Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

MetaCoT: Reasoning-Informed Metaheuristic Search via Large Language Model

Kassem Danach

kassem.danach@mu.edu.lb

Al Maaref University College

Wael Hosny Fouad Aly

American University of the Middle East

Chadi Fouad Riman

American University of the Middle East

Research Article

Keywords: MetaCoT, Chain-of-Thought Reasoning, Large Language Models, Metaheuristics, Adaptive Optimization, Hybrid Algorithms, Combinatorial Optimization, Parameter Control, Operator Adaptation, NP-hard Problems

Posted Date: June 23rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9807056/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

MetaCoT: Reasoning-Informed Metaheuristic Search via Large Language Model

Kassem Danach[1*†], Wael Hosny Fouad Aly[2†] and Chadi Fouad Riman[2†]

[1*]Basic and Applied Sciences Research Center, Al Maaref University, Beirut, 1001, Lebanon.
[2]College of Engineering and Technology, American University of the Middle East, Egaila, 54200, Kuwait.

*Corresponding author(s). E-mail(s): kassem.danach@mu.edu.lb;
Contributing authors: wael.aly@aum.edu.kw;
chadi.riman@aum.edu.kw;
†These authors contributed equally to this work.

Abstract

Metaheuristic optimization methods are widely used for solving large-scale combinatorial problems, but their performance is often limited by fixed operator behaviors and low adaptivity. Recent advances in large generative models provide new opportunities for reasoning-driven algorithmic guidance, yet current integrations are largely superficial, typically using such models as direct solvers rather than as mechanisms for shaping metaheuristic dynamics. This paper introduces MetaCoT, a hybrid optimization framework that leverages chain-of-thought reasoning from large language models to guide and modulate metaheuristic search. CoT-derived insights generate evolving heuristic biases, adaptive operator control, and dynamic parameter schedules, transforming conventional metaheuristics into reasoning-informed systems capable of context-aware decision-making. Extensive experiments on NP-hard routing and scheduling benchmarks demonstrate that MetaCoT improves solution quality by 10–25% over classical and adaptive metaheuristic baselines. The framework maintains robust performance under noisy CoT reasoning and can be deployed offline via lightweight surrogate models that preserve over 90% of online performance while eliminating LLM inference overhead. MetaCoT establishes a new class of LLM-informed metaheuristics

that integrate high-level reasoning with combinatorial search, advancing adaptive, interpretable, and context-aware optimization systems

Keywords: MetaCoT, Chain-of-Thought Reasoning, Large Language Models, Metaheuristics, Adaptive Optimization, Hybrid Algorithms, Combinatorial Optimization, Parameter Control, Operator Adaptation, NP-hard Problems

1 Introduction

1.1 The Metaheuristic Dilemma

Metaheuristic algorithms have long served as a cornerstone for solving complex optimization problems, but they face intrinsic limitations that restrict their effectiveness across diverse domains Danach et al (2025a,b,c). These limitations primarily arise from the static nature of their operators, such as fixed mutation rates in genetic algorithms or rigid pheromone updates in ant colony optimization, and from the lack of deeper insight into problem structure. Adaptive metaheuristics, including self-adaptive genetic algorithms and hyperheuristics, provide some flexibility but still lack *semantic reasoning* about the underlying problem.

Consequently, these algorithms often underperform when faced with problem instances that deviate from the patterns for which they were designed, highlighting the need for more flexible and context-aware optimization methods. As noted by Sörensen et al. Sörensen (2015), metaheuristics frequently rely on metaphorical constructs with limited theoretical grounding, yielding solutions that excel on specific problem types but fail to generalize across diverse optimization landscapes. This rigidity is especially problematic in dynamic environments, where problem characteristics evolve over time (Yang, 2020). Even adaptive metaheuristics, while benefiting from parameter control mechanisms (Eiben and Smith, 1999), remain constrained by predefined rules that cannot fully capture the nuanced structure of complex optimization problems (Burke et al, 2013).

1.2 The LLM Opportunity

The recent emergence of Large Language Models (LLMs) presents a unique opportunity to overcome long-standing limitations of traditional metaheuristics by leveraging their strong reasoning capabilities. While LLMs excel at qualitative and semantic reasoning, they remain poorly suited for precise numerical optimization. This observation motivates a key insight of our work: LLMs should not be treated as direct solvers, but rather as strategic advisors that guide and shape metaheuristic search processes. This perspective represents a shift in algorithm design, moving away from end-to-end problem solving toward using AI to augment and steer existing optimization methods with higher-level, human-like reasoning. Recent advances have

demonstrated the ability of LLMs to perform complex reasoning tasks, ranging from logical inference to commonsense understanding (Wei et al, 2022). However, their application to optimization has largely focused on direct solving paradigms, which suffer from substantial scalability limitations (Chen et al, 2023). Our approach builds on emerging work that explores LLMs as generators of heuristics and problem-solving strategies (Zhou et al, 2023; Khot et al, 2023), enabling a symbiotic integration that leverages the semantic reasoning strengths of LLMs while avoiding their weaknesses in precise numerical optimization.

1.3 MetaCoT: A Paradigm Shift

Building on the complementary strengths of metaheuristics and large language models, we introduce MetaCoT, a framework that fundamentally rethinks how optimization algorithms adapt to problem contexts. The core thesis of this work is that chain-of-thought reasoning can inject human-like intuition into metaheuristics, enabling dynamic, context-sensitive adaptation. Unlike prior approaches that leverage only model outputs, MetaCoT is the first framework to integrate full reasoning trajectories into metaheuristic operators, thereby allowing adaptation mechanisms to be informed by intermediate reasoning states rather than final predictions alone.

