Dynamic Adaptive Reasoning: Optimizing LLM Inference-Time Thinking via Intent-Aware Scheduling

Arthit Wongsawat * and Juncheng Gai

Posted Date: 5 November 2025

doi: 10.20944/preprints202511.0291.v1

Keywords: Intent-Aware Reasoning; Chain-of-Thought; large language models

Article

Dynamic Adaptive Reasoning: Optimizing LLM Inference-Time Thinking via Intent-Aware Scheduling

Arthit Wongsawat * and Juncheng Gai

Rajamangala University of Technology Thanyaburi, Thailand
* Correspondence: 1164108010292@mail.rmutt.ac.th

Abstract

Recent progress in complex reasoning tasks has revealed the potential of large language models driven by Chain-of-Thought (CoT) processes. However, their inference-time reasoning often suffers from inefficiency and limited adaptability. This work introduces the Intent-Aware Reasoning Scheduler (IARS), a framework that dynamically refines inference-time reasoning by enabling models to perceive and adjust their own reasoning intent. IARS integrates an independent Intent-Aware Scheduler (IAS) that continuously analyzes generated thought tokens, identifying reasoning states such as exploring, confirming, ambiguous, or near-answer. Based on these states, IAS issues adaptive directives to modulate reasoning depth and style. The approach requires no retraining, operating purely during inference through lightweight decoding and prompt adjustments. Experiments on diverse reasoning benchmarks show that IARS achieves higher reasoning quality with fewer tokens, while human evaluation and ablation studies confirm its interpretability and efficiency. The results demonstrate that intent-aware scheduling provides a more adaptive and effective mechanism for steering model reasoning.

Keywords: Intent-Aware Reasoning; Chain-of-Thought; large language models

1. Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in tackling complex reasoning tasks, including mathematical problem-solving, code generation, and scientific inquiry, often achieving strong generalization across various domains [1,2]. These advanced capabilities often stem from the models' ability to engage in "thinking" or "Chain-of-Thought" (CoT) reasoning before producing a final answer, thereby mimicking human-like step-by-step deduction [3]. This "slow thinking" paradigm, while powerful, is not always applied with optimal efficiency or effectiveness during the inference stage.

Existing methodologies have begun to explore dynamic control over the model's thinking process at inference time. For instance, the AlphaOne framework [4] pioneered the concept of inserting specific tokens (e.g., "wait") to extend the deliberation phase and then switching to a faster thinking mode at predefined "α-moments" to balance efficiency and accuracy. This work highlights the significant potential of dynamically regulating an LLM's reasoning trajectory during inference. However, current inference-time modulation strategies, which often rely on fixed thinking budgets or heuristic-based switching mechanisms, may not fully capitalize on the model's internal "intent" or "confidence" at various stages of reasoning. This aligns with recent research emphasizing the need for agents to perform meta-verification and trustworthy reasoning based on their internal states to resolve conflicts and enhance reliability [5]. During problem-solving, an LLM might be highly confident in a particular step, or conversely, it might be struggling and require deeper exploration or alternative paths. We posit that by enabling the model to **perceive its current thinking intent and progress state** during inference and subsequently **adapt its thinking depth and strategy**, we can significantly enhance both the efficiency and accuracy of complex reasoning tasks.

In this paper, we introduce a novel and more refined dynamic inference-time reasoning modulation framework, which we term the **Intent-Aware Reasoning Scheduler (IARS)**. Unlike prior approaches that primarily focus on extending thinking duration or applying fixed switching rules, IARS actively monitors the model's internal reasoning process and dynamically adjusts its thinking strategy based on its perceived "intent." Our framework introduces a lightweight, independent "Intent-Aware Scheduler" that analyzes the generated thought tokens in real-time. Based on this analysis, it assesses the model's current state (e.g., "exploring," "confirming," "stuck," or "near-answer") and issues dynamic scheduling directives (e.g., `[DEEPER_THINK]`, `[STREAMLINE_THINK]`, `[CONCLUDE_NOW]`, `[RETRY_SUBPROBLEM]`). These directives, implemented by modifying the LRM's decoding process or through prompt engineering, guide the model to either deepen its thoughts, streamline its reasoning, or directly conclude its answer, without requiring any re-training of the base LLM. This dynamic adaptation aims to improve not only accuracy but also computational efficiency, a goal shared by methods that optimize inference through techniques like co-adaptive token and neuron pruning [6].

