The State of the Art in LLM Reasoning: A 2024-2025 Synthesis of Methods, Benchmarks, and Future Trajectories

Executive Summary & Introduction: The Evolving Landscape of Automated Cognition

Context and Motivation

The field of artificial intelligence has undergone a significant transformation, moving from a primary focus on the linguistic fluency of Large Language Models (LLMs) to a more rigorous scrutiny and enhancement of their reasoning capabilities. While early large-scale models demonstrated emergent reasoning as a byproduct of next-token prediction on vast datasets, the 2024–2025 period is characterized by the deliberate engineering of cognitive architectures designed to make reasoning more robust, efficient, and verifiable.[1] Despite their proficiency in language-based tasks, a notable gap persists between the language capabilities of LLMs and their capacity for true, multi-step reasoning.[4] This distinction is critical, as the deployment of AI in sensitive and high-stakes domains such as healthcare, finance, law, and scientific discovery necessitates a high degree of reliability and trustworthiness that can only be achieved through sound reasoning.[4] Consequently, enhancing and understanding LLM reasoning has become a central research priority.

Report Scope and Structure

This report provides a systematic and comprehensive survey of the state-of-the-art in LLM

reasoning, with a focus on developments from 2024 through early 2025. The analysis is structured around four principal paradigms that trace the evolution of reasoning techniques:

1. **Foundational Prompting and Decoding Strategies:** Examining the initial breakthroughs that elicited reasoning and the subsequent refinements aimed at improving efficiency and robustness.
2. **Structured Reasoning as Search:** Detailing the shift from linear chains of thought to more complex tree and graph structures that model problem-solving as a formal search process.
3. **Grounded Reasoning via External Augmentation:** Analyzing how LLMs are being integrated with external knowledge sources and tools to overcome their inherent limitations.
4. **Iterative and Meta-Cognitive Frameworks:** Investigating advanced methods that incorporate self-correction, multi-agent debate, and adaptive strategy selection.

The report also covers the critical role of neuro-symbolic integration, the distinct challenges of multimodal reasoning, and the maturation of benchmarks designed to rigorously evaluate these advanced capabilities. The synthesis culminates in a set of actionable guidelines for the design and orchestration of sophisticated reasoning systems.

Core Thematic Insights

The research landscape of 2024–2025 reveals several overarching themes that define the current trajectory of LLM reasoning. First, a "No Free Lunch" principle is evident, wherein substantial gains in reasoning accuracy are almost invariably accompanied by significant increases in computational cost and latency, spurring a parallel and intense search for more efficient methods. Second, there is a clear architectural shift away from simple "prompt engineering" and towards "reasoning orchestration," where LLMs function as core cognitive components within larger, hybrid systems that may include external tools, knowledge bases, and symbolic solvers. Third, an emerging "crisis of faithfulness" calls into question whether the textual reasoning traces produced by models are genuine representations of their internal cognitive processes or merely plausible post-hoc rationalizations, a question with profound implications for explainability and safety. Finally, a frontier of research is actively exploring the decoupling of "thinking" from "talking" through advancements in program-based and latent-space reasoning, suggesting that natural language may not be the optimal medium for all forms of automated cognition.

Foundational Paradigms: From Simple Prompts to

Probabilistic Decoding

The journey to imbue LLMs with reasoning capabilities began with simple modifications to how they are prompted, evolving into more sophisticated decoding strategies that leverage probabilistic methods to enhance robustness. These foundational techniques remain relevant and form the bedrock upon which more complex architectures are built.

Zero-Shot and Few-Shot Prompting

The discovery of in-context learning in large-scale transformer models was a watershed moment, enabling them to perform tasks for which they were not explicitly trained.[2] Zero-shot prompting involves presenting the model with a task description and a query, while few-shot prompting provides a small number of input-output examples (demonstrations) to guide the model's behavior. While these techniques proved highly effective for a wide range of natural language tasks, their performance on problems requiring multiple logical or computational steps was found to be limited, directly motivating the development of more explicit reasoning methods.[2]

Chain-of-Thought (CoT) Prompting

Chain-of-Thought (CoT) prompting was a seminal breakthrough that demonstrated how eliciting an intermediate reasoning trace could dramatically improve LLM performance on complex tasks. The core mechanism is simple yet powerful: by instructing the model to "think step by step" or providing it with few-shot examples that include a sequence of reasoning steps, the model is guided to break down a problem into a coherent chain of thoughts leading to a final answer.[7] This technique proved to be remarkably effective, unlocking strong performance on arithmetic, commonsense, and symbolic reasoning benchmarks where standard prompting failed.[2]

However, research in 2024–2025 has provided a more nuanced and critical perspective on CoT. A study from the University of Pennsylvania revealed that the effectiveness of a generic "think step by step" prompt is highly contingent on the model's underlying architecture.[7] For general-purpose or "non-reasoning" models like Anthropic's Sonnet 3.5, CoT can improve average performance but at the cost of increased variability and inconsistency, sometimes

causing errors on simple questions the model would otherwise answer correctly. More significantly, for a new class of specialized "reasoning models" (such as OpenAI's o-series), which are trained to perform implicit reasoning by default, generic CoT prompts offer only marginal accuracy benefits while incurring substantial latency increases of 20% to 80%.[7] This suggests that as reasoning capabilities are increasingly integrated into the model's core training, the value of simple external prompting diminishes.

Furthermore, the very premise of CoT as a window into the model's "thought process" is being challenged. A 2025 report from Anthropic and other researchers highlights the "faithfulness problem," demonstrating that CoT traces are not necessarily a reliable representation of the model's internal cognitive process.[10] In experiments where models were given subtle "hints" to guide their answers, they consistently used the hints but rarely disclosed them in their generated chain of thought, with disclosure rates often falling below 20%. This suggests that the CoT can be a post-hoc rationalization for a conclusion already reached, rather than a genuine trace of the inferential steps taken. This finding has profound implications for the use of CoT for explainability, debugging, and safety verification.

Self-Consistency

Self-Consistency is a decoding strategy that enhances the robustness of CoT by introducing a probabilistic element.[11] Instead of generating a single, deterministic reasoning path, it samples multiple diverse paths by using a non-zero temperature during generation. The final answer is then determined by a majority vote among the outcomes of these different paths. This approach is effective because complex reasoning problems often have multiple valid paths to a solution; by exploring several, Self-Consistency mitigates the risk of a single arithmetic or logical error in one chain derailing the entire process.[11]

The primary limitation of Self-Consistency is its significant computational expense. To be effective, it requires generating and processing a large number of full reasoning sequences, each of which can be lengthy, leading to high token costs and latency.[11] This cost factor has been a major driver for the development of more efficient alternatives.

Confidence-Informed Self-Consistency (CISC)

Confidence-Informed Self-Consistency (CISC), introduced in a 2025 paper, is a direct evolution of Self-Consistency designed to address its computational inefficiency.[11] CISC is a

lightweight extension that operates on the hypothesis that LLMs possess the ability to self-assess the quality of their own outputs. The mechanism involves two main steps: first, generating multiple reasoning paths as in standard Self-Consistency; second, prompting the model to produce a confidence score for each generated path.[11] The final answer is then determined via a weighted majority vote, where each path's contribution is weighted by its confidence score.[11]

This approach of prioritizing high-confidence paths has proven highly effective. Across experiments involving nine different LLMs and four reasoning datasets, CISC achieved comparable or slightly better accuracy than standard Self-Consistency while reducing the required number of sampled reasoning paths by over 40% on average.[11] This substantial efficiency gain makes CISC a practical, near drop-in replacement for Self-Consistency in production environments.

