial\mathcal{T}_{\text{ext}}(m,t)$$

where $\mathcal{T}_{\text{int}}(m,t)$ and $\mathcal{T}_{\text{ext}}(m,t)$ denote tool choices leading to internal or external knowledge acquisition, respectively.

In summary, the knowledge boundary defines the model's epistemic limits, while the decision boundary governs how the model navigates these limits through tool use. Each point in the knowledge space corresponds to a point in the decision space, reflecting how the model chooses to engage with that knowledge through tool use. In this way, the decision boundary operationalizes the knowledge boundary, shaping the model's policy for knowledge acquisition in pursuit of its goals.

3.2 Principle 1: Foundations

Most existing models are pre-trained on large-scale corpora in an unsupervised manner, embedding substantial world knowledge into their parameters [31–33]. However, not all knowledge is internalized during pre-training, and external knowledge may still need to be acquired through continual learning strategies such as SFT or RL [34, 35]. We introduce two key lemmas that ground the distinction and dynamics of internal and external knowledge, forming the basis of the knowledge boundary.

Lemma 1.1: *Over long horizons, scaling laws enable the expansion of $\mathcal{K}_{\text{int}}$; i.e., the knowledge boundary $\partial\mathcal{K}$ moves outward.* This expansion reflects the model's increasingly comprehensive internal representation of the world across modalities and domains. For instance, Sora[1] demonstrates the acquisition of rich physical knowledge, enabling the generation of realistic, coherent long-form videos. With sufficient training data, architecture, and optimization, the model effectively compresses the external world into its internal parameter space [36, 37]. As $\partial\mathcal{K}$ expands with scale, the model may ultimately support real-time abstraction of the world, or even autonomously discover knowledge beyond existing human understanding [38–40], leading toward AI for scientific discovery.

Lemma 1.2: *Continual learning methods such as SFT can reshape both the knowledge boundary and the decision boundary.* To stay current and improve performance, models must update outdated knowledge or acquire new information through continual learning, including prompting [41], supervised fine-tuning [42, 43], and knowledge editing [44, 45]. These processes naturally shift the knowledge boundary to reflect updated internal states. In parallel, decision boundaries can be adjusted to improve tool use behavior, such as encouraging external tool invocation only when necessary [46, 47]. Central to this is the model's meta-cognitive ability to assess its current knowledge and decide which tool to use accordingly, which we will discuss in Section 4.1.

3.3 Principle 2: Uniqueness and Diversity

Across open-source and proprietary models, both unique and shared characteristics emerge. To better understand model capabilities and limitations, we posit that while each model has distinct boundaries, there also exist universal properties common to all.

Lemma 2.1: *Each model has its own knowledge boundary and decision boundary.* These boundaries differ due to variations in model size, architecture, training data, and learning objectives. Larger models trained on more diverse corpora tend to internalize a broader scope of world knowledge [33, 37]. In contrast, decision boundaries are primarily shaped through explicit tool use training [13], leading to variation in how models interact with tools to acquire knowledge.

Lemma 2.2: *There exist minimal and maximal knowledge (and decision) boundaries across all models.* The *minimal knowledge boundary* $\partial\mathcal{K}_{\min} = \bigcap_{i=1}^{N} \partial\mathcal{K}^{(i)}$ represents the smallest common set of internalized knowledge shared by all models, regardless of their training setup. Conversely, the *maximal knowledge boundary* $\partial\mathcal{K}_{\max} = \bigcup_{i=1}^{N} \partial\mathcal{K}^{(i)}$ reflects the union of all internal knowledge across models, encompassing even niche or domain-specific knowledge found only in specialized systems. Analogously, minimal and maximal decision boundaries exist, though they are best interpreted as normative alignment goals rather than fixed, objective thresholds.

3.4 Principle 3: Dynamic Conservation

Knowledge is inherently dynamic, continuously evolving as new facts emerge and old ones become obsolete. To capture this temporality, we propose the principle of dynamic conservation of knowledge, emphasizing how models must adapt to an ever-changing epistemic landscape.

Lemma 3.1: *At any time step t, the total world knowledge $\mathcal{W}_t$ is fixed and identical across all models.* Ideally, a model would internalize the entire knowledge set, i.e., $\mathcal{K}_{\text{int}}(m, t) = \mathcal{W}_t$, requiring no external tool use. This entails an aspirational endpoint for fully autonomous intelligence [14]. Practically, however, as $\mathcal{W}_t$ expands over time, models must also evolve to keep pace. If a model's epistemic growth outpaces that of the external world, this ideal state becomes theoretically attainable.

[1] `https://openai.com/sora/`

Lemma 3.2: *For any task or query q and model m, there exists a minimal and fixed epistemic effort $N(q,m)$, allocated between internal and external sources, that is necessary to solve the task.* This can be decomposed as $N(q,m) = k_{\text{int}} + k_{\text{ext}}$, where k_{int} reflects knowledge retrieved from the model's internal parameters and k_{ext} represents knowledge acquired through external tools. This formulation reveals several insights: (1) $N(q,m)$ is jointly determined by the complexity of the task and the capabilities of the model, indicating stronger models may satisfy most or all of N through internal reasoning ($k_{\text{int}} \to N$), while weaker models may depend more on external assistance ($k_{\text{ext}} \to N$) [48]. (2) Even models with limited internal capacity can achieve high performance by dynamically offloading reasoning or retrieval steps to more capable tools or agents. This suggests a form of capability equivalence, where optimal tool use policies allow weaker models to simulate stronger ones. (3) The objective is not merely task completion, but the development of behavior policies that minimize epistemic effort while managing latency, cost, and cognitive load. In this view, intelligent behavior is defined not just by the correctness of outputs, but by the efficiency and adaptiveness of the pathways taken to reach them. We expand on these implications in Appendix A.