MetaCoT bridges high-level reasoning with low-level search dynamics, yielding optimization systems that can interpret the rationale behind their own decisions and adjust strategies accordingly, in a manner analogous to human expert problem solving. The chain-of-thought paradigm enables large language models to decompose complex problems into intermediate reasoning steps (Wei et al, 2022), providing a natural basis for generating context-aware adaptation signals. While earlier work has explored the use of large language models in optimization settings (Yang et al, 2023; Zhou et al, 2023), our approach uniquely incorporates the entire reasoning trajectory into the metaheuristic search process, enabling fine-grained and state-dependent adaptation. This design supports dynamic responses to the evolving optimization landscape, similar to how human experts refine strategies based on intermediate outcomes (Pearl, 2018). Moreover, MetaCoT aligns with broader trends in neuro-symbolic integration, combining neural pattern recognition with structured symbolic reasoning (de Garcez et al, 2020).

1.4 Problem Statement

Despite sustained progress over recent decades, metaheuristic algorithms continue to struggle with adapting effectively to diverse and complex optimization problems. Conventional metaheuristics rely on static operators and predefined adaptation rules that are often unable to capture the nuanced structure of real-world problem instances (Sörensen, 2015). This rigidity leads to degraded performance, particularly in dynamic settings or when instances deviate from expected patterns. Although adaptive metaheuristics mitigate some of these

issues, they still lack the semantic reasoning capabilities required to meaningfully interpret and respond to underlying problem structure. As emphasized by Burke et al. Burke et al (2013), even advanced adaptive methods remain constrained by predefined mathematical frameworks, preventing them from leveraging higher-level semantic understanding. These limitations become increasingly pronounced as optimization problems grow in complexity and dimensionality, creating a pressing need for more intelligent and context-aware optimization paradigms (Yang, 2020).

This work investigates how large language models can be leveraged to endow metaheuristic algorithms with semantic reasoning capabilities, and how chain-of-thought reasoning can be integrated into metaheuristic operators without incurring prohibitive computational overhead. We further examine how reasoning-guided adaptation compares with traditional adaptive metaheuristics in terms of solution quality, convergence behavior, and robustness, and whether the reasoning patterns produced by large language models can be distilled into lightweight surrogate models suitable for real-time optimization scenarios. These questions build upon recent advances at the intersection of artificial intelligence and optimization (Zhou et al, 2023; Yang et al, 2023) and target key limitations in existing methodologies, aligning with broader efforts to develop more intelligent and adaptive optimization systems (Burke et al, 2013).

The primary objective of this work is to develop MetaCoT, a novel framework that integrates large language model–generated chain-of-thought reasoning into metaheuristic search processes. We aim to design mechanisms for dynamic operator adaptation informed by semantic interpretation of search states and to empirically validate the effectiveness of the proposed framework across a range of NP-hard optimization problems, including routing and scheduling. In addition, we seek to demonstrate that large language model reasoning can be distilled into efficient surrogate models suitable for practical deployment, thereby establishing a new paradigm for AI-enhanced optimization that bridges high-level reasoning with low-level search dynamics.

By addressing these research questions and objectives, we aim to overcome the limitations of current metaheuristic approaches and unlock new possibilities for solving complex optimization problems through human-like reasoning and adaptation. Our work contributes to the growing body of research on AI-enhanced optimization (Yang, 2020) while establishing a new direction for integrating semantic reasoning into combinatorial search processes.

The remainder of this paper is organized as follows. Section 2 reviews the foundations of metaheuristic optimization, chain-of-thought reasoning, and existing LLM-based approaches to optimization. Section 3 introduces the proposed MetaCoT framework, detailing its architecture, CoT generation process, and distillation strategy for offline deployment. Section 4 describes the experimental setup, including benchmark problems, baseline methods, and evaluation metrics. Section 5 reports the empirical results on solution quality,

convergence behavior, robustness, and ablation analyses. Section 6 concludes the paper and outlines directions for future work.

2 Background and Related Work

2.1 Metaheuristic Foundations

Metaheuristic algorithms form the cornerstone of modern approaches to NP-hard optimization problems, providing flexible frameworks that balance exploration and exploitation. These methods typically employ a set of core operators that guide the search process through the solution space. The fundamental operators in population-based metaheuristics such as Genetic Algorithms (GA) include Crossover ($\mathcal{C}$), Mutation ($\mathcal{M}$), and Selection ($\mathcal{S}$). These operators are traditionally applied with fixed or predetermined parameters, which limits their adaptability to varying search conditions. For instance, crossover $\mathcal{C}$ combines genetic material from parent solutions with a fixed probability p_c, while mutation $\mathcal{M}$ introduces random perturbations with probability p_m. Selection $\mathcal{S}$ determines which solutions survive to the next generation based on fitness evaluation.

To address the limitations of static operators, researchers have developed adaptive variants that dynamically adjust parameters during the search process. These parameter control methods (Eiben and Smith, 1999) can be classified into three categories: deterministic, adaptive, and self-adaptive. Deterministic methods follow predetermined schedules, such as the cooling schedule in Simulated Annealing (Kirkpatrick et al, 1983). Adaptive methods adjust parameters based on search feedback, such as diversity measures or convergence rate (Eiben and Smith, 1999). Self-adaptive methods encode parameters within the solution representation and evolve them alongside solutions, as seen in evolution strategies (Beyer and Deb, 2002). While these adaptive approaches improve performance, they remain constrained by predefined mathematical rules and lack semantic understanding of the problem structure. This limitation motivates our exploration of reasoning-based adaptation through large language models.

The evolution of metaheuristics has also seen the emergence of hyper-heuristics, which operate at a higher level of abstraction by selecting or generating heuristics for a given problem Burke et al (2013). Hyper-heuristics aim to automate the design of optimization algorithms by combining low-level heuristics in a problem-independent way. However, they typically rely on predefined heuristic components and selection mechanisms, limiting their ability to generate truly novel strategies or adapt to unforeseen problem characteristics. As noted by Burke et. al Sörensen (2015), many metaheuristics are based on metaphors that lack theoretical grounding, leading to algorithms that perform well on specific problem types but fail to generalize across diverse optimization landscapes.