To validate the efficacy of our proposed IARS framework, we conduct extensive experiments using several prominent open-source large reasoning models, including DeepSeek-R1-Distill-Qwen-1.5B, DeepSeek-R1-Distill-Qwen-7B, and Qwen QwQ-32B. Our evaluation spans six diverse and challenging reasoning benchmarks: AIME 2024 (AIME24), AMC 2023 (AMC23), Minerva-Math, MATH500 for mathematical reasoning; LiveCodeBench (LiveCode) for code generation; and OlympiadBench (Olympiad) for scientific reasoning. The rigorous evaluation of such dynamic decision-making systems is a critical research area, not only for LLMs but also in fields like autonomous driving [7]. We employ standard evaluation metrics, namely Pass@1 accuracy and the average number of generated tokens (#Tk), to assess both the correctness and efficiency of the reasoning process. Our experimental results demonstrate that IARS consistently outperforms baseline models and strong existing methods like Chain-of-Density (CoD) and even the AlphaOne framework across all benchmarks. Specifically, on the DeepSeek-R1-Distill-Qwen-1.5B model, IARS achieves a Pass@1 accuracy of **32.5%** on AIME24 and **22.0%** on LiveCode, representing significant improvements over baseline performance while simultaneously reducing the average token generation count, signifying a more efficient and targeted reasoning process.

The main contributions of this paper are summarized as follows:

- We propose the **Intent-Aware Reasoning Scheduler (IARS)**, a novel framework that enables large language models to dynamically adjust their thinking depth and strategy during inference by perceiving their internal reasoning intent and progress state.
- We introduce the concept of an **Intent-Aware Scheduler** that monitors the LLM's generated thought tokens in real-time and issues dynamic, context-specific scheduling directives, significantly enhancing the adaptability of inference-time reasoning.
- We demonstrate through extensive experiments on diverse reasoning benchmarks that IARS consistently achieves superior accuracy and efficiency compared to existing state-of-the-art inference-time control methods, all without requiring any re-training of the base large language model.

2. Related Work

2.1. Large Language Model Reasoning and Chain-of-Thought

Research in Large Language Model (LLM) reasoning, particularly via Chain-of-Thought (CoT), aims to improve model capabilities, with a significant goal being weak-to-strong generalization [2]. Early work analogous to CoT used contrastive prompts to generate human-interpretable explanations, enhancing both commonsense reasoning and interpretability [8]. To advance structured reasoning, specialized datasets like FinQA were developed for complex multi-step numerical reasoning over financial data [9]. Frameworks such as the Three-hop Reasoning (THOR) CoT have been proposed to decompose complex tasks like implicit sentiment analysis, thereby improving performance by mimicking human thought processes [10]. These principles of structured reasoning are mirrored in other AI fields, including boosting for fraud detection [11], reinforcement learning in anti-money laundering [12], causal modeling for financial security [13], and logistics optimization [14].

Insights into the inferential steps required for robust reasoning can be drawn from related areas like trainable subgraph retrievers for knowledge base question answering (KBQA) [15]. The step-by-step nature of CoT is comparable to structured decision-making algorithms like Monte Carlo Tree Search (MCTS) in multi-agent systems [16,17], and to sequential processing in computer vision for 3D motion capture [18] and collaborative depth estimation [19]. To improve efficiency, smaller models can be fine-tuned on CoT examples generated by larger "reasoning teachers," sometimes even outperforming the original teacher model [20]. However, the effectiveness of CoT is ultimately constrained by the inherent difficulty of the problem and the model's fundamental capabilities [21]. To push these boundaries, methods like "Formula Tuning" train LLMs to generate executable spreadsheet formulas, significantly enhancing performance on complex numerical and symbolic reasoning tasks [22]. LLMs are also being applied to specialized generative tasks, such as personalized architectural design [23], controllable 3D urban block generation [24], and creative digital tools for embroidery [25]. In the multimodal domain, research is exploring visual in-context learning [26] and abnormal-aware feedback for medical applications [27]. Finally, the critical importance of high-quality, well-aligned pretraining data has been demonstrated for achieving state-of-the-art performance, especially in code generation [28].

2.2. *Dynamic Inference-Time Control and Adaptive Thinking for LLMs*

Developing dynamic inference-time controls is crucial for enhancing LLM flexibility and efficiency. A key direction is enabling agents with meta-verification to handle internal reasoning conflicts [5]. Techniques like DExperts dynamically steer text generation by combining expert and anti-expert models at decoding time [29], while methods such as DR-BERT adaptively re-weight sentence regions for improved sentiment understanding [30]. This concept of dynamic adaptation is analogous to online parameter identification in control engineering [31–33]. From an efficiency standpoint, dynamic control includes co-adaptive token and neuron pruning for vision-language models [6] and device-generic latency modeling for mobile neural architecture search [34].

