

# Empirical-MCTS: Continuous Agent Evolution via Dual-Experience Monte Carlo Tree Search

Hao Lu<sup>1</sup> Haoyuan Huang<sup>1</sup> Yulin Zhou<sup>1</sup> Chen Li<sup>1</sup> Ningxin Zhu<sup>1</sup>

## Abstract

Inference-time scaling strategies, particularly Monte Carlo Tree Search (MCTS), have significantly enhanced the reasoning capabilities of Large Language Models (LLMs). However, current approaches remain predominantly stateless, discarding successful reasoning patterns after each problem instance and failing to mimic the empirical accumulation of wisdom characteristic of human problem-solving. To bridge this gap, we introduce Empirical-MCTS, a dual-loop framework that transforms stateless search into a continuous, non-parametric learning process. The framework unifies local exploration with global memory optimization through two novel mechanisms: Pairwise-Experience-Evolutionary Meta-Prompting (PE-EMP) and a Memory Optimization Agent. PE-EMP functions as a reflexive optimizer within the local search, utilizing pairwise feedback to dynamically synthesize adaptive criteria and evolve meta-prompts (system prompts) in real-time. Simultaneously, the Memory Optimization Agent manages a global repository as a dynamic policy prior, employing atomic operations to distill high-quality insights across problems. Extensive evaluations on complex reasoning benchmarks, including AIME25, ARC-AGI-2, and MathArena Apex, demonstrate that Empirical-MCTS significantly outperforms both stateless MCTS strategies and standalone experience-driven agents. These results underscore the critical necessity of coupling structured search with empirical accumulation for mastering complex, open-ended reasoning tasks.

## 1. Introduction

Large Language Models (LLMs) have demonstrated that increasing computational capability at inference time can significantly improve reasoning performance (OpenAI et al., 2024; 2025b). Techniques such as "Best-of-N" sampling (Brown et al., 2024; Li et al., 2022; Lightman et al., 2023) and Monte Carlo Tree Search (MCTS) (Zhou et al., 2024; Feng et al., 2024; Xu, 2023; Zhang et al., 2024a,b; Inoue et al., 2025; Lu et al., 2025; Xie et al., 2024) allow models to explore multiple reasoning paths and self-correct, effectively scaling performance without the need for expensive parameter updates. However, a critical limitation of current inference-time scaling methods is that they are stateless. Whether using standard MCTS or adaptive branching strategies (Inoue et al., 2025), the agent treats each new problem as an isolated event. Once a search process concludes, the successful strategies or valid reasoning patterns discovered are discarded. This contrasts with human problem-solving, which is empirical: experts solve problems by combining long-term experience (accumulated domain knowledge) with short-term experience (immediate feedback and context from the current problem).

Existing methods attempting to address this lack of memory often fall short in integration. Systems like FLEX (Cai et al., 2025b) maintain an experience library but treat retrieval and reasoning as separate steps, preventing the model from evolving its search strategy dynamically. Other approaches, such as Training-Free GRPO (Cai et al., 2025a), use historical data to adjust the generation probability but lack the structured exploration provided by tree search, limiting their effectiveness on complex, multi-step logic tasks.

To address these limitations, we propose Empirical-MCTS, a framework that integrates continuous experience accumulation with structured tree search. Our approach unifies two distinct types of experience via a dual-loop evolutionary mechanism:

**Short-term Experience (PE-EMP):** Within the local search process, we introduce Pairwise-Experience-Evolutionary Meta-Prompting (PE-EMP). Instead of using static prompts for node expansion, PE-EMP functions as a reflexive optimizer. It analyzes pairwise response differences to syn-

<sup>1</sup>JianChengXingYun Technology Co., Ltd., Shenzhen, China.  
Correspondence to: Hao Lu <luhao@jianchengxingyun.com>, Chen Li <lichen@jianchengxingyun.com>.

thesize adaptive criteria and dynamically evolves the meta-prompt (system prompt). This ensures that immediate feedback is not merely used for selection, but actively refines the generation policy for subsequent steps (Liu et al., 2025).

**Long-term Experience (Memory Optimization):** To sustain learning across problems, we implement a Memory Optimization Agent. Rather than a static retrieval database, our repository is treated as a dynamic policy prior. We employ an optimizer-based update mechanism—utilizing atomic operations such as add, modify, and merge—to continuously distill high-quality insights from the search process into a global knowledge base (Cai et al., 2025a).

Furthermore, to ensure rigorous evaluation of the evolved prompts, we integrate a hybrid preference model adapted from recent pairwise optimization strategies (Zhang et al., 2024b). By mapping the explicit “self-principled” scores generated by PE-EMP into a globally consistent ranking via Enhanced Borda Count, we address the non-transitivity inherent in LLM preferences. This integration allows Empirical-MCTS to accurately distinguish and prioritize peak performance trajectories, ensuring that the qualitative insights gained from memory optimization are effectively translated into quantitative search guidance.