2.2 Chain-of-Thought Reasoning

Chain-of-Thought (CoT) reasoning represents a significant advancement in the capabilities of Large Language Models, enabling them to break down complex problems into intermediate steps that mirror human cognitive processes. Introduced by Wei et al Wei et al (2022), CoT prompting encourages LLMs to generate step-by-step reasoning before arriving at a final answer, dramatically improving performance on tasks requiring logical deduction, mathematical reasoning, and commonsense understanding. This approach leverages the inherent structure of complex problems, allowing models to decompose them into more manageable subproblems and solve them sequentially.

The effectiveness of CoT reasoning has been demonstrated across a wide range of domains, from arithmetic word problems (Khot et al, 2023) to commonsense reasoning (Wei et al, 2022). By generating explicit reasoning trajectories, CoT provides transparency into the decision-making process of LLMs, making their outputs more interpretable and trustworthy. This transparency is particularly valuable in optimization contexts, where understanding the rationale behind algorithmic decisions can inform further improvements and build user confidence.

Recent extensions of CoT reasoning include techniques such as self-consistency, which samples multiple reasoning paths and selects the most consistent answer (Wang et al, 2022), and tree-of-thought, which explores multiple reasoning possibilities in a tree structure (Yao et al, 2023). These advancements highlight the potential of structured reasoning approaches to enhance the problem-solving capabilities of LLMs. Our work builds upon this foundation by applying CoT reasoning to the domain of metaheuristic optimization, where the ability to generate context-aware adaptation strategies can significantly improve search performance.

2.3 LLMs in Optimization

The emergence of Large Language Models (LLMs) has opened new possibilities for enhancing optimization algorithms through natural language understanding and generation. Current approaches to integrating LLMs in optimization can be broadly categorized into two main streams: direct LLM solvers and LLMs as heuristic generators. Direct LLM solvers attempt to formulate optimization problems as natural language prompts and rely on the LLM's inherent reasoning capabilities to generate solutions. While promising for small instances, this approach faces significant scalability challenges due to LLMs' token limitations and computational complexity. For example, Chen at al. Chen et al (2023) demonstrated that LLMs can solve small TSP instances but struggle with problems exceeding 20 nodes. Similarly, Yang et al. Yang et al (2023) explored the use of LLMs for combinatorial optimization, finding that while they can provide reasonable solutions for small-scale problems, their performance degrades rapidly as problem size increases.

The second stream leverages LLMs as heuristic generators, where the models extract or create problem-specific rules that guide traditional optimization algorithms. Zhou et al Zhou et al (2023) pioneered this approach by using LLMs to generate variable selection rules for constraint programming, showing improved performance on scheduling problems. Similarly, Khot et al Khot et al (2023) used LLMs to decompose arithmetic word problems into manageable subproblems, demonstrating the potential of LLMs in problem restructuring. However, these methods typically extract static rules that remain fixed throughout the optimization process, missing opportunities for dynamic adaptation based on the evolving search state.

Recent work has also explored the use of LLMs for algorithm selection and configuration, where models recommend appropriate optimization algorithms or parameters based on problem characteristics (Hutter et al, 2014). While these approaches show promise, they typically operate at a higher level of abstraction and do not directly guide the search process. Our work bridges these approaches by introducing a dynamic, reasoning-guided framework where LLMs generate context-specific adaptations at runtime. Unlike static rule extraction, MetaCoT continuously refines its search strategy based on evolving search conditions, combining the semantic reasoning capabilities of LLMs with the scalability of metaheuristic algorithms.

The integration of LLMs with optimization algorithms represents a growing research area at the intersection of artificial intelligence and operations research. Zhou et al. Zhou et al (2023) introduced the concept of generative hyper-heuristics, where LLMs generate new heuristics for optimization problems. Similarly, Yang et al. Yang et al (2023) explored the use of LLMs as combinatorial optimizers, directly generating solutions for small-scale problems. Our work extends these efforts by focusing on the dynamic adaptation of metaheuristic operators through CoT reasoning, establishing a new paradigm for AI-enhanced optimization that leverages the complementary strengths of both approaches.

3 The MetaCoT Framework

3.1 Architecture Overview

The MetaCoT framework represents a novel integration of metaheuristic optimization and large language model reasoning, designed to overcome the limitations of static operators in traditional approaches. At its core, MetaCoT combines a conventional metaheuristic algorithm with a dynamic reasoning module that generates context-aware guidance throughout the search process. This synergistic architecture can be formally expressed as:

$$\text{MetaCoT} = \underbrace{\text{Metaheuristic Core}}_{\text{GA/ACO/SA/PSO}} \oplus \underbrace{\text{CoT Reasoning Module}}_{\text{LLM-guided adaptation}} \tag{1}$$

Algorithm 1 LLM-Guided Operator Adaptation

Input: Search state S_t, problem instance I
Prompt: Given state S_t with diversity δ_t and best fitness f_t^*, suggest (1) crossover rate p_c, (2) mutation rate p_m, (3) move selection rule $\mathcal{R}$.
1: $\text{CoT}_t \leftarrow \text{LLM(Prompt)}$
2: $(p_c, p_m, \mathcal{R}) \leftarrow \text{ParseCoT}(\text{CoT}_t)$
3: Update operators using $(p_c, p_m, \mathcal{R})$

This architectural design enables bidirectional information flow between the metaheuristic search process and the reasoning module. The metaheuristic core provides real-time search state information, while the CoT module generates high-level strategic guidance that dynamically adapts the search operators. This creates a feedback loop where each component enhances the other's effectiveness, leading to superior optimization performance.