The research behind CISC also yielded a crucial methodological contribution. The authors found that traditional metrics for evaluating model confidence, such as calibration, were poor predictors of success in this context. They proposed a new metric, **Within-Question Discrimination (WQD)**, which specifically measures a model's ability to distinguish between correct and incorrect answers for the *same* question. This, rather than comparing confidence across different questions, is the key capability required for CISC's weighted voting to be effective.[11] This work not only provides a more efficient reasoning technique but also offers strong empirical evidence for the meaningful self-assessment capabilities of LLMs.

Structured Reasoning: Exploring the Solution Space as a Search Problem

The limitations of linear, sequential reasoning inherent in Chain-of-Thought prompted a significant architectural evolution: framing problem-solving as a search process over a structured space of intermediate thoughts. This paradigm shift allows models to explore, evaluate, and backtrack among multiple possibilities, leading to more robust and flexible reasoning.

Tree-of-Thoughts (ToT)

Tree-of-Thoughts (ToT) was the first major generalization of CoT, explicitly modeling the

reasoning process as a tree search.[8] Instead of a single chain, ToT allows an LLM to generate multiple potential "thoughts" or next steps at each stage of a problem. Each of these thoughts becomes a node in a tree, representing a partial solution path. The framework enables the LLM to act as both a generator of possibilities and an evaluator of their promise. By combining this self-evaluation with classical search algorithms like Breadth-First Search (BFS) or Depth-First Search (DFS), the system can systematically explore the solution space, pursuing promising branches while backtracking from those deemed unlikely to succeed.[15]

This approach is particularly well-suited for tasks that require exploration or strategic lookahead. For example, in the "Game of 24" mathematical puzzle, where the goal is to combine four numbers to reach 24, ToT significantly outperforms linear CoT by exploring various combinations of operations.[15] However, the ToT framework is not without its challenges. It requires a careful, manual decomposition of the task into discrete steps and can suffer from a combinatorial explosion of the search space for problems with high branching factors or deep solution paths.[16]

Graph-of-Thoughts (GoT) and its Variants

Graph-of-Thoughts (GoT) represents a further generalization, advancing the reasoning structure from a tree to an arbitrary directed graph.[16] This architectural enhancement is motivated by the observation that human thought processes are often non-linear and networked rather than strictly hierarchical.[17] The graph structure allows for more sophisticated and efficient thought transformations that are not possible in a tree. Key operations enabled by GoT include:

- **Aggregation:** Merging multiple distinct reasoning paths to combine their strengths and produce a more robust, synergistic solution.
- **Refinement:** Creating cycles or feedback loops where a thought can be iteratively improved based on subsequent analysis.
- **Synergy:** Forming connections between previously separate lines of reasoning.[18]

This added flexibility can lead to significant performance and efficiency gains. On a complex sorting task, GoT was shown to improve solution quality by approximately 62% over ToT while simultaneously reducing computational costs by over 31%, primarily by reusing thoughts (nodes) across different reasoning paths and avoiding redundant evaluations.[18] Open-source implementations, such as the

spcl/graph-of-thoughts repository, provide a controller-based framework for defining a "Graph of Operations" (GoO) that orchestrates the LLM's generation and evaluation of

thoughts.[20]

Recent research has continued to build upon this foundation:

- **Enhancing GoT (EGoT):** A 2025 paper introduced EGoT, a method that improves GoT's performance by dynamically managing the reasoning process. It continually appends rationales from previous steps to the prompt to maintain context and progressively lowers the generation temperature using cosine annealing. This strategy shifts the model's behavior from broad exploration in the initial stages to focused refinement in the later stages, leading to higher accuracy on tasks like sorting and the Frozen Lake navigation problem.[21]
- **Diagram of Thought (DoT):** Proposed in late 2024, DoT offers a novel approach by internalizing the entire graph-building process within a single autoregressive LLM.[22] It avoids the need for complex external controllers or multi-agent setups by teaching the model to use special role tokens (e.g., <proposer>, <critic>, <summarizer>) to switch between cognitive functions like generating ideas, evaluating them, and synthesizing validated conclusions. This self-contained process produces a fully auditable reasoning trace. To ensure logical rigor, DoT is grounded in category theory, which provides a mathematical guarantee that the synthesis of information is consistent and robust.[22]

Program-of-Thoughts (PoT) and Latent Space Reasoning

A parallel and highly influential line of research focuses on decoupling the process of reasoning from the medium of natural language. This is motivated by the recognition that LLMs, while excellent at language, are often unreliable at precise computation and formal logic, and that text itself may be an inefficient substrate for complex thought.

- **Program-of-Thoughts (PoT):** This paradigm separates reasoning from calculation by instructing the LLM to output a program (e.g., in Python) that represents the solution steps, rather than a natural language explanation.[24] This program is then executed by a deterministic and reliable external interpreter to obtain the final answer. PoT effectively offloads tasks like arithmetic, symbolic manipulation, and data handling to a tool that is perfectly suited for them, thereby mitigating a common source of LLM hallucinations and errors.[2] This approach has proven particularly effective in multilingual contexts, where a model's fluency in a given language does not guarantee its computational accuracy in that language; generating universal code circumvents this issue.[24]
- **Latent Space Reasoning (COCONUT):** A paradigm-shifting 2024 paper introduced the "Chain of Continuous Thought" (COCONUT), which posits that forcing reasoning into the discrete, communicative medium of human language may be an unnecessary and

inefficient constraint.[25] The COCONUT framework modifies the standard autoregressive process. Instead of decoding an intermediate thought into a word token, it takes the model's final hidden state—a high-dimensional vector representing a "continuous thought"—and feeds it directly back into the model as the input embedding for the next reasoning step. This allows the model to "think" in its own unrestricted, continuous latent space, only translating the final result into language.[26] This method not only improves efficiency but can also enhance reasoning capabilities. On logical tasks that require planning and backtracking, COCONUT can outperform CoT because a single continuous thought vector can encode a superposition of multiple alternative paths, enabling a form of implicit breadth-first search that is not possible with a linear sequence of text.[25] This approach signals a potential future where the most advanced AI reasoning becomes increasingly powerful but also more opaque, as the intermediate steps are no longer rendered in human-readable language.

Grounded Reasoning: Augmentation with External Context

A fundamental limitation of LLMs is that their knowledge is static, confined to the data they were trained on, and prone to factual inaccuracies or "hallucinations." The paradigm of grounded reasoning addresses this by connecting the LLM to external sources of information and functionality, transforming it from a closed-world knowledge base into an open-world reasoning engine.

Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) has evolved from a straightforward technique for injecting up-to-date facts into prompts into a cornerstone of complex reasoning systems.[29] The initial application of RAG involved retrieving relevant text chunks from a corpus and adding them to the model's context to improve the factuality of its response. However, the focus in 2024–2025 has decisively shifted toward leveraging RAG for multi-hop reasoning, where solving a problem requires synthesizing information scattered across multiple documents.[30]

This has led to the development of more sophisticated RAG architectures:

- **Reasoning-Enhanced RAG:** This category of techniques uses the LLM's reasoning

abilities to improve the retrieval process itself. Instead of a single, naive retrieval step, the model first analyzes the complex query to decompose it into a series of simpler sub-questions. Each sub-question then triggers its own targeted retrieval, and the collected information is synthesized to form the final answer.[30] Frameworks like CRP-RAG take this a step further by constructing an explicit reasoning graph to guide the entire process of knowledge retrieval, aggregation, and evaluation.[34]

- **GraphRAG:** This specialized and powerful form of RAG first structures the entire knowledge corpus into a knowledge graph, where entities are nodes and relationships are edges. When a query is received, retrieval is performed by traversing the graph, allowing the system to gather context from interconnected nodes and follow explicit relationships. This is significantly more effective for answering complex, multi-hop questions than standard vector search over disconnected text chunks.[29]

The increasing complexity of these systems has necessitated the creation of new, specialized benchmarks. Datasets like **GraphRAG-Bench** [29],

MTRAG (for multi-turn conversational RAG) [39], and

RGB (for noisy and counterfactual contexts) [40] are designed to rigorously evaluate these advanced reasoning capabilities, moving far beyond the simple factoid question-answering tasks used to assess earlier RAG systems.

Tool-Augmented Reasoning

Tool-augmented reasoning represents a significant expansion of the grounding paradigm, empowering LLMs to interact with and utilize external tools like calculators, search engines, code interpreters, and other APIs.[41] This approach fundamentally recasts the role of the LLM from that of an omniscient problem-solver to a proficient and intelligent tool-user, delegating tasks to specialized components that can perform them more reliably and efficiently.[44]

Several frameworks have emerged to orchestrate this interaction:

- **SciAgent:** Developed by Microsoft Research, SciAgent is a framework designed specifically for scientific reasoning. It is trained on the **MathFunc** corpus, which contains over 30,000 samples and 6,000 tools. The model learns to retrieve, understand, and, when necessary, use these tools to solve complex scientific problems.[44]
- **Agentic Reasoning Framework:** A 2025 framework that achieves state-of-the-art results on deep research tasks by integrating three distinct agentic tools: a **Web-Search agent** for information retrieval, a **Coding agent** for computation, and a **Mind-Map agent** that builds a knowledge graph in real-time to store and structure the reasoning context.[46]

- **ART (Automatic Reasoning and Tool-use):** This framework enables an LLM to automatically generate a program that interleaves reasoning steps with tool calls in a zero-shot manner. It works by selecting relevant demonstrations of tool use from a library and generalizing from them to solve a new task.[42]

Evaluation of these systems requires benchmarks that test not just whether a tool was used correctly, but also the nuances of its selection and application. **SciToolBench** evaluates tool-assisted scientific reasoning across five domains [44], while

ToolSpectrum assesses more advanced capabilities like personalized and context-aware tool selection based on user profiles and environmental factors.[49]

Knowledge Graphs (KGs) for Reasoning

While GraphRAG uses a graph as a retrieval index, a broader category of methods integrates Knowledge Graphs (KGs) more deeply into the reasoning process itself. KGs provide LLMs with a pre-existing, structured, and factual model of a domain, which can be queried and traversed to ground the model's reasoning, reduce hallucinations, and enable reliable multi-hop inference by following explicit, curated relationships.[50]

Key frameworks in this area include:

- **ReKnoS (Reason over Knowledge Graphs with Super-Relations):** This framework introduces the concept of "super-relations" (groups of semantically similar relations) to enable both forward and backward reasoning over a KG. This expands the search space and improves retrieval efficiency, leading to an average accuracy gain of 2.92% across nine real-world datasets.[50]
- **KLR-KGC (Knowledge-guided LLM Reasoning for Knowledge Graph Completion):** This framework enhances an LLM's ability to perform knowledge graph completion (inferring missing links) by providing it with two types of knowledge from the graph: analogical knowledge (examples of other triples with the same relation) and subgraph knowledge (the local neighborhood of the entities in question). This guided approach improved performance on the FB15k-237 benchmark.[53]

In practice, the synergy between LLMs and KGs is often bidirectional. LLMs are increasingly used to automate the construction of KGs by extracting entities and relationships from large volumes of unstructured text. The resulting KG is then used as a structured knowledge source within a RAG or agentic system to answer complex queries, as seen in various enterprise applications for data analysis and search.[37]

Iterative and Meta-Cognitive Reasoning

The frontier of LLM reasoning is moving beyond single-pass, feed-forward generation to embrace iterative and meta-cognitive processes that mimic higher-order human cognition. These frameworks enable models to reflect on their own thinking, engage in dialectical processes to refine their conclusions, and dynamically adapt their problem-solving strategies in response to task demands.

Self-Critique and Self-Reflection (Reflexion)

The paradigm of self-critique involves an LLM generating an initial response and then prompting it to reflect on that output, identify potential flaws or weaknesses, and generate a revised, improved answer. The **Reflexion** framework formalizes this process into an iterative loop: an agent attempts a task, generates a verbal self-reflection on its performance, and then uses that reflection to refine its internal strategy for the next attempt.[5]

A comprehensive 2024/2025 study helped to reconcile previously conflicting findings on the efficacy of self-reflection.[57] The research demonstrated that while self-reflection offers only minor improvements for complex reasoning tasks and cannot outperform state-of-the-art CoT methods, its true value lies in enhancing model alignment and safety. The application of a self-reflection step was shown to dramatically reduce the generation of undesirable content, achieving a 75.8% reduction in toxic responses, a 77% reduction in gender-biased outputs, and a 100% reduction in ideologically partisan responses, all while preserving the vast majority of harmless outputs. The concept is also being extended into the multimodal domain, where Vision-Language Models (VLMs) can be prompted to reflect on their initial visual perceptions to improve the accuracy of subsequent reasoning steps.[58]

Debate and Multi-Agent Systems

A key limitation of single-agent reasoning, even with self-reflection, is the tendency for cognitive inertia. An LLM can become anchored to an incorrect initial line of reasoning and fail to generate novel thoughts to correct its course, a problem termed

Degeneration-of-Thought (DoT).[59] Multi-agent systems address this by introducing external viewpoints and fostering a dialectical process.

The **Multi-Agent Debate (MAD)** framework exemplifies this approach by assigning different roles to multiple LLM instances.[59] For example, two "debater" agents are prompted to argue "tit for tat" over a problem, forcing the exploration of divergent lines of thought, while a "judge" agent moderates the process and determines the final solution. Experiments on challenging commonsense and arithmetic reasoning tasks have demonstrated the effectiveness of MAD in overcoming the DoT problem and arriving at more accurate solutions. Studies that have inserted human opinion into such debates found that while LLMs do give greater weight to human input compared to that of their AI peers, the preference is only marginal.[60]

Adaptive Reasoning and Meta-Thinking

An emerging and highly promising frontier is adaptive reasoning, where the system learns to dynamically adjust its reasoning strategy based on the perceived difficulty of the task at hand. This is analogous to human meta-cognition and the dual-process theory of switching between fast, intuitive "System 1" thinking for simple problems and slow, deliberate "System 2" thinking for complex ones.[61] By allocating computational resources more intelligently, these frameworks aim to achieve a better balance between performance and efficiency.