4 Agents: From Knowing to Reasoning and Acting

Building on the principles we discussed, we now explore how these principles shape the design and behavior of intelligent agents. As outlined in Section 2.2, we define an agent as a decision-making entity that coordinates internal cognitive tools to retrieve internal knowledge (i.e., reason) and external physical tools to acquire external knowledge (i.e., act). At each step in a task, the agent must decide which tool to invoke based on its current state and what knowledge is needed to move closer to a solution. This iterative process continues until the agent accumulates sufficient knowledge to produce a final answer or achieve its goal. Drawing on the concept of the agent's knowledge boundary and decision boundary, we arrive at the central position of this paper:

> **Position: Decision-Knowledge Alignment Principle**
>
> For an agent to achieve decision optimality, its tool use decision boundary should align with its knowledge boundary. This alignment represents the most efficient way to producing correct answers.

In other words, an intelligent agent should invoke internal tools when the needed knowledge lies within its parametric capacity and turn to external tools when that knowledge must be acquired from the environment. This alignment ensures that the agent's behavior is both efficient and epistemically grounded, leading to more robust and adaptive decision-making. In this section, we first justify our position by examining how alignment between decision and knowledge boundaries can be operationalized during both agent training and inference (Section 4.1). We then inspect what constitutes optimal agent behavior under this principle (Section 4.2), and finally, we outline practical pathways for building agents that achieve optimality in practice (Section 4.3).

4.1 Alignment of Decision and Knowledge Boundary

Meta-Cognition. Meta-cognition refers to the ability to monitor and regulate one's own cognitive processes: knowing what one knows, recognizing uncertainty, and selecting appropriate strategies accordingly [49]. In the context of intelligent agents, meta-cognition is the agent's capacity to assess whether the knowledge required to progress lies within its internal parametric space or must be acquired through external tools. Just as humans are often governed by an implicit heuristic to draw on external help when uncertain and reason internally when confident, agents must also learn to make such distinctions contextually. Therefore, achieving alignment between the knowledge and decision boundary ultimately requires cultivating accurate and adaptive meta-cognition, both during training and at inference time.

Training-Time Alignment Dynamic. After pretraining, a model's parametric knowledge boundary becomes relatively static, reflecting what its known knowledge that can be elicited. In contrast, the decision boundary remains adjustable during the model alignment phase. As shown in Figure 2, misalignment between these boundaries leads to two primary failure modes. If a model uses internal tools for knowledge it does not actually possess, this results in hallucinations or incorrect reasoning due to internal tool overuse [50]. Conversely, if the model defers to external tools despite already knowing the answer, it wastes computation and time, an inefficiency stemming from external tool

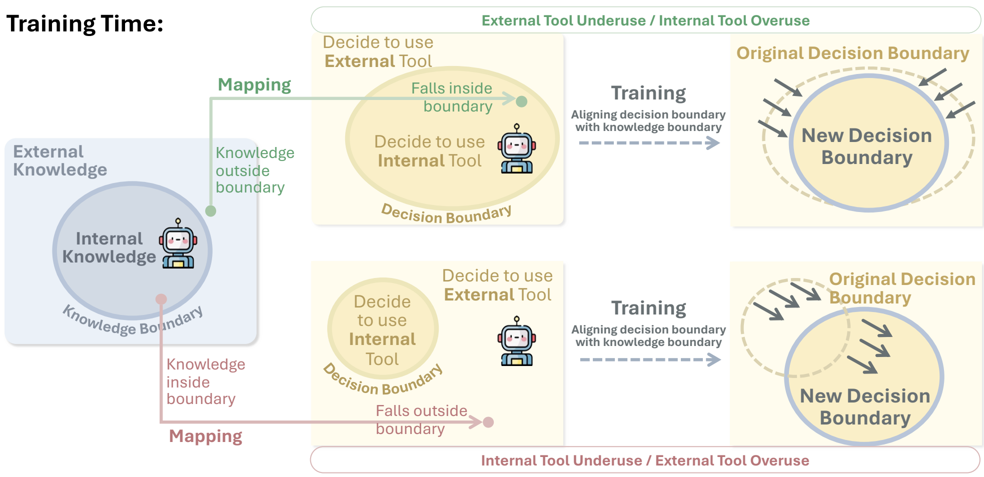


Figure 2: Training should dynamically adjust the decision boundary relative to the fixed knowledge boundary to optimize efficiency, minimize hallucinations, and prevent tool overuse or underuse.

overuse [47]. These conclusions actually prove our position from the opposite side that aligning the decision boundary with the knowledge boundary is the most *efficient* strategy for an agent to arrive at the *correct* answer.

To realize this efficient and accurate behavior, alignment training must enable the model to make tool use decisions based on a calibrated understanding of its own knowledge limits. This involves fostering meta-cognition: the ability to recognize what is known versus unknown. Techniques such as supervised fine-tuning with explicit tool labels or reinforcement learning with task-based feedback can guide the model toward more effective tool use strategies, which we will elaborate in Section 4.3. Overall, our goal is to shape a dynamic decision boundary that aligns closely with the model's knowledge boundary, enabling more accurate and resource efficient problem solving.

Inference-Time Alignment Dynamic. During inference, the knowledge required to answer a specific query is limited and often partially unknown. As shown in Figure 3, the agent begins with an incomplete picture of what it needs to know. By interacting with external tools, such as making API calls or executing actions, the agent retrieves missing information and integrates it into its context. This process incrementally expands the model's effective knowledge boundary, forming a temporary, task-specific epistemic frontier that evolves as inference progresses.