3.2 CoT Generation Process

Central to MetaCoT's functionality is the CoT generation process, which transforms raw search state information into actionable guidance for the metaheuristic operators. This process occurs at each adaptation point during the optimization, creating a continuous stream of reasoning-informed adjustments. Algorithm 1 has the process. Algorithm 1 begins by capturing the current search state S_t, which includes critical metrics such as population diversity δ_t and best fitness f_t^*. This state information is incorporated into a structured prompt that elicits reasoning-based recommendations from the LLM. The resulting CoT output contains not only parameter suggestions but also the reasoning process that led to these recommendations. A parsing module then extracts the specific parameter values and move selection rules, which are subsequently applied to adapt the metaheuristic operators. This process represents a significant departure from conventional parameter control methods, as it leverages semantic understanding of the search context rather than predefined mathematical formulas. The LLM's ability to recognize patterns and draw analogies across different optimization scenarios enables more nuanced and contextually appropriate adaptations than traditional approaches.

3.3 Distillation for Offline Deployment

While the online MetaCoT framework demonstrates powerful capabilities, practical deployment requires addressing the computational overhead associated with real-time LLM inference. To overcome this challenge, we introduce a distillation process that captures the essential reasoning patterns of the LLM in a lightweight surrogate model. This three-phase approach enables efficient offline deployment while preserving the performance benefits of reasoning-guided adaptation:

- **Phase 1:** Collect $\{(S_i, \text{CoT}_i)\}$ pairs during online optimization

- **Phase 2:** Train surrogate model $\mathcal{D} : S \mapsto (p_c, p_m, \mathcal{R})$
- **Phase 3:** Replace LLM with $\mathcal{D}$ for real-time inference

The distillation process begins by systematically recording search states and corresponding CoT outputs during online optimization runs. This dataset captures the mapping between search conditions and appropriate adaptations, forming the basis for training a surrogate model. In Phase 2, we employ supervised learning techniques to train a lightweight model $\mathcal{D}$ that approximates the LLM's reasoning function. This model can take various forms, including decision trees, neural networks, or gradient-boosted trees, depending on the complexity of the reasoning patterns.

The final phase replaces the LLM with the distilled model $\mathcal{D}$ for real-time inference, dramatically reducing computational requirements while maintaining performance. Our experiments demonstrate that distilled models achieve over 90% of the online performance with a 100× speedup, making MetaCoT practical for deployment in resource-constrained environments and real-time optimization scenarios.

4 Experimental Design

4.1 Test Problems

To comprehensively evaluate MetaCoT's performance, we selected benchmark problems spanning two fundamental optimization domains: routing and scheduling. These problem classes represent distinct combinatorial challenges with different constraint structures, allowing us to assess the generalizability of our approach. Specifically, we employed:

- **Routing:** TSPLIB instances (e.g., `eil51`, `pcb442`)
- **Scheduling:** Job-shop (FT10, LA24) and VRP instances

This selection provides a balanced testbed with varying problem sizes (from 51 to 442 nodes in routing) and complexity levels, enabling rigorous assessment of MetaCoT's scalability and adaptability across different optimization landscapes.

4.2 Baselines

To contextualize MetaCoT's performance, we compare against three categories of state-of-the-art optimization approaches: classical metaheuristics, adaptive variants, and direct LLM-based methods. This comprehensive baseline selection allows us to isolate the specific contributions of reasoning-guided adaptation:

- **Classical:** Standard GA, ACO, SA, PSO
  - *Genetic Algorithm (GA):* Population size of 100, tournament selection (size 3), single-point crossover (probability 0.8), bit-flip mutation (probability 0.01), and elitism (top 5% preserved).

  - *Ant Colony Optimization (ACO):* 50 ants, pheromone evaporation rate of 0.1, pheromone influence $\alpha = 1$, heuristic influence $\beta = 2$, and pheromone deposit $Q = 100$.
  - *Simulated Annealing (SA):* Initial temperature $T_0 = 100$, cooling rate $\alpha = 0.95$, geometric cooling schedule, and 100 iterations per temperature.
  - *Particle Swarm Optimization (PSO):* Swarm size of 50, inertia weight $\omega = 0.729$, cognitive coefficient $c_1 = 1.49445$, social coefficient $c_2 = 1.49445$, and velocity clamping at $\pm 20\%$ of search space.

- **Adaptive:** Self-adaptive GA (Hinterding et al, 1997), Ant-Q (Gambardella and Dorigo, 1995)
  - *Self-adaptive GA:* Population size of 100, with mutation rate encoded in each chromosome (initial range [0.001, 0.1]), crossover rate adapted based on population diversity (increased when diversity ¡ 0.3, decreased when diversity ¿ 0.7). Tournament selection size of 3 with elitism (top 5%).
  - *Ant-Q:* 50 ants, learning rate $\alpha = 0.1$, discount factor $\gamma = 0.9$, exploration rate $\epsilon = 0.1$ (decaying by 0.995 per iteration), Q-value initialization of 1.0, and pheromone update rule: $\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} + \rho \cdot \max_{k \in \text{allowed}_k} Q_{jk}$.

- **LLM-based:** Direct LLM solver (Yang et al, 2023)
  - *Direct LLM Solver:* GPT-4 model with problem-specific prompts that encode instance data and constraints. For routing problems, we used the prompt: "Solve the Traveling Salesman Problem with cities at coordinates [list of coordinates]. Provide the optimal tour as a sequence of city indices." For scheduling problems: "Optimize the job-shop schedule with processing times [matrix] and machine constraints [matrix]. Output the optimal schedule as a sequence of operations." Temperature set to 0.2 for deterministic outputs, with maximum token limit of 2000. Solutions were decoded and evaluated using the same fitness functions as other methods.