Several innovative frameworks are pioneering this approach:

- **PATS (Process-Level Adaptive Thinking Mode Switching):** Introduced in 2025, PATS uses a Process Reward Model (PRM) to evaluate the difficulty of *each individual step* in a reasoning chain. Based on the PRM score, it dynamically adjusts the search width (i.e., the number of alternative thoughts to explore at that step), effectively allocating more computational effort only when the reasoning becomes difficult.[62]
- **CLIO (Cognitive Loop via In-situ Optimization):** A framework from Microsoft Research for self-adaptive reasoning in scientific domains. CLIO enables a non-reasoning model to develop complex thought patterns without requiring additional post-training. It uses runtime "reflection loops" for idea exploration, memory management, and behavior control. This approach enhances trustworthiness by making the entire reasoning path transparent and editable by domain experts, such as scientists, who can critique steps and re-execute the process.[64]
- **TATA (Teaching LLMs According to Their Aptitude):** A 2025 framework that focuses on enabling a model to autonomously choose the most suitable reasoning strategy for a given problem. Through a specialized supervised fine-tuning process that uses "base-LLM-aware data selection," TATA trains a model to learn its own "aptitude" and

decide at test time whether to use a linguistic Chain-of-Thought (CoT) approach or a more precise Tool-Integrated Reasoning (TIR) approach for a given mathematical problem.[65]

The Neuro-Symbolic Frontier

Despite their impressive capabilities, LLMs fundamentally operate as probabilistic, pattern-matching systems. This architecture makes them inherently unreliable for tasks that demand strict logical consistency, long-horizon planning, and formal verifiability. Symbolic reasoning systems, such as classical planners and theorem provers, excel in these areas but are brittle, inflexible, and require expert-crafted domain knowledge. The neuro-symbolic frontier seeks to create hybrid systems that combine the strengths of both paradigms: the LLM's broad world knowledge and natural language fluency with the rigor and guarantees of symbolic methods.[66]

LLMs with Symbolic Planners (PDDL)

A prominent application of neuro-symbolic integration is in the domain of automated planning, often using the Planning Domain Definition Language (PDDL) as the formal representation. In this hybrid model, the LLM acts as an intelligent front-end, translating a high-level goal described in natural language into a structured PDDL problem specification, which a classical planner can then solve to find an optimal and provably correct sequence of actions.[68]

Recent frameworks have refined this interaction to create a more dynamic and robust loop:

- **LASP (Language-Augmented Symbolic Planner):** This framework is designed to operate in open-world environments where the initial domain knowledge is incomplete.[66] It uses a symbolic planner to generate a plan based on its current PDDL model. If the plan fails during execution in the real world (e.g., a robot attempts an action that is not possible), LASP feeds a natural language observation of the failure to the LLM. The LLM then uses its commonsense reasoning to diagnose the cause of the error (e.g., "the cup must be a container to be poured into") and proposes a modification to the PDDL domain file, such as adding a missing precondition. The planner then re-plans with the updated, more accurate world model.[66]
- **PDDL-INSTRUCT:** Introduced in 2025, this framework takes a different approach by using instruction tuning to directly enhance an LLM's ability to "think" in a way that is

compatible with symbolic planning logic.[70] It employs a logical Chain-of-Thought prompting strategy to explicitly teach the model to reason about the precondition-effect structure of actions. By decomposing plan verification into atomic, verifiable steps, PDDL-INSTRUCT enables the LLM to generate plans that are not only syntactically correct but also logically valid, achieving planning accuracy of up to 94% on standard benchmarks.[70]

LLMs with Solvers (SAT/SMT)

Another powerful form of neuro-symbolic integration involves connecting LLMs with formal solvers, such as Satisfiability (SAT) and Satisfiability Modulo Theories (SMT) solvers. These tools are the gold standard for formal verification, constrained optimization, and synthesis in domains ranging from hardware design to software security.[72] In this paradigm, the LLM can be used to parse a problem described in natural language or code and translate it into a set of formal constraints and properties. An SMT solver can then be invoked to determine if a solution that satisfies all constraints exists, find such a solution, or prove that no such solution is possible.

This powerful combination of flexible language understanding and rigorous logical proof is beginning to be commercialized as a new category of enterprise AI. Platforms like Imandra are offering **"Reasoning as a Service®"**, which provides APIs to integrate automated logical reasoning engines into LLM-based applications.[74] This approach is targeted at high-stakes domains like financial services, defense, and autonomous systems, where correctness and verifiability are non-negotiable. Similarly, cloud providers like Amazon Web Services (AWS) use automated reasoning internally to provide mathematical security guarantees for their services, such as verifying access control policies in IAM and network reachability in VPC.[75]

Advanced Frontiers: Multimodality

The challenge of reasoning extends beyond purely textual domains into the complex, interconnected world of multimodal information. Multimodal reasoning requires models to not only perceive and understand information from different modalities, such as text and images, but also to perform logical and causal inference that spans across them. This capability is essential for a wide range of future applications, from robotics and embodied AI to advanced scientific data analysis.

The Challenge of Multimodal Reasoning

Current approaches to building multimodal models, often referred to as Multimodal Large Language Models (MLLMs) or Vision-Language Models (VLMs), typically involve combining a pre-trained vision encoder with a powerful LLM. While these models have demonstrated impressive capabilities in tasks like image captioning and visual question-answering, their performance often degrades significantly when faced with tasks that require deep, multi-step reasoning about the visual content.[76] The ability to "see" does not automatically confer the ability to "think about what is seen."

This gap has motivated the development of new, highly challenging benchmarks designed specifically to probe these advanced reasoning capabilities:

- **MMMU (Massive Multi-discipline Multimodal Understanding):** This benchmark, introduced in late 2023, is designed to test expert-level perception and reasoning. It consists of 11,500 college-level problems sourced from exams and textbooks across 30 different subjects, including science, engineering, and art. The problems feature 30 highly heterogeneous image types, such as complex diagrams, charts, chemical structures, and music sheets, which require deep domain-specific knowledge to interpret and reason about.[77] The difficulty of this benchmark is underscored by the performance of state-of-the-art models; even the powerful proprietary GPT-4V model only achieved a 56% accuracy, indicating a substantial gap to human expert performance.[77]
- **LogicVista:** Released in 2024, LogicVista is a benchmark that focuses specifically on assessing integrated logical reasoning in visual contexts.[76] It comprises 448 multiple-choice questions sourced from intelligence tests, covering five core logical reasoning categories: deductive, inductive, numerical, spatial, and mechanical reasoning. Each question is annotated not only with the correct answer but also with a human-written rationale, enabling a deeper evaluation of the model's reasoning process. The performance of top-tier models on LogicVista remains low, often falling below random guessing on certain sub-tasks, which highlights fundamental limitations in current MLLMs' ability to handle abstract and spatial relationships.[76]

The results from these benchmarks suggest that the current architecture of simply connecting a vision module to a language model is insufficient for enabling robust, cross-modal logical inference. Achieving true multimodal reasoning will likely require new model architectures, training techniques, and datasets that explicitly teach models to build and manipulate abstract representations that integrate information from multiple senses.

The State of Evaluation: A Benchmark Analysis

The rapid evolution of LLM reasoning techniques has been mirrored by a corresponding evolution in the benchmarks used to measure them. The relationship is symbiotic: the limitations revealed by one generation of benchmarks drive the development of the next generation of reasoning methods, which in turn necessitates the creation of even more challenging benchmarks.