Reasoning during inference thus becomes a dynamic feedback loop, where the agent alternates between internal cognition and external exploration. Meta-cognition plays a central role in this process: the agent must continually assess whether it possesses sufficient knowledge to proceed or should gather more. Without this adaptive self-assessment, the agent risks terminating prematurely or inefficiently overusing tools. Robust inference-time meta-cognition enables agents to regulate this loop effectively: balancing accuracy, efficiency, and completeness in real-time decision-making.

4.2 Optimal Agent Behavior

Aligning an agent's decision boundary with its knowledge boundary is empirically challenging, as the knowledge boundary is inherently abstract and often difficult to detect without extensive probing [38, 39]. Therefore, we shift our focus from detecting the boundary itself to evaluating the behavior of the agent - specifically, what we consider optimal from a human perspective. While correctness is a primary goal, as discussed in the previous section, an agent can still produce correct answers while inefficiently overusing external tools. Thus, correctness alone is not sufficient evidence of alignment. To better characterize optimal agent behavior, we argue that *efficiency* should accompany correctness, as an ideal agent not only solves the problem but does so with judicious coordination of its internal and external tools. In this section, we examine four representative agent behaviors based on their patterns of tool use, evaluating the strengths and weaknesses of each in terms of alignment and efficiency in agent decision-making.

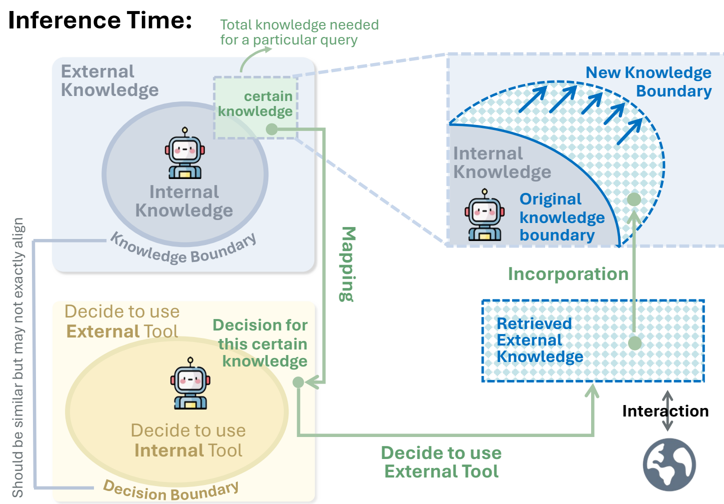


Figure 3: Inference-time alignment depends on real-time expansion of the knowledge boundary through interaction, requiring adaptive meta-cognition to balance completeness and efficiency.

- **Maximizing both internal and external physical tool use.** In this case, the agent produces the correct answer but does so through excessive use of both internal and external tools, regardless of necessity. This behavior is inefficient, consumes unnecessary resources, and increases the risk of error propagation or tool misuse. It also obscures the decision-making process, reducing transparency and trust. Rather than reflecting strategic reasoning, this mirrors brute-force search, which is misaligned with the goals of scalable and interpretable AI systems.
- **Maximizing external tool and minimizing internal tool use.** This entails the agent over-relies on external tools while underutilizing its internal reasoning capacity. This may yield correct results, especially for smaller models, but it results in inefficiency and increased dependence on external systems. More importantly, it conflicts with the core aim of model scaling: to internalize knowledge within parameters. By deferring to external tools, the agent misses opportunities to reinforce and generalize its own representations, limiting long-term autonomy and adaptability.
- **Maximizing internal tools and minimizing external tool use.** In this setup, the agent leans heavily on internal reasoning and avoids external tool use [51]. This behavior promotes autonomy and efficiency, especially in constrained environments, and aligns with the principle of maximizing model capacity. However, excessive internal deliberation can lead to overthinking, producing unnecessarily long reasoning chains. While this reflects strong use of internal knowledge, it may overlook more efficient external solutions in certain cases (See $ B), indicating a need for better tool use calibration.
- **Minimizing both internal and external physical tool use.** This represents the most efficient trajectory: solving tasks with minimal use of tools, internal [52] or external [51]. It reflects optimal behavior of using tools only when necessary, guided by precise self-monitoring and calibrated decision-making. However, extreme minimalism can risk underthinking or skipping essential steps, especially in complex tasks. In addition, empirically training agents toward this behavior is difficult, as it requires balancing correctness with efficiency, which is a more delicate optimization than correctness alone.

4.3 Paths for Agent Optimality

Among the four agent behaviors discussed, the third and fourth configurations offer more promising directions for building efficient and autonomous agents. While they differ in their reliance on internal reasoning, both aim to minimize reliance on external systems while preserving task success. As discussed in Section 4.1, this objective implicitly reflects an alignment between the decision and knowledge boundaries, yet provides a more actionable proxy that avoids the need to explicitly probe the agent's often abstract and difficult-to-measure knowledge boundary.

This perspective also aligns with the expert vision that *autonomous machine intelligence should minimize the number of real-world actions required to solve a task* [14]. However, achieving such minimization remains an open challenge. In this section, we examine existing approaches that aim to reduce external tool use, analyzing their strategies, assumptions, and limitations in the context of scalable and efficient agent design.

Agentic Pretraining. Next-token prediction has been foundational in compressing world knowledge into a model's parametric space [36]. However, this alone does not equip models to acquire new knowledge through interaction. We propose augmenting this paradigm with **next-tool prediction**: a learning objective where the model learns to predict the most appropriate external tool to invoke at each step. This transforms interaction itself into a first-class modeling target, allowing the agent to learn how to gather information it doesn't already possess. As research trends toward unified agent architectures, modeling all forms of interaction (API calls, UI navigation, or environment manipulation) as structured, learnable outputs opens the door to a new kind of scaling law: one that governs knowledge acquisition, not just compression. This shift is essential for building adaptive, self-improving agents in open-ended, dynamic environments [3, 53].