We evaluate Empirical-MCTS on diverse benchmarks, including AIME25 (Zhang & Math-AI, 2025), ARC-AGI-2 (Chollet et al., 2026), and MathArena Apex (Balunović et al., 2025), using DeepSeek-V3.1-Terminus (DeepSeek-AI, 2024), gpt-oss-120b (OpenAI et al., 2025a) and Gemini 3 Pro (DeepMind, 2025b). Empirical-MCTS achieved better results than previous approaches, such as repeated sampling, LLaMA-Berry, FLEX and Training-free GRPO. Our key contributions are:

- We introduce Empirical-MCTS, a novel paradigm that bridges the gap between structured search and continuous learning. By unifying local exploration with global memory optimization, we transform MCTS from a stateless inference technique into a non-parametric online learning agent capable of accumulating “wisdom” across disparate problem instances without weight updates.
- We propose Pairwise-Experience-Evolutionary Meta-Prompting (PE-EMP). Moving beyond static expansion, PE-EMP acts as a reflexive optimizer that synthesizes “self-principles” from pairwise feedback. This mechanism dynamically evolves the meta-prompt (system prompt) in real-time, allowing the agent to actively refine its reasoning policy and navigate complex logic spaces with increasing precision during the search.
- We show that Empirical-MCTS outperforms standard MCTS and previous experience-based models.

It achieves state-of-the-art results on high-difficulty benchmarks like AIME25, ARC-AGI-2, and MathArena Apex. Our results prove that “remembering” past reasoning patterns is the key to solving complex, multi-step logic problems more efficiently than traditional search methods.

## 2. Related Work

### 2.1. Inference-Time Scaling and Tree Search

Inference-time scaling focuses on improving model outputs by allocating more compute during generation. Early methods, such as Self-Consistency (Wang et al., 2023), generate multiple independent answers and select the most frequent one. To handle more complex tasks, tree-search algorithms like Tree of Thoughts (ToT) (Yao et al., 2023a) and MCTS applied to LLMs (Zhou et al., 2024; Feng et al., 2024; Xu, 2023; Zhang et al., 2024a;b; Inoue et al., 2025; Lu et al., 2025; Xie et al., 2024) were developed to structure the reasoning process into steps. Recent work has focused on optimizing the search structure. AB-MCTS (Inoue et al., 2025) introduces adaptive branching and LLaMA-Berry (Zhang et al., 2024b) integrates MCTS with a pairwise reward model to improve selection accuracy. However, despite these structural improvements, these methods remain stateless across problems. They do not retain the successful reasoning patterns discovered during the search. Empirical-MCTS builds upon the robust search backbone of LLaMA-Berry but introduces a stateful paradigm, persisting empirical experience to evolve the agent’s policy continuously.

### 2.2. Experience-Driven Agents

Several studies aim to enable agents to learn from interaction without parameter updates. FLEX (Cai et al., 2025b) proposes a “Forward Learning” paradigm that stores successful trajectories in a library for future retrieval. However, FLEX largely relies on static retrieval mechanisms and does not deeply integrate this experience into the step-by-step decision-making of a search tree. Training-Free GRPO (Cai et al., 2025a) takes a different approach by acting as an “inference-time optimizer,” using the LLM to analyze group results and update the context effectively. While this avoids gradient updates, it operates primarily as a policy adjustment tool rather than a reasoning search engine. Empirical-MCTS bridges these approaches by using MCTS to generate high-quality reasoning data and a global repository to store and refine this data, ensuring the agent becomes more efficient over time.

### 2.3. Meta-Reasoning and Refinement

Iterative refinement techniques, such as Self-Refine (Madaan et al., 2023) and Reflexion (Shinn et al., 2023), allow models to critique their own outputs. However, standard self-refinement can sometimes lead to the model discarding correct intermediate steps or drifting away from the original problem constraints. MCTSR-Zero (Lu et al., 2025) suggests using "Meta-Prompts"—high-level instructions generated by the model—to guide subsequent generation. We formalize this as our "Short-term Experience" module. In our MCTS expansion step, we use explicit meta-prompts containing critiques and suggestions to generate child nodes. This ensures that the reasoning process remains focused and builds constructively on previous steps.

## 3. Methodology: The Empirical-MCTS Framework

### 3.1. Pairwise-Experience-Evolutionary Meta-Prompting (PE-EMP)

In Empirical-MCTS, the evolution of the meta-prompt (system prompt) is not an isolated heuristic but a derivative of the reward modeling process. Drawing inspiration from the Self-Principled Critique Tuning (SPCT) framework proposed for inference-time scaling of generalist reward models (Liu et al., 2025), we introduce a unified mechanism termed Pairwise-Experience-Evolutionary Meta-Prompting (PE-EMP).

SPCT demonstrates that generating specific "self-principles" prior to critiquing significantly enhances the scalability and accuracy of reward models. We extend this paradigm by transforming the Judge model from a mere discriminator into a reflexive optimizer. While SPCT utilizes principles to scale inference accuracy via sampling, PE-EMP leverages these generated principles (Adaptive Criteria) to drive the evolution of the generation policy itself.