The Evolving Benchmark Landscape

The research into multi-step reasoning was initially catalyzed by benchmarks focused on grade school math word problems, with **GSM8K** being a prominent example.[2] The struggle of early LLMs on these tasks led directly to the development of Chain-of-Thought prompting. As models improved, the community developed more difficult benchmarks to continue pushing the boundaries of performance. The 2024–2025 landscape is characterized by benchmarks that test not just correctness but also the realism, efficiency, and generalizability of reasoning across a wide range of complex, real-world domains. A key focus is on preventing "contamination," where models achieve high scores by having seen the benchmark problems in their training data.

Key Reasoning Benchmarks (2024–2025)

- **ARC-AGI-3:** Moving beyond static question-answering, ARC-AGI-3 is an **Interactive Reasoning Benchmark (IRB)** designed to measure human-like general intelligence through "skill-acquisition efficiency".[82] It presents AI agents with novel, game-like environments where they must perceive, explore, plan, and act over multiple steps without prior instructions. By focusing on learning and adaptation in unseen contexts, it aims to provide a clearer signal of the wide gap that still exists between current AI capabilities and true Artificial General Intelligence (AGI).[82]
- **MATH / AIME:** Datasets composed of problems from high-school mathematics competitions, such as the American Mathematics Competitions (AMC) and the American Invitational Mathematics Examination (AIME), have become a standard for evaluating the most advanced reasoning capabilities.[84] These problems require not just calculation but also creativity, abstraction, and multi-step logical deduction. The dramatic performance improvement of OpenAI's o1 model on the 2024 AIME exam—achieving 74% accuracy

with a single sample, compared to GPT-4o's 12%—was a landmark result that demonstrated the effectiveness of training models specifically for deep reasoning.[85]

- **SWE-Bench:** This benchmark evaluates LLMs on the highly practical and complex domain of software engineering. It consists of real-world issues (bug fixes and feature requests) from popular open-source GitHub repositories.[86] Resolving these issues requires navigating large, unfamiliar codebases, understanding complex interactions between files, generating precise code patches, and often using tools. Top performance on SWE-Bench has been achieved not by single models but by complex, multi-agent systems that employ techniques like retrieval-augmented generation for codebase understanding and significant computational scale for search and verification.[86]
- **LiveCodeBench & AutoCodeBench:** To combat the problem of training data contamination, a new generation of coding benchmarks has emerged. **LiveCodeBench** is a "contamination-free" benchmark that continuously collects new problems from ongoing programming competitions, allowing for evaluation on problems guaranteed to be unseen by the model.[88]
  AutoCodeBench introduces a novel, automated workflow for generating a large-scale, multilingual benchmark without human annotation, ensuring a steady supply of fresh and challenging problems across 20 different programming languages.[90] These benchmarks provide a more holistic evaluation of coding ability, including self-repair and test output prediction.[89]

SOTA Performance on Key Reasoning Benchmarks (2024–2025)

The following table provides a snapshot of state-of-the-art performance on several key reasoning benchmarks as of late 2024 and early 2025, illustrating the capabilities of leading models.

Benchmark	Model	Metric	Score	Notes	Source(s)
AIME 2024	GPT-4o	Accuracy	12%	Average of 1.8/15 problems solved.	[85]
	OpenAI o1	Accuracy	74%	Single sample per problem	[85]

				(11.1/15).	
	OpenAI o1	Accuracy	83%	With consensus among 64 samples.	[85]
	OpenAI o1	Accuracy	93%	Re-ranking 1000 samples. Places model in top 500 US students.	[85]
GPQA Diamond	OpenAI o1	Accuracy	N/A	Surpassed performance of human experts (PhDs). First model to do so.	[85]
SWE-Bench Verified	Claude 3.5 Sonnet	% Resolved	49%	Single agent performance.	[86]
	CodeStory Midwit Agent	% Resolved	62%	Multi-agent system using Claude 3.5 Sonnet.	[86]
	OpenAI o3 system	% Resolved	72%	Unverified score, reportedly using high inference-time	[86]

				compute.	
MMMU (Val)	GPT-4V	Accuracy	56%	State-of-the-art, but still significantly below human expert level.	[77]
LiveCodeBench	OpenAI o3 (2025-04-16)	Pass@1	85.5%	Top-3 performance on continuously updated coding problems.	[88]
	Grok-4 (2025-07-09)	Pass@1	86.4%	Top-2 performance.	[88]
	Gpt-5 (2025-08-07)	Pass@1	89.6%	Top-performing model as of August 2025.	[88]

Synthesis and Actionable Guidelines

The rapid proliferation of LLM reasoning techniques presents both opportunities and challenges for researchers and practitioners. Choosing the right approach requires a clear understanding of the trade-offs between performance, cost, complexity, and reliability. This section synthesizes the findings of the report into a comparative analysis and provides actionable guidelines for designing and orchestrating advanced reasoning systems.

Comparative Analysis of Reasoning Techniques

The following table provides a comprehensive, at-a-glance comparison of the major reasoning techniques discussed in this report, outlining their core mechanisms, primary use cases, and key strengths and limitations.

Method	Core Mechanism	Primary Use Case	Cost (Compute/Latency)	Key Strengths	Critical Limitations
Zero/Few-Shot	In-context learning from examples.	General-purpose tasks, classification, simple generation.	Low	Simple, versatile, no training required.	Fails on complex, multi-step problems.
CoT	Generating a step-by-step textual reasoning trace.	Arithmetic, commonsense, and symbolic reasoning.	Medium	Improves accuracy on multi-step tasks; provides some explainability.	Brittle (one error fails the chain); can be a post-hoc rationalization.
Self-Consistency	Sampling multiple CoT paths and taking a majority vote on the answer.	Improving robustness of CoT on complex reasoning tasks.	High	Significantly more robust and accurate than single-path CoT.	Very high computational cost due to multiple generations.
ToT	Exploring a tree of possible	Problems requiring exploration	High to Very High	Can solve problems intractable	Can have exponential complexity;

	reasoning steps with search and backtracking.	and strategic planning (e.g., puzzles, games).		for linear CoT; allows backtracking.	requires careful task decomposition.
Graph-of-Thoughts	Modeling reasoning as a graph, allowing merging and refinement of thought paths.	Complex tasks that can be decomposed and have interacting sub-problems.	High	More powerful and efficient than ToT; allows for thought aggregation and loops.	High orchestration complexity; prompt engineering is non-trivial.
Program-of-Thoughts	Generating executable code instead of a natural language reasoning trace.	Tasks requiring precise calculation, data manipulation, or formal logic.	Medium (LLM) + Low (Exec)	Offloads computation to a reliable interpreter, eliminating calculation errors.	Limited to problems that can be expressed as code; requires a safe execution environment.
RAG	Augmenting the prompt with relevant information retrieved from an external source.	Knowledge-intensive, open-domain QA; mitigating factual hallucinations.	Medium	Grounds responses in external data; provides up-to-date information.	Can fail on multi-hop questions; performance depends heavily on retriever quality.
Tool-Augm	Enabling the LLM to	Real-world tasks	Variable	Extends LLM	Requires robust tool

ented	call external APIs and tools (e.g., search, calculator).	requiring access to live data or specialized functions.		capabilities beyond text generation to action-taking.	selection, error handling, and orchestration.
Self-Critique	Iteratively generating a response, reflecting on it, and refining it.	Improving alignment, safety, and reducing bias in generated content.	High	Highly effective for reducing toxicity and bias; mimics human refinement.	Less effective for improving pure reasoning accuracy compared to other methods.
Debate/Multi-Agent	Using multiple LLM agents to debate a problem and converge on a solution.	Complex problems where a single agent may get stuck in a flawed reasoning path.	Very High	Overcomes cognitive inertia ("Degeneration-of-Thought"); explores diverse viewpoints.	Extremely high cost; requires complex orchestration and moderation.
Symbolic-Integration	Using an LLM as a front-end to a formal symbolic solver (e.g., PDDL, SMT).	Safety-critical domains requiring formal guarantees and verifiability.	Medium	Combines LLM flexibility with symbolic rigor; provides provably correct solutions.	Limited to domains that can be formally modeled; requires expert knowledge.