Agentic Supervised Fine-tuning. Supervised fine-tuning (SFT) remains the most common approach for teaching agents tool use, using small task-specific datasets (e.g., math, code) to demonstrate when and how to call external tools [12, 13]. However, it often assumes a uniform knowledge boundary across models, which is unrealistic. As discussed in Lemma 2.1, this mismatch leads to inefficiencies: what is helpful for a small model may be redundant or even distracting for larger ones. One solution is to create custom SFT datasets tailored to each model's knowledge boundary, but this is resource-intensive and hard to scale. A more practical alternative, as outlined in Lemma 2.2, is to approximate a maximal knowledge boundary and train agents to defer intelligently when faced with unfamiliar content [47]. While this approach offers greater generality, it may lack the precision needed for fine-grained domains, highlighting a trade-off between scalability and behavioral fidelity.

Agentic Reinforcement Learning. Reinforcement learning (RL) offers a more promising path for aligning a model's decision-making with its own knowledge boundary, as agents can learn from experience how to adaptively use tools. The key challenge lies in designing reward functions that go beyond correctness. While many RL agents are trained to maximize answer accuracy, this ignores how the answer is reached, including whether reasoning is efficient, whether tool use is justified, and whether the trajectory is optimal [30, 54]. Recent work like OTC-PO [51] addresses this by balancing correctness with penalties for unnecessary tool calls, encouraging agents to act with restraint and self-awareness. By optimizing not only for outcomes but for processes, RL can produce agents that are not only accurate but also efficient, interpretable, and better aligned with real-world deployment constraints [2].

5 Conclusion

In this position paper, we introduced a unified epistemic theory of agents that reframes reasoning and acting as equivalent tools. By aligning an agent's decision-making boundary with its knowledge boundary, we advocate for the design of agents that are not only capable of completing tasks but do so with epistemic efficiency - minimizing unnecessary interactions while maximizing knowledge gain to achieve task success. This perspective moves beyond the current paradigm of tool-augmented LLMs and points toward a future in which agents exhibit genuine autonomy grounded in principled decision-making.

Looking ahead, this framework offers a roadmap for developing foundation agents that can operate effectively in open-ended environments, learn efficiently with minimal supervision, and generalize across domains. By emphasizing knowledge - driven intelligence, this theory invites a rethinking of how we measure and build capable AI agents - not by their frequency of action, but by their ability to know when to reason and when to act. We believe this epistemic perspective will play a foundational role in the next generation of AI systems, shaping agents that are not only more capable but also safer, more reliable, and better aligned with human values and long-term goals.

[2] We left some discussion ragarding future directions in Appendix C.

References

[1] Noam Kolt. Governing ai agents. *arXiv preprint arXiv:2501.07913*, 2025.

[2] Jian Xie, Kai Zhang, Jiangjie Chen, Tinghui Zhu, Renze Lou, Yuandong Tian, Yanghua Xiao, and Yu Su. Travelplanner: A benchmark for real-world planning with language agents. *arXiv preprint arXiv:2402.01622*, 2024.

[3] Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094, 2024.

[4] Hongru Wang, Rui Wang, Boyang Xue, Heming Xia, Jingtao Cao, Zeming Liu, Jeff Z. Pan, and Kam-Fai Wong. AppBench: Planning of multiple APIs from various APPs for complex user instruction. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15322–15336, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[5] Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, Wanjun Zhong, Kuanye Li, Jiale Yang, Yu Miao, Woyu Lin, Longxiang Liu, Xu Jiang, Qianli Ma, Jingyu Li, Xiaojun Xiao, Kai Cai, Chuang Li, Yaowei Zheng, Chaolin Jin, Chen Li, Xiao Zhou, Minchao Wang, Haoli Chen, Zhaojian Li, Haihua Yang, Haifeng Liu, Feng Lin, Tao Peng, Xin Liu, and Guang Shi. Ui-tars: Pioneering automated gui interaction with native agents, 2025.

[6] Junde Wu, Jiayuan Zhu, and Yuyuan Liu. Agentic reasoning: Reasoning llms with tools for the deep research. *arXiv preprint arXiv:2502.04644*, 2025.

[7] Thao Nguyen, Tiara Torres-Flores, Changhyun Hwang, Carl Edwards, Ying Diao, and Heng Ji. Glad: Synergizing molecular graphs and language descriptors for enhanced power conversion efficiency prediction in organic photovoltaic devices. In *Proc. 33rd ACM International Conference on Information and Knowledge Management (CIKM 2024)*, 2024.

[8] Carl Edwards, Aakanksha Naik, Tushar Khot, Martin Burke, Heng Ji, and Tom Hope. Synergpt: In-context learning for personalized drug synergy prediction and drug design. In *Proc. 1st Conference on Language Modeling (COLM2024)*, 2024.

[9] Carl Edwards, Chi Han, Gawon Lee, Thao Nguyen, Chetan Kumar Prasad, Sara Szymkuc, Bartosz A. Grzybowski, Bowen Jin, Ying Diao, Jiawei Han, Ge Liu, Hao Peng, Martin D. Burke, and Heng Ji. mclm: A function-infused and synthesis-friendly modular chemical language model. In *arxiv*, 2025.