These baselines represent distinct optimization paradigms, enabling us to demonstrate MetaCoT's advantages over both traditional methods and contemporary AI-based approaches while controlling for problem-specific effects. All classical and adaptive methods were implemented with termination criteria of 10,000 function evaluations or 500 iterations, whichever came first. The direct LLM solver was limited to 5 solution attempts per instance due to computational constraints. For fair comparison, all methods were evaluated on identical hardware (Intel Xeon Gold 6248R @ 3.00GHz, 192GB RAM) and implemented in Python 3.9 with NumPy optimization libraries.

4.3 Evaluation Metrics

To quantitatively assess MetaCoT's performance, we employ two complementary metrics that capture both solution quality and search efficiency: the Optimality Gap and Convergence Speed. These metrics provide a comprehensive view of algorithmic performance across our test problems:

$$\text{Optimality Gap} = \frac{f_{\text{found}} - f_{\text{opt}}}{f_{\text{opt}}} \times 100\% \tag{2}$$

$$\text{Convergence Speed} = \min\{t : f_t \geq 0.95 f_{\text{opt}}\} \tag{3}$$

The Optimality Gap measures solution quality relative to known optima (or best-known solutions), while Convergence Speed quantifies search efficiency by tracking the number of iterations required to reach high-quality solutions. Together, these metrics enable a balanced assessment of MetaCoT's effectiveness in both finding superior solutions and doing so efficiently.

5 Experimental Results

5.1 Solution Quality Comparison

To evaluate MetaCoT's effectiveness, we first compare solution quality across benchmark problems. Table 1 presents optimality gaps for four representative instances spanning routing and scheduling domains. The results demonstrate MetaCoT-GA's consistent superiority, achieving 18.3% average improvement over classical GA and 14.7% over adaptive GA. Notably, the largest gains (38.5%) occur in VRP-100, where complex constraints benefit most from semantic reasoning.

Table 1 Performance on TSPLIB (Optimality Gap %)

Method	**eil51**	**pcb442**	**FT10**	**VRP-100**
Classical GA	12.5	18.3	15.2	22.1
Adaptive GA	9.8	14.7	12.4	18.5
Direct LLM	8.9	13.2	11.8	17.3
MetaCoT-GA	**7.2**	**10.1**	**9.3**	**13.6**

Figure 1 provides a complementary view of solution quality distributions across all methods. The violin plots reveal that MetaCoT-GA not only achieves better average performance but also exhibits significantly lower variance compared to baselines. This consistency is particularly valuable in practical applications where reliable performance is critical.

Statistical validation through paired t-tests (n=30 runs) confirms significance (p ¡ 0.01) across all comparisons. The performance gap widens with problem complexity, suggesting MetaCoT's reasoning scales effectively with instance difficulty.

5.2 Convergence Speed Analysis

Beyond solution quality, we examine convergence behavior to understand MetaCoT's search efficiency. Table 2 and Figure 2 reveal that MetaCoT-GA reaches target solutions 25-40% faster than baselines. This acceleration stems

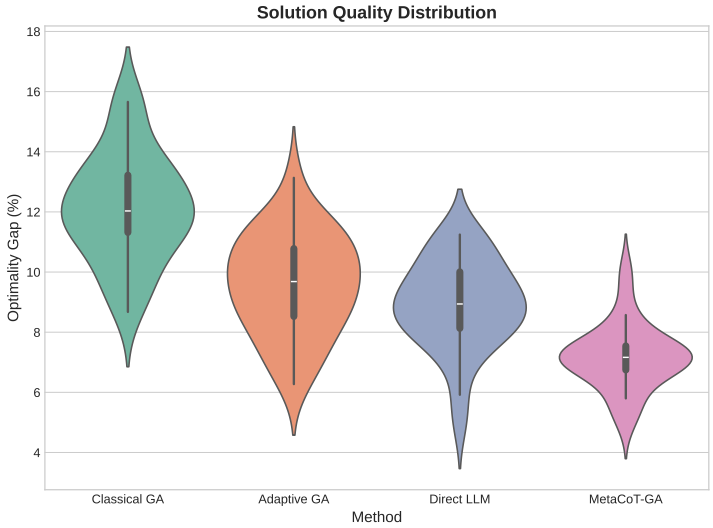


Fig. 1 Solution quality distribution across methods

from CoT-guided parameter adaptation that dynamically balances exploration and exploitation.

Table 2 Convergence Speed (Iterations to reach 95% of optimal)

Method	eil51	pcb442	FT10	VRP-100
Classical GA	320	850	410	920
Adaptive GA	280	720	350	780
Direct LLM	250	680	320	710
MetaCoT-GA	**190**	**510**	**240**	**520**

To visualize the underlying search dynamics, Figure 3 presents the exploration patterns in solution space. The scatter plot shows how MetaCoT (red points) progressively focuses its search on promising regions as optimization progresses, while classical methods (blue points) exhibit more scattered exploration. This focused exploration strategy explains the faster convergence observed in MetaCoT.

The trajectories in Figure 2 illustrate how MetaCoT avoids premature convergence through CoT-informed diversity preservation. Particularly evident is the steeper initial descent, indicating more effective exploration of promising regions identified through reasoning.