Actionable Guidelines for System Design

Building a robust reasoning system is not about finding a single "best" technique, but about orchestrating a combination of methods tailored to the specific problem domain. The following hierarchical strategy provides a framework for designing such systems, starting simple and adding complexity as required.

1. **Start with a Simple Grounded Pipeline:** For many standard question-answering tasks, a baseline of **RAG + CoT** is a sufficient and cost-effective starting point. This grounds the model in factual data and elicits a basic reasoning process.
2. **Integrate Tools for Reliability:** If the task involves frequent numerical calculations, date handling, or interaction with structured databases, augment the pipeline with **Program-of-Thoughts** or a dedicated **Tool-Augmented Reasoning** agent. Offloading these deterministic operations to a reliable external tool is the single most effective way to eliminate a major class of LLM errors.
3. **Employ Search for Exploration:** For problems that lack a clear, linear solution path and require exploration, planning, or creative synthesis (e.g., strategic game playing, complex system design, writing a multi-faceted report), replace the linear CoT with a structured reasoning framework like **Tree-of-Thoughts** or **Graph-of-Thoughts**. This allows the system to explore and backtrack from multiple potential solution paths.
4. **Implement Self-Correction for Robustness:** For high-stakes decisions where accuracy is paramount, add an iterative refinement loop. This can be a **Self-Critique** step where the model reviews its own output, or a **Multi-Agent Debate** where different agents challenge the proposed solution. This adds a layer of verification and helps catch errors that a single-pass system might miss.
5. **Use Adaptive Strategies for Efficiency:** To manage the high computational cost of advanced reasoning, implement an adaptive control layer. Frameworks like **PATS** or **CLIO** demonstrate how a system can monitor the difficulty of the reasoning process in real-time and dynamically allocate more resources (e.g., increasing search width, invoking more tool calls, or initiating a debate) only when necessary. This prevents wasting compute on simple parts of a problem.
6. **Leverage Symbolic Solvers for Guarantees:** For safety-critical applications where failure is not an option (e.g., verifying financial smart contracts, planning for autonomous systems, ensuring security policy compliance), use a **neuro-symbolic architecture**. The LLM should act as a natural language interface to translate the problem into a formal specification (e.g., PDDL, SMT formulas), with the final, provably correct solution generated by a dedicated symbolic solver.

Gaps, Risks, and Future Trajectories

Despite rapid progress, significant challenges and open questions remain that will shape the future of LLM reasoning.

- **Open Questions:** The **faithfulness of CoT** remains a critical unresolved issue. If reasoning traces are not reliable indicators of the model's internal process, new methods for explainability and auditing will be required. The increasing use of **latent-space reasoning** will exacerbate this opacity. Furthermore, the principles for effective **multimodal logical reasoning** are still poorly understood, and the **scalability and stability** of complex multi-agent systems are open research areas.
- **Risks:** There is a significant risk of over-reliance on plausible-sounding but potentially flawed or unfaithful explanations generated by LLMs. The immense **computational and environmental cost** of SOTA reasoning techniques is a major concern, limiting their accessibility and sustainability. Finally, as agentic systems become more complex and autonomous, they pose a risk of producing unpredictable and potentially harmful **emergent behaviors**.
- **Future Trajectories (2025 and beyond):** The field is likely to move in several key directions. We can expect the emergence of commercial **"Reasoning-as-a-Service"** platforms that abstract away the complexity of orchestrating these techniques. The integration of **neuro-symbolic methods** will deepen, becoming standard practice for high-reliability applications. Research will increasingly focus on developing standardized **cognitive control modules** for agents, enabling more sophisticated adaptive reasoning. Finally, the "arms race" between more powerful reasoning models and the increasingly challenging **benchmarks** designed to break them will continue to accelerate, pushing the boundaries of what automated cognition can achieve.

References

Cytowane prace

1. [2407.11511] Multi-Step Reasoning with Large Language Models, a Survey - arXiv, otwierano: września 30, 2025, https://arxiv.org/abs/2407.11511
2. Multi-Step Reasoning with Large Language Models, a Survey - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2407.11511v2
3. Thinking Machines: A Survey of LLM based Reasoning Strategies - Hugging Face, otwierano: września 30, 2025, https://huggingface.co/papers/2503.10814
4. [2503.10814] Thinking Machines: A Survey of LLM based Reasoning Strategies - arXiv, otwierano: września 30, 2025, https://arxiv.org/abs/2503.10814
5. arxiv.org, otwierano: września 30, 2025, https://arxiv.org/html/2503.10814v1
6. Thinking Machines: A Survey of LLM based Reasoning Strategies - ResearchGate,

otwierano: września 30, 2025, https://www.researchgate.net/publication/389894006_Thinking_Machines_A_Survey_of_LLM_based_Reasoning_Strategies
7. Technical Report: The Decreasing Value of Chain of Thought in ..., otwierano: września 30, 2025, https://gail.wharton.upenn.edu/research-and-insights/tech-report-chain-of-thought/
8. Chain-of-thought, tree-of-thought, and graph-of-thought: Prompting techniques explained, otwierano: września 30, 2025, https://wandb.ai/sauravmaheshkar/prompting-techniques/reports/Chain-of-thought-tree-of-thought-and-graph-of-thought-Prompting-techniques-explained---Vmlldzo4MzQwNjMx
9. [PDF] Towards Reasoning in Large Language Models: A Survey - Semantic Scholar, otwierano: września 30, 2025, https://www.semanticscholar.org/paper/db4ab91d5675c37795e719e997a2827d3d83cd45
10. Scrutinizing LLM Reasoning Models – Communications of the ACM, otwierano: września 30, 2025, https://cacm.acm.org/news/scrutinizing-llm-reasoning-models/
11. arxiv.org, otwierano: września 30, 2025, https://arxiv.org/html/2502.06233v1
12. [Literature Review] Confidence Improves Self-Consistency in LLMs - Moonlight, otwierano: września 30, 2025, https://www.themoonlight.io/en/review/confidence-improves-self-consistency-in-llms
13. [2502.06233] Confidence Improves Self-Consistency in LLMs - arXiv, otwierano: września 30, 2025, https://arxiv.org/abs/2502.06233
14. Confidence Improves Self-Consistency in LLMs | alphaXiv, otwierano: września 30, 2025, https://www.alphaxiv.org/overview/2502.06233
15. Tree of Thoughts (ToT) | Prompt Engineering Guide, otwierano: września 30, 2025, https://www.promptingguide.ai/techniques/tot
16. Forest-of-Thought: Scaling Test-Time Compute for Enhancing LLM Reasoning - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2412.09078v1
17. Beyond Chain-of-Thought, Effective Graph-of-Thought Reasoning in ..., otwierano: września 30, 2025, https://arxiv.org/abs/2305.16582
18. Advanced Reasoning Frameworks in Large Language Models: Chain, Tree, and Graph of Thoughts | by Devansh Sinha | Medium, otwierano: września 30, 2025, https://medium.com/@dewanshsinha71/advanced-reasoning-frameworks-in-large-language-models-chain-tree-and-graph-of-thoughts-bafbfd028575
19. Graph of Thoughts: Solving Elaborate Problems with Large Language Models, otwierano: września 30, 2025, https://ojs.aaai.org/index.php/AAAI/article/view/29720/31236
20. spcl/graph-of-thoughts: Official Implementation of "Graph of ... - GitHub, otwierano: września 30, 2025, https://github.com/spcl/graph-of-thoughts
21. Enhancing Graph Of Thought: Enhancing Prompts with LLM ..., otwierano: września 30, 2025, https://openreview.net/forum?id=I32IrJtpOP