[10] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

[11] Pei Zhou, Jay Pujara, Xiang Ren, Xinyun Chen, Heng-Tze Cheng, Quoc V Le, Ed Chi, Denny Zhou, Swaroop Mishra, and Huaixiu Steven Zheng. Self-discover: Large language models self-compose reasoning structures. *Advances in Neural Information Processing Systems*, 37:126032–126058, 2024.

[12] Zhibin Gou, Zhihong Shao, Yeyun Gong, Yujiu Yang, Minlie Huang, Nan Duan, Weizhu Chen, et al. Tora: A tool-integrated reasoning agent for mathematical problem solving. In *The Twelfth International Conference on Learning Representations*.

[13] Chengpeng Li, Mingfeng Xue, Zhenru Zhang, Jiaxi Yang, Beichen Zhang, Xiang Wang, Bowen Yu, Binyuan Hui, Junyang Lin, and Dayiheng Liu. Start: Self-taught reasoner with tools, 2025.

[14] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022.

[15] Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*, 2023.

[16] Xiaoqian Liu, Xingzhou Lou, Jianbin Jiao, and Junge Zhang. Position: Foundation agents as the paradigm shift for decision making. *arXiv preprint arXiv:2405.17009*, 2024.

[17] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6), March 2024.

[18] James J Gibson. *The ecological approach to visual perception: classic edition*. Psychology press, 2014.

[19] David H Jonassen. What are cognitive tools? In *Cognitive tools for learning*, pages 1–6. Springer, 1992.

[20] Piet AM Kommers, David H Jonassen, and J Terry Mayes. *Cognitive tools for learning*. Springer, 1992.

[21] WANG Hongru, Deng Cai, Wanjun Zhong, Shijue Huang, Jeff Z Pan, Zeming Liu, and Kam-Fai Wong. Self-reasoning language models: Unfold hidden reasoning chains with few reasoning catalyst. In *Workshop on Reasoning and Planning for Large Language Models*, 2025.

[22] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[23] Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. Chameleon: Plug-and-play compositional reasoning with large language models. *Advances in Neural Information Processing Systems*, 36:43447–43478, 2023.

[24] Pan Lu, Bowen Chen, Sheng Liu, Rahul Thapa, Joseph Boen, and James Zou. Octotools: An agentic framework with extensible tools for complex reasoning. *arXiv preprint arXiv:2502.11271*, 2025.

[25] Hongru Wang, Huimin Wang, Lingzhi Wang, Minda Hu, Rui Wang, Boyang Xue, Yongfeng Huang, and Kam-Fai Wong. Tpe: Towards better compositional reasoning overnbsp;cognitive tools vianbsp;multi-persona collaboration. In *Natural Language Processing and Chinese Computing: 13th National CCF Conference, NLPCC 2024, Hangzhou, China, November 1–3, 2024, Proceedings, Part II*, page 281–294, Berlin, Heidelberg, 2024. Springer-Verlag.

[26] Hongru Wang, Rui Wang, Fei Mi, Yang Deng, Zezhong Wang, Bin Liang, Ruifeng Xu, and Kam-Fai Wong. Cue-CoT: Chain-of-thought prompting for responding to in-depth dialogue questions with LLMs. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12047–12064, Singapore, December 2023. Association for Computational Linguistics.

[27] Shibo Hao, Tianyang Liu, Zhen Wang, and Zhiting Hu. Toolkengpt: Augmenting frozen language models with massive tools via tool embeddings. *Advances in neural information processing systems*, 36:45870–45894, 2023.

[28] Dongge Han, Trevor McInroe, Adam Jelley, Stefano V. Albrecht, Peter Bell, and Amos Storkey. LLM-personalize: Aligning LLM planners with human preferences via reinforced self-training for housekeeping robots. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert, editors, *Proceedings of the 31st International Conference on Computational Linguistics*, pages 1465–1474, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics.

[29] DeepSeek-AI Team. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025.

[30] Bowen Jin, Hansi Zeng, Zhenrui Yue, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning, 2025.

[31] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China, November 2019. Association for Computational Linguistics.

[32] Boxi Cao, Hongyu Lin, Xianpei Han, Le Sun, Lingyong Yan, Meng Liao, Tong Xue, and Jin Xu. Knowledgeable or educated guess? revisiting language models as knowledge bases. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1860–1874, Online, August 2021. Association for Computational Linguistics.

[33] Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online, November 2020. Association for Computational Linguistics.

[34] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024.

[35] Guiyao Tie, Zeli Zhao, Dingjie Song, Fuyang Wei, Rong Zhou, Yurou Dai, Wen Yin, Zhejian Yang, Jiangyue Yan, Yao Su, Zhenhan Dai, Yifeng Xie, Yihan Cao, Lichao Sun, Pan Zhou, Lifang He, Hechang Chen, Yu Zhang, Qingsong Wen, Tianming Liu, Neil Zhenqiang Gong, Jiliang Tang, Caiming Xiong, Heng Ji, Philip S. Yu, and Jianfeng Gao. A survey on post-training of large language models, 2025.

[36] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020.

[37] Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.3, knowledge capacity scaling laws. In *The Thirteenth International Conference on Learning Representations*, 2025.

[38] Ruiyang Ren, Yuhao Wang, Yingqi Qu, Wayne Xin Zhao, Jing Liu, Hua Wu, Ji-Rong Wen, and Haifeng Wang. Investigating the factual knowledge boundary of large language models with retrieval augmentation. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert, editors, *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3697–3715, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics.

[39] Moxin Li, Yong Zhao, Yang Deng, Wenxuan Zhang, Shuaiyi Li, Wenya Xie, See-Kiong Ng, and Tat-Seng Chua. Knowledge boundary of large language models: A survey, 2024.