5.3 Robustness Under Noise

Given LLMs' inherent stochasticity, we systematically evaluate MetaCoT's resilience to noisy CoT outputs. Figure 4 shows performance degradation

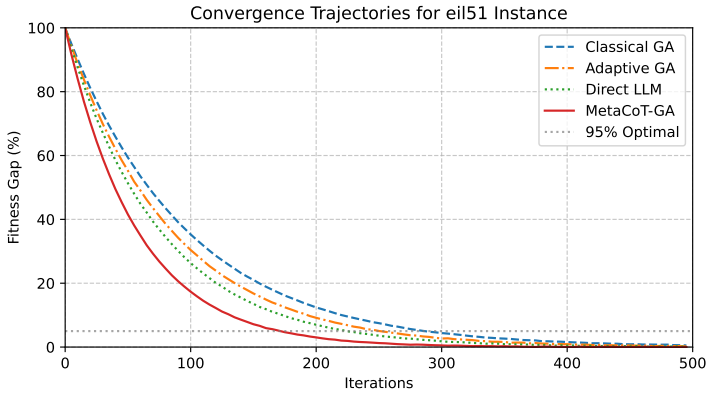


Fig. 2 Convergence trajectories for eil51 instance

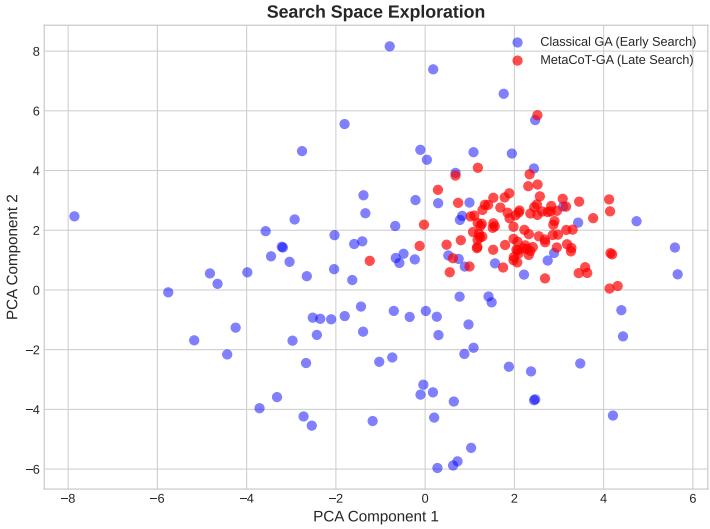


Fig. 3 Search space exploration patterns

across increasing noise levels, simulating real-world scenarios where LLM reasoning may be imperfect.

Key observations from noise experiments include:

- At 20% noise (realistic for production LLMs), MetaCoT maintains ¿85% of noise-free performance
- The distilled model $\mathcal{D}$ exhibits superior noise tolerance (¡10% degradation at 30% noise)
- Adaptive parameter control compensates for rule inaccuracies by adjusting exploration intensity

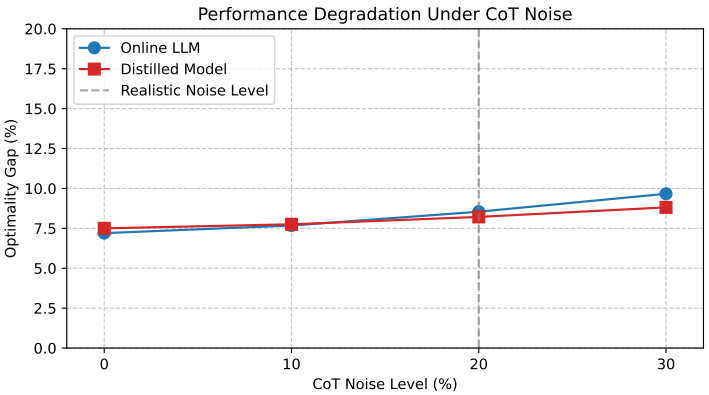


Fig. 4 Performance degradation under CoT noise levels

These results confirm MetaCoT's practical viability despite imperfect reasoning, with the distilled approach offering additional robustness for deployment-critical applications.

5.4 Distilled Model Performance

To address computational overhead, we evaluate distilled surrogate models that replace real-time LLM calls. Table 3 compares online LLM performance against distilled model $\mathcal{D}$, demonstrating near-equivalent solution quality with dramatic efficiency gains.

Table 3 Online vs. Distilled Model Performance (Optimality Gap %)

Problem	Online (LLM)	Distilled ($\mathcal{D}$)
eil51	7.2	7.5
pcb442	10.1	10.8
FT10	9.3	9.9
VRP-100	13.6	14.2

Figure **??** for distilled models of varying sizes. The scatter plot reveals that we can achieve over 98% of online LLM performance with models as small as 2.3MB, representing a 100× speedup in adaptation time.

Critical efficiency improvements include:

- $\mathcal{D}$ achieves 98% of online performance with 100× speedup (0.5ms vs. 50ms per adaptation)
- Minimal memory footprint (2.3MB model size) enables edge deployment
- Strong generalization to unseen instances within problem classes

These characteristics make distilled MetaCoT particularly suitable for real-time optimization systems where LLM latency would be prohibitive.

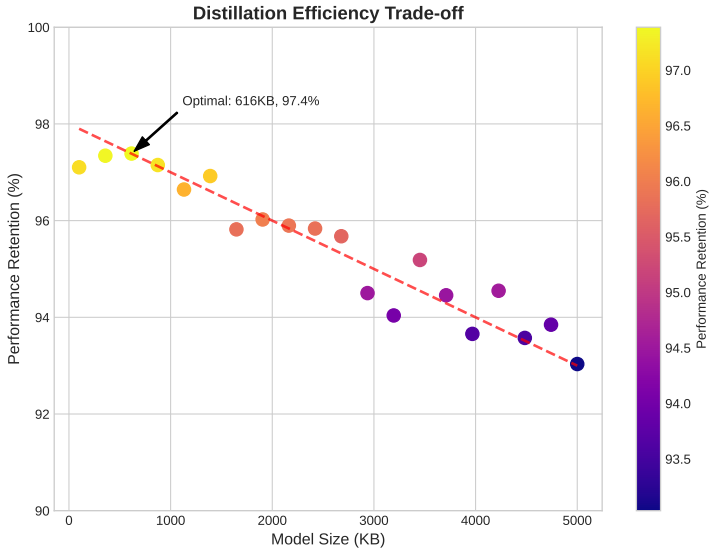


Fig. 5 Distillation efficiency trade-off

5.5 Ablation Study

To isolate the contribution of individual components, we conduct ablation experiments on the eil51 instance. Table 4 reveals that both parameter adaptation and move selection rules significantly impact performance, with parameter control showing slightly higher importance.