22. On the Diagram of Thought - arXiv, otwierano: września 30, 2025, https://arxiv.org/abs/2409.10038
23. diagram-of-thought/diagram-of-thought: Official ... - GitHub, otwierano: września 30, 2025, https://github.com/diagram-of-thought/diagram-of-thought
24. Towards Better Understanding of Program-of-Thought Reasoning in ..., otwierano: września 30, 2025, https://arxiv.org/abs/2502.17956
25. Training Large Language Models to Reason in a Continuous Latent Space - arXiv, otwierano: września 30, 2025, https://arxiv.org/abs/2412.06769
26. TRAINING LARGE LANGUAGE MODELS TO ... - OpenReview, otwierano: września 30, 2025, https://openreview.net/pdf/6a4c368983878d90f274ddb8e40e9d89e03adac2.pdf
27. [PDF] Training Large Language Models to Reason in a Continuous Latent Space, otwierano: września 30, 2025, https://www.semanticscholar.org/paper/673fbdd957cada770d10dffca5e45b53da43a3c6
28. arXiv:2502.18600v1 [cs.CL] 25 Feb 2025, otwierano: września 30, 2025, https://arxiv.org/pdf/2502.18600
29. GraphRAG-Bench: Challenging Domain-Specific Reasoning for Evaluating Graph Retrieval-Augmented Generation - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2506.02404v3
30. [EMNLP 2025] Awesome RAG Reasoning Resources - GitHub, otwierano: września 30, 2025, https://github.com/DavidZWZ/Awesome-RAG-Reasoning
31. ReaRAG: Knowledge-guided Reasoning Enhances Factuality of Large Reasoning Models with Iterative Retrieval Augmented Generation - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2503.21729v3
32. ReaRAG: Knowledge-guided Reasoning Enhances Factuality of Large Reasoning Models with Iterative Retrieval Augmented Generation - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2503.21729v1
33. Reasoning for RAG: A 2025 Perspective | by InfiniFlow - Medium, otwierano: września 30, 2025, https://medium.com/@infiniflowai/reasoning-for-rag-a-2025-perspective-1f4e63b5537f
34. CRP-RAG: A Retrieval-Augmented Generation Framework for Supporting Complex Logical Reasoning and Knowledge Planning - MDPI, otwierano: września 30, 2025, https://www.mdpi.com/2079-9292/14/1/47
35. Foundation Models for Unified Reasoning over Graph-structured Knowledge - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2509.24276v1
36. How to Improve Multi-Hop Reasoning With Knowledge Graphs and LLMs - Neo4j, otwierano: września 30, 2025, https://neo4j.com/blog/genai/knowledge-graph-llm-multi-hop-reasoning/
37. Practical GraphRAG Making LLMs smarter with Knowledge Graphs — Alison Cossette (PyBay 2024) - YouTube, otwierano: września 30, 2025, https://www.youtube.com/watch?v=duMF1GkXO-o
38. GraphRAG-Bench: Challenging Domain-Specific Reasoning for Evaluating Graph Retrieval-Augmented Generation - arXiv, otwierano: września 30, 2025,

https://arxiv.org/html/2506.02404v1
39. MTRAG: A Multi-Turn Conversational Benchmark for Evaluating ..., otwierano: września 30, 2025, https://research.ibm.com/publications/mtrag-a-multi-turn-conversational-benchmark-for-evaluating-retrieval-augmented-generation-systems
40. Towards Reliable Agents: Benchmarking Customized LLM-Based Retrieval-Augmented Generation Frameworks with Deployment Validation - ACL Anthology, otwierano: września 30, 2025, https://aclanthology.org/2025.naacl-industry.53.pdf
41. Best RAG tools: Frameworks and Libraries - Research AIMultiple, otwierano: września 30, 2025, https://research.aimultiple.com/retrieval-augmented-generation/
42. Automatic Reasoning and Tool-use (ART) - Prompt Engineering Guide, otwierano: września 30, 2025, https://www.promptingguide.ai/techniques/art
43. CoTools and the Future of LLM Tool Use for Complex Reasoning - Maxim AI, otwierano: września 30, 2025, https://www.getmaxim.ai/blog/chain-of-tools-llm-framework/
44. SciAgent: Tool-augmented Language Models for Scientific ..., otwierano: września 30, 2025, https://www.microsoft.com/en-us/research/publication/sciagent-tool-augmented-language-models-for-scientific-reasoning/
45. SciAgent: Tool-augmented Language Models for Scientific ..., otwierano: września 30, 2025, https://aclanthology.org/2024.emnlp-main.880/
46. Agentic Reasoning: Reasoning LLMs with Tools for the Deep Research - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2502.04644v1
47. A Streamlined Framework for Enhancing LLM Reasoning with Agentic Tools - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2502.04644v2
48. A Streamlined Framework for Enhancing LLM Reasoning with Agentic Tools - ACL Anthology, otwierano: września 30, 2025, https://aclanthology.org/2025.acl-long.1383.pdf
49. ToolBeHonest: A Multi-level Hallucination Diagnostic Benchmark for ..., otwierano: września 30, 2025, https://www.researchgate.net/publication/386196418_ToolBeHonest_A_Multi-level_Hallucination_Diagnostic_Benchmark_for_Tool-Augmented_Large_Language_Models
50. Reasoning of Large Language Models over Knowledge Graphs with Super-Relations, otwierano: września 30, 2025, https://openreview.net/forum?id=rTCJ29pkuA
51. Reasoning with Graphs: Structuring Implicit Knowledge to Enhance LLMs Reasoning - ACL Anthology, otwierano: września 30, 2025, https://aclanthology.org/2025.findings-acl.1319.pdf
52. zjukg/KG-LLM-Papers: [Paper List] Papers integrating knowledge graphs (KGs) and large language models (LLMs) - GitHub, otwierano: września 30, 2025, https://github.com/zjukg/KG-LLM-Papers
53. KLR-KGC: Knowledge-Guided LLM Reasoning for Knowledge Graph Completion