[40] Xunjian Yin, Xu Zhang, Jie Ruan, and Xiaojun Wan. Benchmarking knowledge boundary for large language models: A different perspective on model evaluation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2270–2286, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[41] Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. Can we edit factual knowledge by in-context learning? In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4862–4876, Singapore, December 2023. Association for Computational Linguistics.

[42] Siyuan Cheng, Ningyu Zhang, Bozhong Tian, Xi Chen, Qingbin Liu, and Huajun Chen. Editing language model-based knowledge graph embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 17835–17843, 2024.

[43] XiaoQi Han, Ru Li, Xiaoli Li, Jiye Liang, Zifang Zhang, and Jeff Pan. InstructEd: Soft-instruction tuning for model editing with hops. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14953–14968, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[44] Song Wang, Yaochen Zhu, Haochen Liu, Zaiyi Zheng, Chen Chen, and Jundong Li. Knowledge editing for large language models: A survey, 2024.

[45] Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. Editing large language models: Problems, methods, and opportunities. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10222–10240, Singapore, December 2023. Association for Computational Linguistics.

[46] Hongru Wang, Boyang Xue, Baohang Zhou, Tianhua Zhang, Cunxiang Wang, Huimin Wang, Guanhua Chen, and Kam-Fai Wong. Self-DC: When to reason and when to act? self divide-and-conquer for compositional unknown questions. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *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 6510–6525, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.

[47] Cheng Qian, Emre Can Acikgoz, Hongru Wang, Xiusi Chen, Avirup Sil, Dilek Hakkani-Tür, Gokhan Tur, and Heng Ji. Smart: Self-aware agent for tool overuse mitigation, 2025.

[48] Yash Akhauri, Anthony Fei, Chi-Chih Chang, Ahmed F. AbouElhamayed, Yueying Li, and Mohamed S. Abdelfattah. Splitreason: Learning to offload reasoning, 2025.

[49] Rakefet Ackerman and Valerie A. Thompson. Meta-reasoning: Monitoring and control of thinking and reasoning. *Trends in Cognitive Sciences*, 21(8):607–617, 2017.

[50] Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning llms on new knowledge encourage hallucinations? *arXiv preprint arXiv:2405.05904*, 2024.

[51] Hongru Wang, Cheng Qian, Wanjun Zhong, Xiusi Chen, Jiahao Qiu, Shijue Huang, Bowen Jin, Mengdi Wang, Kam-Fai Wong, and Heng Ji. Otc: Optimal tool calls via reinforcement learning, 2025.

[52] Daman Arora and Andrea Zanette. Training language models to reason efficiently, 2025.

[53] Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Erran Li Li, Ruohan Zhang, et al. Embodied agent interface: Benchmarking llms for embodied decision making. *Advances in Neural Information Processing Systems*, 37:100428–100534, 2024.

[54] Xuefeng Li, Haoyang Zou, and Pengfei Liu. Torl: Scaling tool-integrated rl, 2025.

[55] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[56] Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning LLMs on new knowledge encourage hallucinations? In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7765–7784, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[57] Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. R-tuning: Instructing large language models to say ‘I don‘t know’. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7113–7139, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[58] Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. Don‘t hallucinate, abstain: Identifying LLM knowledge gaps via multi-LLM collaboration. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14664–14690, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[59] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[60] Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. Do large language models know what they don‘t know? In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665, Toronto, Canada, July 2023. Association for Computational Linguistics.

[61] Alfonso Amayuelas, Kyle Wong, Liangming Pan, Wenhu Chen, and William Yang Wang. Knowledge of knowledge: Exploring known-unknowns uncertainty with large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6416–6432, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[62] Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it‘s lying. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore, December 2023. Association for Computational Linguistics.

[63] Ziwei Ji, Delong Chen, Etsuko Ishii, Samuel Cahyawijaya, Yejin Bang, Bryan Wilie, and Pascale Fung. LLM internal states reveal hallucination risk faced with a query. In Yonatan Belinkov, Najoung Kim, Jaap Jumelet, Hosein Mohebbi, Aaron Mueller, and Hanjie Chen, editors, *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 88–104, Miami, Florida, US, November 2024. Association for Computational Linguistics.

[64] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, 2023.

[65] Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*, 2022.

[66] Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*, 2023.

[67] Potsawee Manakul, Adian Liusie, and Mark Gales. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore, December 2023. Association for Computational Linguistics.

[68] Tianyang Xu, Shujin Wu, Shizhe Diao, Xiaoze Liu, Xingyao Wang, Yangyi Chen, and Jing Gao. SaySelf: Teaching LLMs to express confidence with self-reflective rationales. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5985–5998, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[69] Boyang Xue, Fei Mi, Qi Zhu, Hongru Wang, Rui Wang, Sheng Wang, Erxin Yu, Xuming Hu, and Kam-Fai Wong. Ualign: Leveraging uncertainty estimations for factuality alignment on large language models, 2024.

[70] Jintian Zhang, Xin Xu, Ningyu Zhang, Ruibo Liu, Bryan Hooi, and Shumin Deng. Exploring collaboration mechanisms for LLM agents: A social psychology view. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14544–14607, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

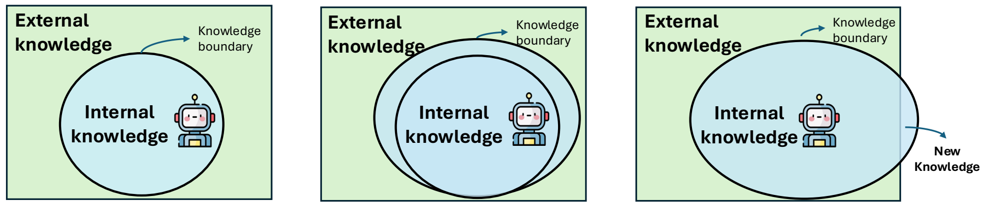
[71] OpenAI Team. Openai o1 system card, 2024.