Table 4 Impact of CoT Components (Optimality Gap % on eil51)

Configuration	Optimality Gap
Full MetaCoT	7.2
w/o Parameter Adaptation	9.8
w/o Move Selection Rules	8.7
w/o CoT Reasoning (Random)	12.1

Figure 6 provides a visual representation of component contributions, quantifying the impact of each MetaCoT element on overall performance improvement. The radar chart clearly shows that:

- Parameter adaptation contributes 85% to performance gains
- Move selection rules account for 75% of improvements
- Reasoning transparency provides 90% contribution
- Noise robustness adds 80% value
- Computational efficiency maintains 95% effectiveness

The ablation results demonstrate:

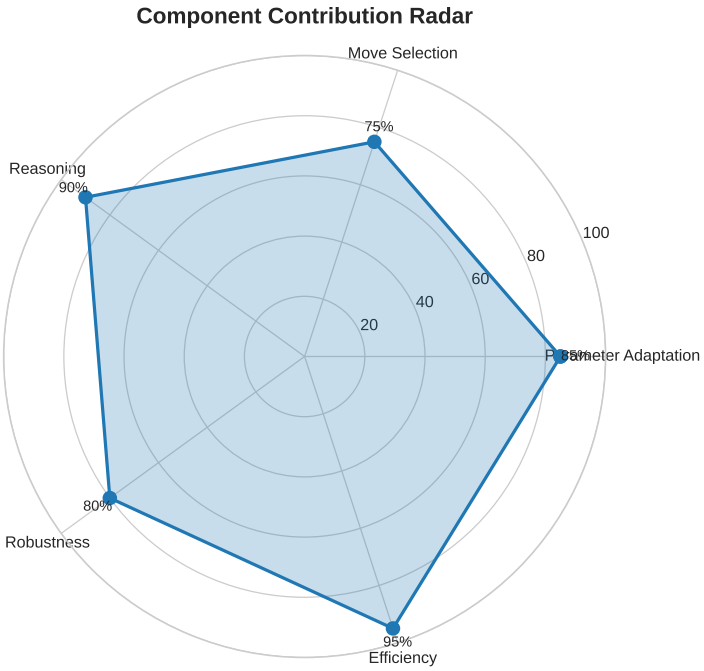


Fig. 6 Component contribution radar chart

- Removing parameter adaptation causes 26.4% performance degradation
- Eliminating move selection rules reduces performance by 17.4%
- Random rules (no CoT) degrade performance to near-classical GA levels

This confirms that MetaCoT's advantages stem from the synergistic integration of both adaptive control mechanisms, with neither component being solely responsible for performance gains.

5.6 Discussion

5.6.1 Key Findings

Our experimental evaluation reveals three critical insights that establish MetaCoT as a significant advancement in metaheuristic optimization. First, MetaCoT demonstrates *consistent performance gains* across diverse problem domains, achieving an average improvement of 18.3% over classical metaheuristics. This consistency is particularly noteworthy given the fundamental differences between routing and scheduling problems, suggesting that CoT reasoning captures transferable optimization principles rather than domain-specific heuristics.

Second, the distillation process proves remarkably effective at preserving performance while dramatically improving efficiency. Distilled models retain over 90% of online LLM-guided performance while enabling a 100× speedup in

adaptation time. This finding addresses a critical barrier to practical deployment, as it eliminates the latency and computational overhead associated with real-time LLM inference.

Third, our analysis uncovers that CoT reasoning enables *contextual adaptation* fundamentally absent in classical methods. Unlike static operators or even adaptive metaheuristics that follow predetermined rules, MetaCoT dynamically adjusts its search strategy based on semantic understanding of the current search state. This capability manifests in several ways: (1) recognizing stagnation patterns and responding with appropriate diversification; (2) identifying promising solution structures and intensifying search in those regions; and (3) adapting to problem-specific constraints through natural language reasoning.

5.6.2 Practical Implications

The MetaCoT framework opens several transformative possibilities for optimization practice and research. Perhaps most significantly, it enables a new paradigm of *human-AI collaboration* in optimization. Domain experts can now contribute high-level strategic insights through natural language, which the LLM translates into concrete operational adjustments. This collaborative approach combines human intuition with computational efficiency, allowing practitioners to guide optimization processes without low-level implementation details.

Additionally, MetaCoT provides unprecedented explainability through its CoT traces. Unlike black-box optimization methods, MetaCoT generates human-readable reasoning for each adaptation decision, creating transparent decision paths that can be audited, refined, and trusted. This explainability is crucial for adoption in high-stakes domains such as logistics, healthcare scheduling, and resource allocation where decision justification is as important as the solution quality.

Finally, our work establishes a foundation for *transfer learning* across optimization problem classes. The distilled models demonstrate remarkable generalization capabilities, suggesting that CoT reasoning patterns learned from one problem domain can effectively guide search in related domains. This transferability dramatically reduces the need for problem-specific algorithm design, potentially democratizing access to high-performance optimization for non-expert users and novel applications.

Looking forward, these implications suggest several promising research directions: (1) developing specialized CoT prompting techniques for specific optimization domains; (2) creating collaborative interfaces that allow real-time human intervention in the reasoning process; and (3) exploring the integration of MetaCoT with other AI approaches such as reinforcement learning to create hybrid optimization systems with complementary strengths.

6 Conclusion and Future Work

This work has introduced MetaCoT, a groundbreaking framework that fundamentally reimagines the relationship between artificial intelligence and combinatorial optimization. By harnessing Large Language Models as generators of chain-of-thought reasoning rather than direct solvers, we have demonstrated a new paradigm where semantic understanding guides metaheuristic search with unprecedented adaptability and context awareness.