Formally, let a node expansion result in a pairwise comparison between a candidate state  $S_c = (\mathcal{P}_c, r_c)$  and a baseline (parent) state  $S_p = (\mathcal{P}_p, r_p)$ , where  $\mathcal{P}$  denotes the meta-prompt and  $r$  the model response. The PE-EMP module  $\mathcal{J}$  processes the query  $q$ , prior accumulated experiences  $\mathcal{E}_{prior}$ , and the two states to yield a triadic output:

$$\mathcal{J}(q, \mathcal{E}_{prior}, S_c, S_p) \rightarrow (\mathbf{s}, \mathcal{E}_{new}, \mathcal{P}_{evolved}) \quad (1)$$

where  $\mathbf{s}$  is the raw score vector,  $\mathcal{E}_{new}$  denotes the new empirical insights derived from practice, and  $\mathcal{P}_{evolved}$  is the evolved meta-prompt. The internal logic of the PE-EMP executes a strict seven-stage cognitive process that mirrors and extends the SPCT pipeline:

---

### Algorithm 1 Empirical-MCTS: Continuous Agent Evolution

---

```

1: Input: Query  $q$ , Initial Prompt  $\mathcal{P}_0$ , Global Experience Library  $\mathcal{D}$ , Iterations  $T$ 
2: Initialize: Search tree  $\mathcal{T}$  with root  $v_0(q, \mathcal{P}_0)$ 
3:  $\mathcal{P}_{evolved} \leftarrow \mathcal{P}_0$  {Initialize evolved prompt with base prompt}
4:  $\mathcal{E}_{prior} \leftarrow \text{Retrieve}(\mathcal{D}, q)$  {Retrieve long-term experience}
5: for  $t = 1$  to  $T$  do
6:    $S_p \leftarrow \text{Selection}(\mathcal{T}, \text{UCB})$  {Navigate tree nodes}
7:   {— Local Loop: PE-EMP Expansion —}
8:    $S_c \leftarrow \text{Expand}(S_p, \mathcal{P}_{evolved}, \mathcal{E}_{prior})$  {Generate child state}
9:    $(\mathbf{s}, \mathcal{E}_{new}, \mathcal{P}_{evolved}) \leftarrow \mathcal{J}(q, \mathcal{E}_{prior}, S_c, S_p)$  {PE-EMP Reflexive Optimizer}
10:  // PE-EMP internally: (1) Generate Criteria → (2) Critique → (3) Scoring → (4) Prompt Evolution
11:  {— Value Backpropagation —}
12:   $R \leftarrow \text{Hybrid-Reward}(\mathbf{s}, \text{Borda-Count}(\mathcal{T}))$  {Section 3.3}
13:  Update-Q-Values( $\mathcal{T}, S_p, R$ ) {Decay-based back-prop}
14:  {— Global Loop: Memory Update —}
15:  if  $\mathcal{E}_{new}$  contains high-value insights then
16:     $\pi_{mem} \leftarrow \text{Optimizer}(\mathcal{D}, \mathcal{E}_{new})$  {Determine Add/Modify/Merge/Delete}
17:     $\mathcal{D} \leftarrow \text{Update-Library}(\mathcal{D}, \pi_{mem})$  {Non-parametric policy update}
18:  end if
19: end for
20: Output: Best response  $r^*$  from  $\mathcal{T}$ 

```

---

1. **Adaptive Criteria Generation (Self-Principled Phase):** Aligning with SPCT's methodology, the model first acts as a principle generator. It dynamically formulates specific evaluation dimensions tailored to the user's query, ensuring that the subsequent critique is grounded in context-aware standards rather than generic rules.
2. **Comparative Chain-of-Thought (Critique Phase):** A step-by-step critical analysis is performed, contrasting the candidate and baseline responses against the generated criteria.
3. **Dynamic Weight Allocation:** Weights are assigned to each criterion based on their relevance to the specific problem context, ensuring  $\sum w_i = 100\%$ .
4. **Weighted Scoring:** A quantitative calculation determines the performance scores for both responses based on the weighted criteria.

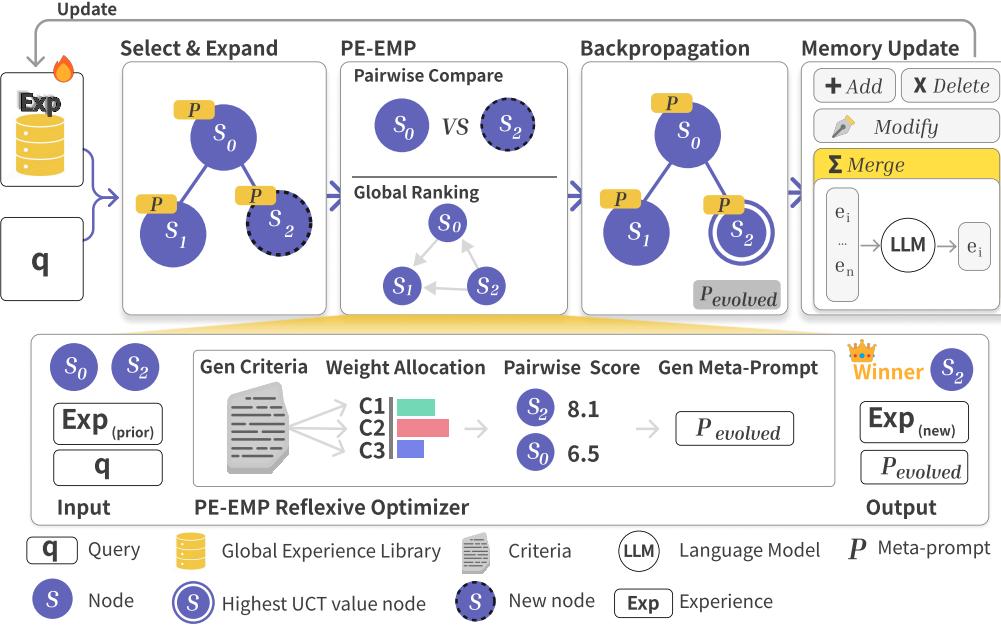


Figure 1. Concrete Instantiation of Empirical-MCTS Framework.