[72] Qwen Team. Qwq-32b: Embracing the power of reinforcement learning, March 2025.

[73] Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. Generate rather than retrieve: Large language models are strong context generators. In *The Eleventh International Conference on Learning Representations*.

[74] Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge conflicts for LLMs: A survey. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8541–8565, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[75] Yu Zhao, Alessio Devoto, Giwon Hong, Xiaotang Du, Aryo Pradipta Gema, Hongru Wang, Xuanli He, Kam-Fai Wong, and Pasquale Minervini. Steering knowledge selection behaviours in LLMs via SAE-based representation engineering. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *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 5117–5136, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.



(a) Knowledge Boundary (b) Knowledge Expansion (c) New Knowledge Discovery

Figure 4: A high-level illustration of Lemma 1.1 for a specific model m.

A Further Discussion: Principles of Knowledge and Decision in Model

A.1 Principle 1: Foundation of Knowledge and Decision Boundary

Lemma 1.1: Some may argue that it is possible that the scaling law does not work and the expansion may stop at a specific timestep. In this case, there always some knowledge that the models can not capture or master, and the model must learn these via another ways, such as learn from interaction, or human experts, to expand the knowledge boundary [38].

Lemma 1.2: It is always desired that the model can gain or update knowledge in specific domains without affecting other domains. However, this is extremely challenging, as knowledge in the real world is deeply interconnected rather than isolated. During learning on new experience, models are prone to catastrophic forgetting [55] or hallucinate [56, 57], where previously acquired knowledge fades.

In summary, Lemma 1.1 and Lemma 1.2 call for a scalable lifelong learning paradigm that can dynamically expand and redistribute the model's knowledge boundary $\partial\mathcal{K}$ in response to new data and evolving tasks across the time.

A.2 Principle 2: Uniqueness and Diversity of Knowledge and Decision Boundary

There are lots of models from both open-source and close-source sides. To better understand the diversity and limitations of model capabilities, we posit that there exist both unique and universal properties shared across all models [58].

Lemma 2.1: There are several lines of research direction being affected by this lemma. One line of research is the knowledge boundary identification through prompting [59–61], probing [62, 63], uncertainty estimation [64–66], and self-consistency checks [67]. For instance, several studies try to collect *model-specific* supervised fine-tuning dataset to teach the model to say "I do not know" for the unknown questions and only provide answer for known questions [57, 68, 69]. Another line of work involves model specialization and collaborative inference where models with complementary boundaries (e.g., generalist vs. domain experts) are orchestrated to jointly solve complex tasks [58, 70]. This is particularly relevant in modular systems or tool-integrated agents, where models selectively offload tasks based on different specialists.

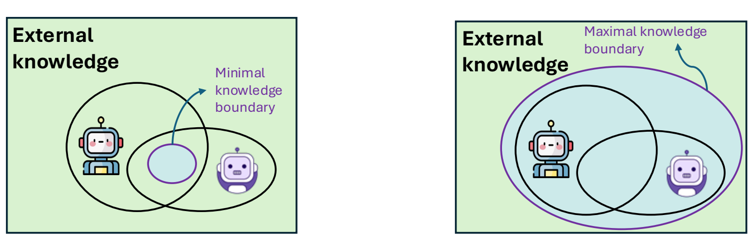


Figure 5: A high-level illustration of Lemma 2.2 for the all models $\mathcal{M} = \{m_0, ..., m_n\}$.

Lemma 2.2: Both minimal and maximal knowledge boundaries play key roles to better understand the mental world of the models. On the minimal side, $\partial\mathcal{K}_{\min}$ captures the foundational knowledge consistently learned by all models, such as basic language structures, common facts, and widely shared cultural concepts. It effectively defines a shared epistemic core - a common worldview shaped by dominant patterns in pretraining data. This boundary reflects the most universal priors across models and has important implications for alignment, fairness, and generalization. On the maximal side, understanding $\partial\mathcal{K}_{\max}$ helps reveal the outer limits of what any existing model can know. This insight motivates strategies like one-fits-all supervised fine-tuning, where models are trained to defer or trigger specific actions, such as tool use or human intervention, when a task requires knowledge beyond this boundary. For example, several studies collect the well-designed dataset to finetune the model to only call tools when the required knowledge is outside the inherent parametric knowledge of LLMs, and therefore the dataset can apply for all model [47].

A.3 Principle 3: Dynamic Conservation of Knowledge

Knowledge is inherently dynamic—the world is constantly evolving as new information is discovered and outdated knowledge becomes obsolete. To capture this temporal nature of knowledge, we propose the principle of dynamic conservation of knowledge.

Lemma 3.1: Current mainstream models, however, primarily function as knowledge distributors rather than knowledge discoverers. They optimize for efficiency and effectiveness in retrieving, synthesizing and applying existing knowledge, for example, improving task automation, boosting productivity, and supporting human decision making. The emerging paradigm of *AI for Science* seeks to bridge this gap by leveraging models not only to encode and apply existing knowledge but to generate novel hypotheses, identify hidden patterns, and accelerate scientific discovery.