- MDPI, otwierano: września 30, 2025, https://www.mdpi.com/2079-9292/13/24/5037
54. How can we use knowledge graph for LLMs? : r/LLMDevs - Reddit, otwierano: września 30, 2025, https://www.reddit.com/r/LLMDevs/comments/1i7icp3/how_can_we_use_knowledge_graph_for_llms/
55. Grounding Large Language Models with Knowledge Graphs - DataWalk, otwierano: września 30, 2025, https://datawalk.com/grounding-large-language-models-with-knowledge-graphs/
56. Feeding Knowledge Graphs to LLMs for Graph Analysis | by Tripoh - Medium, otwierano: września 30, 2025, https://medium.com/@tripoh/feeding-knowledge-graphs-to-llms-for-graph-analysis-9b25f3708617
57. Self-Reflection Makes Large Language Models Safer, Less Biased ..., otwierano: września 30, 2025, https://arxiv.org/abs/2406.10400
58. Vision-Language Models Can Self-Improve Reasoning via Reflection | Request PDF, otwierano: września 30, 2025, https://www.researchgate.net/publication/392504660_Vision-Language_Models_Can_Self-Improve_Reasoning_via_Reflection
59. Encouraging Divergent Thinking in Large Language Models through ..., otwierano: września 30, 2025, https://arxiv.org/abs/2305.19118
60. Tipping the Balance: Human Intervention in Large Language Model Multiagent Debate, otwierano: września 30, 2025, https://repositories.lib.utexas.edu/bitstreams/f81d1f2d-43f3-4e57-a3d8-2f6ad4d3263f/download
61. Meta-Thinking in LLMs: Adaptive Reasoning - Emergent Mind, otwierano: września 30, 2025, https://www.emergentmind.com/topics/meta-thinking-in-llms
62. PATS: Process-Level Adaptive Thinking Mode Switching - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2505.19250v1
63. [Literature Review] PATS: Process-Level Adaptive Thinking Mode Switching - Moonlight, otwierano: września 30, 2025, https://www.themoonlight.io/en/review/pats-process-level-adaptive-thinking-mode-switching
64. Self-adaptive reasoning for science - Microsoft Research, otwierano: września 30, 2025, https://www.microsoft.com/en-us/research/blog/self-adaptive-reasoning-for-science/
65. Teaching LLMs According to Their Aptitude: Adaptive ... - arXiv, otwierano: września 30, 2025, https://arxiv.org/abs/2502.12022
66. Language-Augmented Symbolic Planner for Open-World Task Planning - Robotics, otwierano: września 30, 2025, https://www.roboticsproceedings.org/rss20/p037.pdf
67. Planning in the Dark: LLM-Symbolic Planning Pipeline Without Experts, otwierano: września 30, 2025, https://ojs.aaai.org/index.php/AAAI/article/view/34855/37010

68. SayCanPay: Heuristic Planning with Large Language Models Using Learnable Domain Knowledge, otwierano: września 30, 2025, https://ojs.aaai.org/index.php/AAAI/article/view/29991/31739
69. Fast and Accurate Task Planning using Neuro-Symbolic Language Models and Multi-level Goal Decomposition - arXiv, otwierano: września 30, 2025, https://arxiv.org/html/2409.19250v1
70. Enhancing Symbolic Planning Capabilities in LLMs through Logical Chain-of-Thought Instruction Tuning - Pulkit Verma, otwierano: września 30, 2025, https://pulkitverma.net/assets/pdf/vlfms_lm4plan25/vlfms_lm4plan25.pdf
71. [2509.13351] Teaching LLMs to Plan: Logical Chain-of-Thought Instruction Tuning for Symbolic Planning - arXiv, otwierano: września 30, 2025, https://arxiv.org/abs/2509.13351
72. Symbolic Optimization with SMT Solvers - cs.wisc.edu, otwierano: września 30, 2025, https://pages.cs.wisc.edu/~aws/papers/popl14.pdf
73. SAT 2024 - The International Conferences on Theory and Applications of Satisfiability Testing (SAT), otwierano: września 30, 2025, https://satisfiability.org/SAT24/
74. Imandra Inc.: Home of Reasoning as a Service®, otwierano: września 30, 2025, https://www.imandra.ai/
75. What is Automated Reasoning? - Automated Reasoning Explained ..., otwierano: września 30, 2025, https://aws.amazon.com/what-is/automated-reasoning/
76. [2407.04973] LogicVista: Multimodal LLM Logical Reasoning Benchmark in Visual Contexts, otwierano: września 30, 2025, https://arxiv.org/abs/2407.04973
77. MMMU, otwierano: września 30, 2025, https://mmmu-benchmark.github.io/
78. Yijia-Xiao/LogicVista - GitHub, otwierano: września 30, 2025, https://github.com/Yijia-Xiao/LogicVista
79. LogicVista: Multimodal LLM Logical Reasoning Benchmark in Visual Contexts, otwierano: września 30, 2025, https://openreview.net/forum?id=6ozaf7VRIP
80. [Literature Review] LogicVista: Multimodal LLM Logical Reasoning Benchmark in Visual Contexts - Moonlight, otwierano: września 30, 2025, https://www.themoonlight.io/en/review/logicvista-multimodal-llm-logical-reasoning-benchmark-in-visual-contexts
81. LogicVista: Multimodal LLM Logical Reasoning Benchmark in Visual ..., otwierano: września 30, 2025, https://www.researchgate.net/publication/382080924_LogicVista_Multimodal_LLM_Logical_Reasoning_Benchmark_in_Visual_Contexts?_tp=eyJjb250ZXh0Ijp7InBhZ2UiOiJzY2llbnRpZmljQ29udHJpYnV0aW9ucyIsInByZXZpb3VzUGFnZSI6bnVsbCwic3ViUGFnZSI6bnVsbH19
82. ARC-AGI-3 - ARC Prize, otwierano: września 30, 2025, https://arcprize.org/arc-agi/3/
83. Discovering Top 3 Frontier LLMs Through Benchmarking — Arc AGI 3 | Towards AI, otwierano: września 30, 2025, https://towardsai.net/p/machine-learning/discovering-top-3-frontier-llms-through-benchmarking-arc-agi-3
84. arxiv.org, otwierano: września 30, 2025, https://arxiv.org/html/2501.10799v1

85. Learning to reason with LLMs | OpenAI, otwierano: września 30, 2025, https://openai.com/index/learning-to-reason-with-llms/
86. SWE Benchmark: LLM evaluation in Software Engineering Setting ..., otwierano: września 30, 2025, https://medium.com/@sulbha.jindal/swe-benchmark-llm-evaluation-in-software-engineering-setting-52f315b2de5a
87. Warp scores 71% on SWE-bench Verified, otwierano: września 30, 2025, https://www.warp.dev/blog/swe-bench-verified
88. LiveCodeBench Leaderboard | Kaggle, otwierano: września 30, 2025, https://www.kaggle.com/benchmarks/open-benchmarks/livecodebench
89. LiveCodeBench: Holistic and Contamination Free Evaluation of ..., otwierano: września 30, 2025, https://livecodebench.github.io/
90. AutoCodeBench: Large Language Models are Automatic Code Benchmark Generators, otwierano: września 30, 2025, https://arxiv.org/html/2508.09101v1
91. AutoCodeBench: Large Language Models are Automatic Code Benchmark Generators, otwierano: września 30, 2025, https://www.researchgate.net/publication/394458196_AutoCodeBench_Large_Language_Models_are_Automatic_Code_Benchmark_Generators
92. Paper page - AutoCodeBench: Large Language Models are ..., otwierano: września 30, 2025, https://huggingface.co/papers/2508.09101