5. **Raw Score Generation:** The model performs a step-by-step arithmetic aggregation, calculating the weighted sum of dimension-specific scores derived in the previous step. This yields a precise scalar value within the range  $[0, 10]$  for each response, serving as the high-fidelity signal for probabilistic preference modeling.
6. **Context-Aware Insight Synthesis:** Building upon the *prior experiences* retrieved from the RAG repository, the model synthesizes the immediate feedback with historical wisdom. It distills the current interaction into new, high-order empirical insights ( $\mathcal{E}_{new}$ ), explicitly identifying how the current case refines, expands, or corrects the existing knowledge boundary provided by the retrieval system.
7. **Strategic Prompt Evolution:** Finally, the model analyzes the fundamental execution differences between  $\mathcal{P}_c$  and  $\mathcal{P}_p$ . By internalizing the *Critical Success Factors* of the winner and the *Failure Lessons* of the loser, it synthesizes a more advanced meta-prompt  $\mathcal{P}_{evolved}$ .

This architecture ensures that  $\mathcal{P}_{evolved}$  is not a random mutation but a conscious adaptation. By incorporating the *self-principled* logic from SPCT, the meta-prompt explicitly encodes the high-quality evaluation criteria and lessons learned from the immediate pairwise battle, creating a directed evolutionary pressure toward prompts that produce higher-reward responses.

To prevent semantic drift in abstract reasoning tasks,

we further filter retrieved experiences by task taxonomy alignment. Each experience in  $\mathcal{D}$  is tagged with a coarse-grained task type (e.g., *geometry-proof*, *combinatorial-optimization*) derived from the original problem source. Only experiences sharing the same task type as  $q$  (determined via zero-shot classification with the base LLM) are included in  $\mathcal{E}_{prior}$ . We set  $k = 5$  as the default retrieval size, balancing context richness against prompt length constraints.

### 3.2. Experience Memory Optimization

A critical contribution of Empirical-MCTS is its active management of the experience library, a methodology directly inspired by the Training-Free GRPO (Cai et al., 2025a). Training-Free GRPO posits that LLM agents can achieve alignment effects analogous to parametric reinforcement learning (e.g., PPO (Schulman et al., 2017)) by iteratively refining experiential knowledge as a "token prior" rather than updating model weights.

Adopting this paradigm, we implement a *Memory Optimization Agent* that functions as the optimizer in the context space. Unlike static RAG systems that simply append inputs, this agent treats the distilled insights  $\mathcal{E}_{new}$  as gradients for updating the global knowledge state.

Let  $\mathcal{D}_t$  be the current experience library. Upon receiving the batch of new insights from the PE-EMP phase, the system invokes the optimization routine. Following the optimization protocol defined in Training-Free GRPO, the agent analyzes the new insights against the existing library  $\mathcal{E}_{exist}$

and generates a structured update plan  $\pi_{mem}$  consisting of four atomic operations:

- **Add:** Append distinct, high-quality experiences described in the new insights to the library  $\mathcal{D}_t$ , effectively expanding the policy’s action space.
- **Modify:** Refine or correct an existing experience  $e_i \in \mathcal{D}_t$  based on the fresh “semantic advantage” derived from the current iteration, sharpening the policy’s precision.
- **Merge:** Synthesize multiple fragmented experiences  $\{e_i, e_j, \dots\}$  into a single, cohesive principle to reduce context redundancy and improve retrieval efficiency.
- **Delete:** Prune obsolete or low-quality experiences that no longer align with high-reward outcomes, functioning as a “forgetting” mechanism for bad policies.

Formally, this step represents a non-parametric policy update:

$$\mathcal{D}_{t+1} \leftarrow \text{Optimizer}(\mathcal{D}_t, \pi_{mem}(\mathcal{E}_{new}, \mathcal{E}_{exist})) \quad (2)$$

By continuously optimizing  $\mathcal{D}$ , Empirical-MCTS shifts the output distribution  $\pi(y|q, \mathcal{D})$  towards higher rewards over time, achieving continuous self-improvement without the computational cost of gradient-based training.

### 3.3. Value Estimation via Hybrid Preference Integration

To transform the discrete insights from PE-EMP into a continuous reward signal, we adapt the robust Pairwise Preference Reward Model (PPRM) architecture from LLaMA-Berry (Zhang et al., 2024b). Unlike previous approaches that rely on opaque logits from a separate reward model, our framework uniquely leverages the explicit “Self-Principled Scores” generated by PE-EMP (Section 3.1) as the high-fidelity input source. This integrates the fine-grained, instruction-following assessment of our reflexive optimizer with the global consistency guarantees of graph-theoretic ranking.

**Local and Global Valuation.** We map the raw score vector  $s = (S_c, S_p)$  from PE-EMP to a transition probability via the Bradley-Terry model:  $Q_{local}(s_c) = \frac{\exp(S_c)}{\exp(S_c) + \exp(S_p)}$ . To address the non-transitivity inherent in pairwise LLM preferences, we employ the Enhanced Borda Count (EBC) method (Zhang et al., 2024b) to construct a global ranking from the pairwise transition graph. The final reward  $R(s)$  is a weighted fusion of the local probability and the global Borda rank, ensuring that the evolved prompts are evaluated within the context of the entire search history.