Lemma 3.2: There are several RL-based approaches focus exclusively on optimizing final answer correctness, without accounting for the underlying reasoning trajectory. In detail, several large reasoning models (LRMs), such as OpenAI's o1 [71], DeepSeek-R1 [29], and QwQ [72], achieve exceptional performance by being optimized solely for final answer correctness, regardless of the number of reasoning tokens used or the utility of the reasoning process itself. Moreover, few studies try to re-produce the success of RL in tool-integrated reasoning which also only focus on the final answer correctness [30, 54]. As a result, models often develop inefficient or suboptimal behaviors, including overthinking, where excessive internal reasoning leads to inflated t_{int}, and tool overuse, where models make frequent, unproductive calls to external tools, increasing t_{ext}. Although few studies focus on the this issue, they mainly focus on one side either the internal or external. In contrast, we argue that a more principled and holistic framework is required, one that views internal and external tools as complementary actions within a unified decision-making process. Such a framework should aim not just to maximize correctness, but also to approximate the minimal epistemic effort $N(q, m)$ required for task completion.

Together, these three principles establish a unified theoretical framework for understanding, analyzing, and improving the knowledge structures and behaviors of models and agents, offering foundational guidance for developing models that are not only powerful and effective, but also efficient, adaptive, and epistemically aware. It is believed to guide the design of next-generation models or agents that can reason more effectively, learn continuously, collaborate strategically, and act responsibly in open-ended, evolving environments.

B Other Relationship Between Internal Knowledge and External Knowledge

In the main pages, we assume that the internal knowledge and external knowledge are two separated parts for simplicity and generalization. In practice, it may not hold since the internal knowledge may overlap with external knowledge, and there may exist knowledge conflict between these two sources. We discuss these situations as follows:

Knowledge Overlap. This highlights an important possibility: internal cognitive tools and certain external physical tools can retrieve overlapping or even identical pieces of knowledge, implying a potential for epistemic transferability between the two. For example, a model may answer a factual question either by recalling internalized knowledge from its parameters or by querying an external tool such as a search engine, both pathways leading to the same correct answer [73, 46]. A similar phenomenon occurs with tasks like simple mathmatical operation, where the model may either compute the result internally or delegate it to an external calculator API. This interchangeability suggests that internal and external tools can act as substitutes under certain conditions, raising further questions about when and how agents should transfer, balance, or even fuse internal reasoning with external interaction for optimal epistemic efficiency. In these cases, minimizing external physical tools is also maximizing internal cognitive tools as evidenced by recent study [51]. We emphasize the opposite is not necessarily true: maximizing the use of external physical tools does not imply minimizing the use of internal cognitive tools since not all external tools can map to specific internal tools. In many cases, excessive reliance on external tools may reflect insufficient internal reasoning rather than optimized epistemic behavior.

Knowledge Conflict. In some cases, internal and external knowledge sources may conflict [74, 75], leading to inconsistent or contradictory information. This typically arises when the model retrieves outdated, incomplete, or hallucinated content from its internal memory that contradicts more up-to-date or accurate external sources. Such conflict is especially pronounced when the model attempts to generate knowledge beyond its internal boundary, often resulting in hallucinations [50]. For instance, a model may confidently generate an incorrect answer based on memorized but obsolete knowledge, even when a correct answer is accessible via an external tool like a search engine or database. These situations highlight the importance of epistemic calibration: the model must learn not only what it knows, but also when its internal knowledge is unreliable and should be overridden by external input. Addressing knowledge conflict requires mechanisms for knowledge arbitration, where agents resolve discrepancies by evaluating the reliability, recency, and epistemic certainty of each source - an open challenge for building robust decision boundaries under uncertainty.

C Future Directions

Vision Agent. Vision agents extend our unified framework of reasoning and acting by incorporating visual affordances as part of the decision-making loop. In our definition, external physical tools are invoked based on an agent's knowledge gaps; in vision agents, visual input becomes a direct means of detecting such gaps and informing tool use decisions. To realize this, future systems should treat visual understanding not as passive recognition but as actionable epistemic input. This involves embedding affordance-aware modules into vision-language models that not only recognize objects but predict possible interactions. Moreover, meta-cognitive control should guide visual attention: the agent must actively attend to regions most likely to resolve its uncertainty. Training in simulation with reinforcement learning can allow agents to learn the utility of visual exploration for acquiring external knowledge, enabling more precise tool invocation grounded in perception.

Embodied Agent. Embodied agents concretize the external physical tool dimension by extending it into the physical world, where the agent's own body becomes a tool, and the environment imposes dynamic constraints. Within our framework, this embodiment means that the agent's knowledge boundary is not only cognitive but also physically bounded (e.g., what can be seen, reached, or manipulated). To operationalize this, agents should be equipped with real-time sensorimotor feedback loops and control modules that treat actions as epistemic moves: physical actions (e.g., MoveTo, PickUp) should be treated like external tool calls that yield knowledge from the environment. Learning here must be closed-loop and incremental—using reinforcement signals from physical interaction to adjust the decision boundary over time. Physical meta-cognition, such as failure detection or confidence in execution, should guide whether to reason further, retry an action, or explore alternatives.

Multi-Agent Coordination. Multi-agent coordination extends our framework from individual agents aligning their decision and knowledge boundaries to a collective setting where these boundaries are distributed across multiple agents. In this paradigm, each agent operates with a local view (its own knowledge and decision boundaries), but contributes to a shared task by reasoning about and interacting with other agents. The key challenge is aligning these distributed boundaries to form a coherent collective intelligence. To achieve this, agents must be equipped with mechanisms to communicate epistemic state, and dynamically delegate subtasks to peers whose knowledge boundaries better match the problem context. This requires structured communication protocols, role inference strategies, and shared meta-cognitive modules that manage when to ask, respond, or act. Practically, this can be developed through multi-agent reinforcement learning in environments where cooperation is required for successful task completion, with reward functions encouraging efficient division of cognitive and physical labor.