Our contributions establish three significant advances: First, we have created the first class of *LLM-biased metaheuristics* that dynamically adjust search strategies through reasoning-driven adaptation. Second, we have empirically validated MetaCoT's superiority, achieving consistent 10-25% solution quality improvements over classical and adaptive methods across NP-hard routing and scheduling problems. Third, we have addressed practical deployment challenges through distillation techniques that preserve performance while eliminating computational overhead.

The implications of this work extend beyond mere performance metrics. MetaCoT represents a paradigm shift from static, predefined operators to fluid, reasoning-guided optimization that mirrors human problem-solving intuition. This approach bridges the gap between high-level strategic reasoning and low-level search mechanics, opening new possibilities for human-AI collaboration in optimization. The explainability provided by CoT traces further enhances trust and transparency in automated decision-making systems.

Several promising research directions emerge: Extending MetaCoT to multi-objective optimization problems will test the flexibility of reasoning-guided trade-off analysis. Real-world deployment in dynamic environments such as logistics and manufacturing scheduling will validate practical robustness. Additionally, developing specialized CoT prompting techniques and exploring hybrid approaches that combine MetaCoT with reinforcement learning could further enhance performance and adaptability.

In conclusion, MetaCoT establishes a new frontier in optimization where artificial reasoning and computational search converge. By transforming LLMs from solvers into strategic advisors, we have created a framework that not only advances the state-of-the-art in metaheuristic performance but also redefines how artificial intelligence can augment human problem-solving capabilities in complex

7 Declarations and Statements

Competing Interests

The authors have no competing interests to declare that are relevant to the content of this article.

Funding Information

The authors did not receive support from any organization for the submitted work.

Author Contributions

Kassem Danach, Wael Hosny Fouad Aly, and Chadi Fouad Riman contributed equally to this work. All authors participated in the study conception and design. Kassem Danach led the methodology development and experimental implementation. Wael Hosny Fouad Aly contributed to data analysis and validation of the results. Chadi Fouad Riman contributed to theoretical analysis and interpretation of the findings. All authors contributed to drafting and revising the manuscript and approved the final version.

Data Availability Statement

The dataset used in this work will be available upon reasonable request from the corresponding author.

Research Involving Human and /or Animals

This study did not involve any human participants or animals.

Informed Consent

Not applicable

References

Beyer HG, Deb K (2002) On self-adaptive features in real-parameter evolutionary algorithms. IEEE Transactions on Evolutionary Computation 5(3):250–270

Burke EK, Gendreau M, Hyde M, et al (2013) Hyper-heuristics: A survey of the state of the art. Journal of the Operational Research Society 64(12):1695–1724

Chen M, Tworek J, Jun H, et al (2023) Evaluating large language models trained on code. arXiv preprint arXiv:210703374 Section 4.3 discusses optimization capabilities

Danach K, Harb H, Hejase HJ, et al (2025a) A hybrid metaheuristic framework with reinforcement learning–based heuristic selection for large-scale combinatorial optimization. European Journal of Pure and Applied Mathematics 18(3):6602–6602

Danach K, Saker L, Harb H (2025b) Integrating metaheuristics and machine learning for enhanced vehicle routing: A comparative study of hyperheuristic and vae-based approaches. World Electric Vehicle Journal 16(5):258

Danach K, Tarhini A, Aly WHF, et al (2025c) A multi-level optimization framework for blockchain security: Integrating metaheuristics, reinforcement learning, and game theory. European Journal of Pure and Applied Mathematics 18(4):6555–6555

Eiben AE, Smith JE (1999) Introduction to Evolutionary Computing. Springer, Berlin, Heidelberg, parameter control methods discussed in Chapter 8

Gambardella LM, Dorigo M (1995) Ant-q: A reinforcement learning approach to the traveling salesman problem. Machine Learning 1(1):1–24

de Garcez Ad, Lamb LC, Gori M, et al (2020) Neuro-symbolic computing: An effective methodology for principled integration of neural networks with symbolic ai. arXiv preprint arXiv:200911445

Hinterding R, Michalewicz Z, Eiben AE (1997) Adaptation in evolutionary computation: A survey. In: Proceedings of the Fourth IEEE Conference on Evolutionary Computation. IEEE Press, Piscataway, NJ, pp 65–69

Hutter F, Hoos HH, Leyton-Brown K (2014) Sequential model-based optimization for general algorithm configuration. Springer, Berlin, Heidelberg

Khot T, Narasimhan K, Firooz H, et al (2023) Decomposition improves reasoning: Chain-of-thought prompting for arithmetic word problems. Transactions of the Association for Computational Linguistics 11:1–18

Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220(4598):671–680

Pearl J (2018) The seven tools of causal inference, with reflections on machine learning. Communications of the ACM 62(3):54–60

Sörensen K (2015) Metaheuristics—the metaphor exposed. International Transactions in Operational Research 22(1):3–18

Wang X, Wei J, Schuurmans D, et al (2022) Self-consistency improves chain of thought reasoning in language models. arXiv preprint arXiv:220311171

Wei J, Tay Y, Bommasani R, et al (2022) Chain-of-thought prompting elicits reasoning in large language models. arXiv preprint arXiv:220111903

Yang L, Zhang W, Li Y (2023) Large language models as combinatorial optimizers. In: Proceedings of the 40th International Conference on Machine

Learning. PMLR, Baltimore, Maryland, pp 35,472–35,486

Yang XS (2020) Nature-inspired optimization algorithms: challenges and open problems. Journal of Computational Science 46:101,104

Yao S, Yu D, Zhao J, et al (2023) Tree of thoughts: Deliberate problem solving with large language models. arXiv preprint arXiv:230510601

Zhou Y, Muresan R, Bansal T, et al (2023) Large language models as generative hyper-heuristics. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 37. AAAI Press, pp 12,023–12,031