### 3.4. Search Strategy

We employ a standard Upper Confidence Bound (UCB) applied to Trees for node selection. For value backpropagation, we adopt the decay-based update rule from Self-Refine MCTS (Zhang et al., 2024b):  $Q(s_p) \leftarrow (1 - \gamma)Q(s_p) + \gamma Q(s_c)$ . This “soft-max” mechanism is particularly essential for our framework, as it allows the sharp, high-reward insights discovered by the Memory Optimization Agent to propagate efficiently up the tree without being diluted by earlier, less experienced exploration attempts.

## 4. Experiments

### 4.1. Experimental Setup

**Benchmarks.** We evaluated Empirical-MCTS on three high-difficulty benchmarks designed to test frontier reasoning and generalization capabilities: AIME25 (Zhang & Math-AI, 2025), MathArena Apex (Balunović et al., 2025), and ARC-AGI-2 (Chollet et al., 2026). AIME25 represents a standard high-school competition mathematics benchmark. MathArena Apex is a dynamic evaluation framework specifically curated to mitigate data contamination—a pervasive issue in current LLM evaluation—by utilizing problems from recent competitions (e.g., CMIMC 2025, IMO 2025). It is currently the only benchmark evaluating proof-writing capabilities. ARC-AGI (Abstraction and Reasoning Corpus) measures general fluid intelligence, requiring agents to synthesize transformation programs from minimal examples, a task where traditional LLMs historically struggle.

**Models.** Experiments were conducted using a diverse set of frontier models to demonstrate model-agnostic applicability: DeepSeek-V3.1-Terminus (DeepSeek-AI, 2024), gpt-oss-120b (OpenAI et al., 2025a), and the Gemini 3 Flash (DeepMind, 2025a).

**Baselines.** We benchmark Empirical-MCTS against a spectrum of inference-time strategies: (1) LLM with In-Context Learning (ICL) (Dong et al., 2024), (2) LLM agent with reasoning-and-acting workflow (ReAct) (Yao et al., 2023b), (3) Repeated Sampling (Best-of- $N$ ) (Brown et al., 2024; Li et al., 2022; Lightman et al., 2023), (4) FLEX (Cai et al., 2025b), (5) LLaMA-Berry (Zhang et al., 2024b), and (6) Training-Free GRPO (Cai et al., 2025a).

### 4.2. Results on Mathematical Reasoning

We first assess the impact of adding empirical memory to structured search on mathematical reasoning tasks. Table 1 summarizes the results on AIME25 and MathArena Apex.

**Surpassing Stateless Search.** On AIME25, Empirical-MCTS achieves 73.3%, outperforming both the stateless LLaMA-Berry framework (63.3%) and the memory-augmented FLEX agent (66.6%). This result validates

**Table 1.** Performance comparison on AIME25 and MathArena-Apex. **Baseline** denotes DeepSeek-V3.1-Terminus. All evaluated frameworks utilize this model as the underlying backbone.

METHOD / FRAMEWORK	AIME25	MATHARENA-APEX
BASELINE	56.7	0.00 (16 RUNS)
ICL	53.3	-
REACT	60.0	-
FLEX	66.6	-
REPEATED SAMPLING	70.0	0.00 (4 RUNS)
LLAMA-BERRY	63.3	2.08 (4 RUNS)
<b>Ours</b>	<b>73.3</b>	<b>4.17 (4 RUNS)</b>

our core hypothesis: By integrating memory optimization directly into the node expansion via PE-EMP, Empirical-MCTS allows the agent to navigate the solution space more efficiently.

**Solving the Unsolvable.** The results on MathArena Apex are noteworthy. The base model, DeepSeek-V3.1-Terminus, scored 0.00% across 16 runs, indicating that the model lacks the requisite reasoning priors for these problems. While Repeated Sampling also failed (0.00%) and LLaMA-Berry achieved only marginal success (2.08%), Empirical-MCTS reached 4.17%. While the absolute score is low due to the extreme difficulty of the benchmark, the relative improvement is substantial. It demonstrates that our framework does not merely extract existing knowledge but synthesizes new solution paths through the accumulation of empirical wisdom during the search.

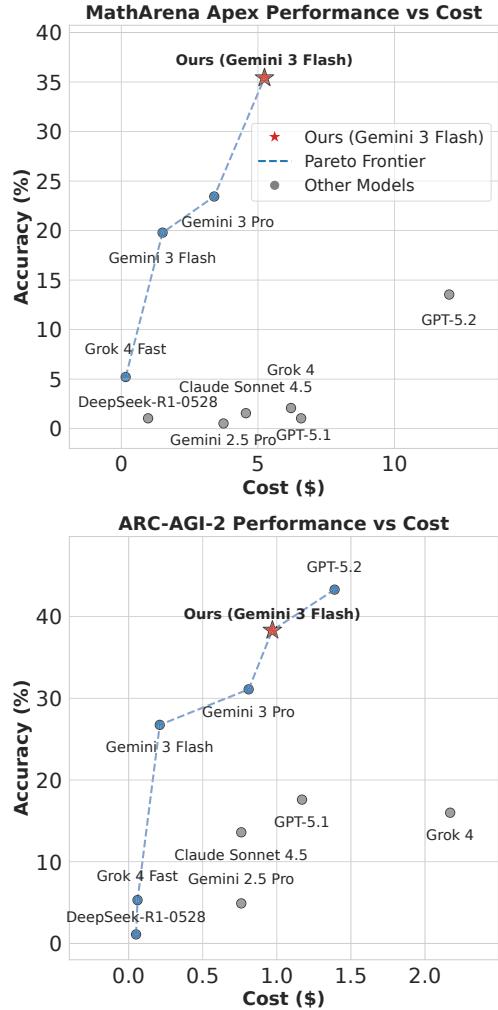
#### 4.3. Efficiency and Cost Analysis

A critical barrier to inference-time scaling is cost. In Table 2, we compare Empirical-MCTS using the cost-efficient Gemini 3 Flash against state-of-the-art frontier models on MathArena Apex and ARC-AGI-2.

**High Performance at Low Cost.** Empirical-MCTS enables smaller, faster models to punch above their weight class. On MathArena Apex, Ours (Gemini 3 Flash) achieves an accuracy of 35.42% with a total cost of only \$5.24. This outperforms Gemini 3 Pro, which scores 23.44% at a cost of \$3.40, and drastically outperforms GPT-5.2 (High), which scores 13.54% at a cost of \$12.00. Similarly, on ARC-AGI-2, our method achieves 38.33%, surpassing the much more expensive Gemini 3 Pro (31.1%) and Grok 4 (16.0%).

Figure 2 visualizes the Pareto frontier of reasoning efficiency. As shown in the upper panel (MathArena Apex), our approach strictly dominates all baselines, achieving the highest accuracy while maintaining a cost significantly lower than the heavy-compute models (e.g., GPT-5.2 at \$12.00). In the ARC-AGI-2 landscape, although GPT-5.2 holds a marginal accuracy lead, Empirical-MCTS resides on the “knee” of the curve—delivering comparable frontier-level

reasoning at a reduced price point.



**Figure 2. Cost-Performance Pareto Frontier Analysis.** We plot accuracy against inference cost for various models on MathArena Apex and ARC-AGI-2. The dashed blue line indicates the Pareto frontier, representing the optimal trade-off between cost and performance.

#### 4.4. Ablation Studies and Scalability Analysis

To disentangle the contributions of PE-EMP, Memory Optimization, and Meta-Prompting, we conducted extensive ablation studies using gpt-oss-120b on the AIME25 benchmark. We also analyzed the scalability of our method with respect to the number of search rollouts.

**Component Contribution and Scaling Behavior.** Figure 3 presents the performance trajectory across different rollout configurations. Key observations include:

- **Full Framework:** Achieves a peak of **76.7%** at 8 rollouts, demonstrating continuous improvement (63.3% → 76.7%) that suggests further gains with increased

Table 2. Performance and cost analysis on MathArena Apex and ARC-AGI-2. Empirical-MCTS with Gemini 3 Flash outperforms more expensive models (e.g., GPT-5.2, Grok 4), establishing a new Pareto frontier for cost-effective reasoning.

MODEL	MATHARENA APEX		ARC-AGI-2	
	ACC	COST (\$)	ACC	COST (\$)
<b>OURS (GEMINI 3 FLASH)</b>	<b>35.42 %</b>	5.24	38.33 %	0.97
GPT-5.2 (HIGH) (OPENAI, 2025B)	13.54 %	12.00	<b>43.3 %</b>	1.39
GEMINI 3 PRO (DEEPMIND, 2025B)	23.44 %	3.40	31.1 %	0.81
GEMINI 3 FLASH (DEEPMIND, 2025A)	19.79 %	1.51	26.75 %	0.21
GROK 4 (X.AI, 2025A)	2.08 %	6.21	16.0 %	2.17
GPT-5.1 (HIGH) (OPENAI, 2025A)	1.04 %	6.58	17.6 %	1.17
CLAUDE SONNET 4.5 (ANTHROPIC, 2025)	1.56 %	4.56	13.6 %	0.76
GROK 4 FAST (REASONING) (X.AI, 2025B)	5.21 %	0.16	5.3 %	0.06
GEMINI 2.5 PRO (GOOGLE, 2025)	0.52 %	3.74	4.9 %	0.76
DEEPSPEECH-R1-0528 (GUO ET AL., 2025)	1.04 %	0.98	1.1 %	0.05

compute.

- **w/o Meta-Prompt:** Removing explicit meta-prompts (with static prompts) in PE-EMP but retaining memory drops performance to 66.7% at 8 rollouts. This indicates that while memory provides the *content*, meta-prompts are essential for *instructing* the model on how to apply that content effectively.
- **w/o Memory & Meta-Prompt:** This configuration, which excludes the experience library and utilizes static prompts, peaks at 66.7% (rollout 3) but then regresses to 63.3% at rollout 8, failing to accumulate "wisdom" from previous rollouts.
- **w/o PE-EMP & Memory:** To rigorously isolate the efficacy of the self-principled and pairwise comparison mechanisms, this variant removes the PE-EMP module, effectively degenerating into a standard MCTS framework. In this setting, the reward model utilizes static pre-defined criteria and assesses node quality pointwise scoring of individual responses, rather than through relative pairwise comparison. The resulting performance drop (to 56.7% at rollout 8) mirrors that of naive repeated sampling, underscoring that the core engine of our improvement is the *evolution* of the prompt and experience driven by pairwise signals, not merely the tree structure.
- **Repeated Sampling:** Plateaus at 56.7% after 2 rollouts, confirming the limitations of stateless approaches.

**Empirical Memory Growth Dynamics.** Figure 4 quantifies how the global experience repository expands during search. Starting from zero experiences at initialization, the repository grows monotonically with each rollout (110 → 311 experiences after 8 rollouts), demonstrating the framework's capacity for continuous knowledge accumulation without parameter updates.

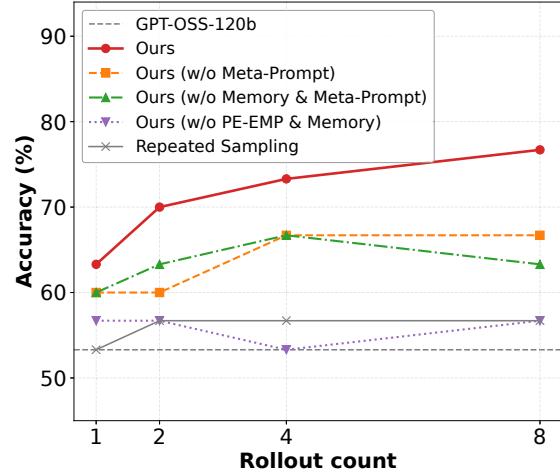
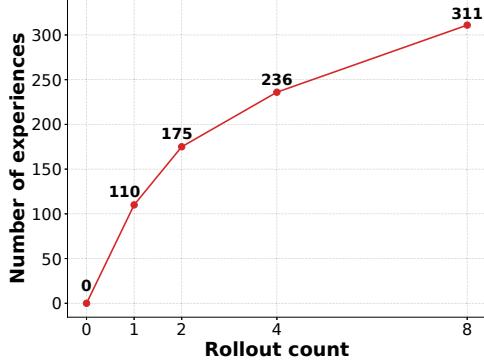


Figure 3. Ablation study on AIME25 using gpt-oss-120b.

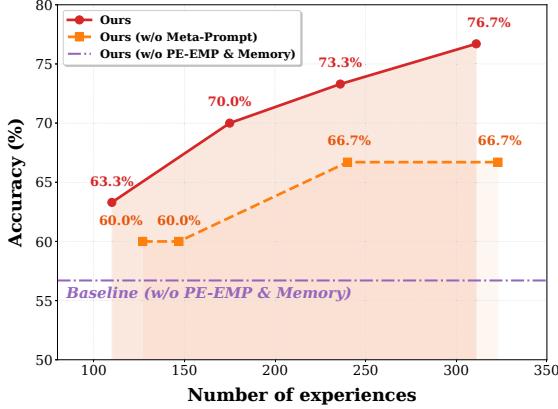
**Performance-Experience Correlation.** Figure 5 reveals a strong positive correlation between accumulated experiences and reasoning performance. The full framework (red) shows a clear upward trajectory (63.3% → 76.7%) as experiences grow from 110 to 311. In contrast, the variant without meta-prompts (yellow) exhibits diminishing returns after 240 experiences, while the baseline (w/o PE-EMP & Memory at rollout 8) ultimately reached 56.7%. This confirms that both memory accumulation and meta-prompts are essential for translating experiences into performance gains.

#### 4.5. Qualitative Analysis of Policy Evolution

To understand *how* Empirical-MCTS improves, we analyzed the evolution of meta-prompts in the MathArena Apex tasks. In initial rollouts, the prompt  $\mathcal{P}_{\text{evolved}}$  focused on generic instructions (e.g., "Check your calculations"). By Rollout 4, after the Memory Optimization Agent had distilled fail-



**Figure 4. Growth of empirical memory repository during search.** The number of distilled experiences increases monotonically with rollout count ( $0 \rightarrow 8$ ), enabling progressive policy refinement across problem instances.



**Figure 5. Performance vs. accumulated experiences on AIME25.** The full framework (red) demonstrates a strong positive correlation between experience count and accuracy, while ablated variants show limited improvement (yellow). Baseline performance shown as horizontal reference line (56.7%).

ures from invalid proofs, the meta-prompt had evolved to include highly specific constraints: *“For geometry proofs involving cyclic quadrilaterals, explicitly verify Ptolemy’s inequality before assuming existence.”* This transition from generic to domain-specific guidance, driven by the feedback loop between PE-EMP and the Memory Agent, explains the model’s ability to solve problems that were previously intractable for the base model.

## 5. Conclusion

We presented Empirical-MCTS, a framework that unifies structured search with non-parametric online learning. By coupling Pairwise-Experience-Evolutionary Meta-Prompting (PE-EMP) with a dynamic global memory optimizer, our approach enables agents to accumulate empirical wisdom and refine their reasoning policies in real-time without gradient updates.

Empirical evaluations across AIME25, MathArena Apex, and ARC-AGI-2 confirm that this dual-loop mechanism establishes a new Pareto frontier for reasoning efficiency. Our results demonstrate that converting search history into actionable policy priors allows significantly smaller models to outperform larger, stateless baselines, validating that “remembering” reasoning patterns is as critical as the search capability itself.

**Limitations.** A primary limitation of our framework is the assumption that the accumulation of experience and the evolution of meta-prompts are inherently positive. In practice, this relies on the base model’s capability to accurately verify reasoning steps. For extremely complex tasks that exceed the model’s verification boundary, there is a risk of integrating low-quality insights or “hallucinated wisdom.” This can lead to degenerative evolution, where the meta-prompt acts on false premises, steering the search towards suboptimal regions. Future research will focus on mitigating this risk by introducing robust consistency checks and uncertainty quantification into the memory optimization loop to filter toxic experiences. Additionally, we aim to extend Empirical-MCTS to multi-turn interactions and longer-horizon planning tasks, investigating how empirical memory can persist and adapt over extended operational contexts.

## References

- Anthropic. Claude sonnet 4.5. Technical report, Anthropic, September 2025.
- Balunović, M., Dekoninck, J., and Ivo Petrov, e. a. Matharena: Evaluating llms on uncontaminated math competitions. *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmark*, 2025.
- Brown, B., Juravsky, J., and Ryan Ehrlich, e. a. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.
- Cai, Y., Cai, S., and Yuchen Shi, e. a. Training-free group relative policy optimization. *arXiv preprint arXiv:2510.08191*, 2025a.
- Cai, Z., Guo, X., and Yu Pei, e. a. Flex: Continuous agent evolution via forward learning from experience. *arXiv preprint arXiv:2511.06449*, 2025b.
- Chollet, F., Knoop, M., and Gregory Kamradt, e. a. Arcagi-2: A new challenge for frontier ai reasoning systems. *arXiv preprint arXiv:2505.11831*, 2026.
- DeepMind, G. Gemini 3 flash model card. Technical report, Google DeepMind, December 2025a.
- DeepMind, G. Gemini 3 pro model card. Technical report, Google DeepMind, November 2025b.

- DeepSeek-AI. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.
- Dong, Q., Li, L., and Damai Dai, e. a. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2024.
- Feng, X., Wan, Z., and Muning Wen, e. a. Alphazero-like tree-search can guide large language model decoding and training. *arXiv preprint arXiv:2309.17179*, 2024.
- Google. Gemini 2.5 pro preview model card. Technical report, Google, May 2025.
- Guo, D., Yang, D., and Zhang, e. a. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 1476-4687.
- Inoue, Y., Misaki, K., Imajuku, Y., and So Kuroki, e. a. Wider or deeper? scaling LLM inference-time compute with adaptive branching tree search. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.
- Li, Y., Choi, D., Chung, J., and Kushman, e. a. Competition-level code generation with alphacode. *Science*, 378(6624): 1092–1097, December 2022. ISSN 1095-9203. doi: 10.1126/science.abq1158.
- Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., Leike, J., Schulman, J., Sutskever, I., and Cobbe, K. Let’s verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.
- Liu, Z., Wang, P., and Runxin Xu, e. a. Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*, 2025.
- Lu, H., Gu, Y., and Haoyuan Huang, e. a. Mcts-zero: Self-reflective psychological counseling dialogues generation via principles and adaptive exploration. *arXiv preprint arXiv:2505.23229*, 2025.
- Madaan, A., Tandon, N., and Gupta, e. a. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46534–46594. Curran Associates, Inc., 2023.
- OpenAI. Introducing gpt-5.1. Technical report, OpenAI, November 2025a.
- OpenAI. Introducing gpt-5.2. Technical report, OpenAI, December 2025b.
- OpenAI, ;, Jaech, A., and Adam Kalai, e. a. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.
- OpenAI, ;, Agarwal, S., and Lama Ahmad, e. a. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025a.
- OpenAI, ;, El-Kishky, A., Wei, A., and Andre Saraiva, e. a. Competitive programming with large reasoning models. *arXiv preprint arXiv:2502.06807*, 2025b.
- Schulman, J., Wolski, F., and Prafulla Dhariwal, e. a. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- Shinn, N., Cassano, F., and Gopinath, e. a. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652. Curran Associates, Inc., 2023.
- Wang, X., Wei, J., and Dale Schuurmans, e. a. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2023.
- x.ai. Grok 4. Technical report, x.ai, July 2025a.
- x.ai. Grok 4 fast. Technical report, x.ai, September 2025b.
- Xie, Y., Goyal, A., and Wenyue Zheng, e. a. Monte carlo tree search boosts reasoning via iterative preference learning. *arXiv preprint arXiv:2405.00451*, 2024.
- Xu, H. No train still gain. unleash mathematical reasoning of large language models with monte carlo tree search guided by energy function. *arXiv preprint arXiv:2309.03224*, 2023.
- Yao, S., Yu, D., and Zhao, Jeffrey, e. a. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 11809–11822. Curran Associates, Inc., 2023a.
- Yao, S., Zhao, J., and Dian Yu, e. a. React: Synergizing reasoning and acting in language models, 2023b.
- Zhang, D., Huang, X., and Dongzhan Zhou, e. a. Accessing gpt-4 level mathematical olympiad solutions via monte carlo tree self-refine with llama-3 8b. *arXiv preprint arXiv:2406.07394*, 2024a.
- Zhang, D., Wu, J., and Jingdi Lei, e. a. Llama-berry: Pairwise optimization for o1-like olympiad-level mathematical reasoning. *arXiv preprint arXiv:2410.02884*, 2024b.
- Zhang, Y. and Math-AI, T. American invitational mathematics examination (aime) 2025, 2025.
- Zhou, A., Yan, K., Shlapentokh-Rothman, M., and Haohan Wang, e. a. Language agent tree search unifies reasoning acting and planning in language models. *arXiv preprint arXiv:2310.04406*, 2024.