In multimodal contexts, frameworks like MRL enable adaptive, agent-based reasoning between modalities [35], moving beyond fixed fusion strategies seen in earlier work on attentive graph convolutional networks [36]. Similarly, the CTRLsum framework allows users to control summarization aspects at inference time through prompts, boosting efficiency [37]. Adaptive thinking also encompasses model self-improvement through methods like strategic self-play [38], a concept that finds parallels in other scientific domains such as real-time material analysis [39] and electrocatalysis [40]. For fine-grained control, the EDGE framework leverages semantic frames to guide dialogue generation and prevent literal copying [41]. The need for such adaptive mechanisms may vary with task complexity, a principle also observed in the pretraining of medical vision models [42]. The advancement of these control mechanisms requires robust evaluation criteria for interactive autonomous systems [7]. Comprehensive evaluations of LLMs on tasks like sentiment analysis provide crucial insights into performance trade-offs, informing how computational resources can be dynamically allocated to meet varied demands [43].

3. Method

In this section, we present the **Intent-Aware Reasoning Scheduler (IARS)** framework, designed to dynamically optimize the thinking process of Large Reasoning Models (LRMs) during inference. Unlike previous approaches that rely on static rules, fixed budgets for controlling reasoning depth, or pre-defined switching heuristics, IARS introduces a novel mechanism for real-time monitoring of the LRM's internal thought process. This enables adaptive adjustments to its reasoning strategy based on perceived intent, leading to more efficient and accurate problem-solving.

3.1. Overview of the IARS Framework

The core objective of IARS is to empower LRMs with a more intelligent and adaptive "internal mentor" that guides their step-by-step reasoning. Traditional Chain-of-Thought (CoT) methods or even more advanced inference-time modulation techniques often employ pre-defined or heuristic-driven switching points between "slow" and "fast" thinking modes. IARS departs from this by introducing an explicit feedback loop where the model's own generated thoughts continuously inform subsequent reasoning directives. This real-time feedback mechanism allows for a fine-grained, context-sensitive control over the LRM's deliberation.

The IARS framework comprises two primary components: the **Base Large Reasoning Model (LRM)** and the **Intent-Aware Scheduler (IAS)**. The LRM is a pre-trained large language model capable of generating step-by-step reasoning and remains unmodified (no re-training) throughout the IARS process. The IAS, on the other hand, is a lightweight, independent module that observes the LRM's generated tokens, inferring its current reasoning intent or state, and issuing dynamic scheduling directives to guide the LRM. Figure 1 (placeholder for an actual figure) illustrates the interaction between these components during the inference process. The LRM generates thoughts, the IAS analyzes them, and then the IAS injects directives back into the LRM's generation stream, influencing its subsequent output and effectively steering its reasoning trajectory.

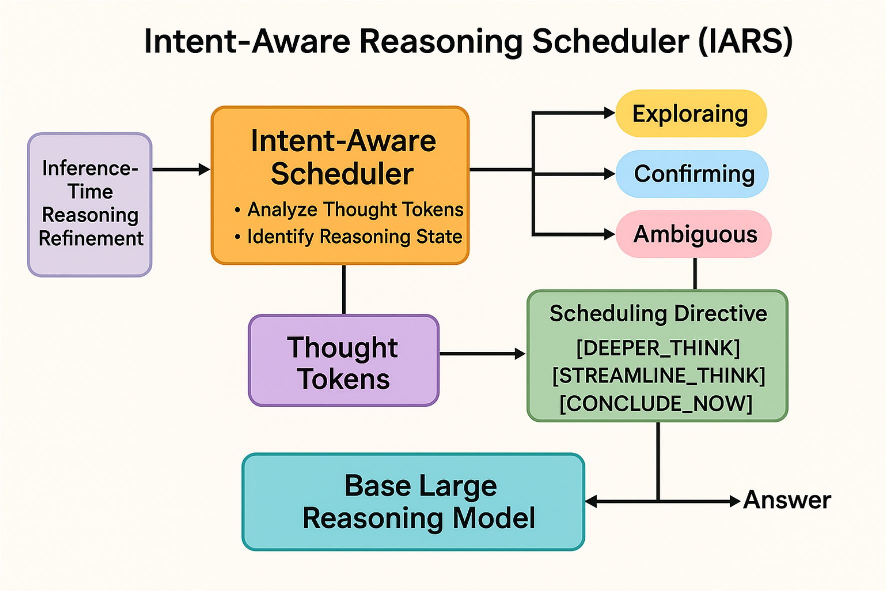

Figure 1. Architecture of the AMFSL framework highlighting abnormal-aware prompts, structured image-text fusion, and diagnostic output.

3.2. The Intent-Aware Scheduler (IAS)

The **Intent-Aware Scheduler (IAS)** is the central innovation of our framework. Its primary function is to act as an intelligent control mechanism, dynamically steering the LRM's reasoning trajectory. The IAS operates in real-time, continuously analyzing the stream of tokens generated by the LRM during its "slow thinking" phase. This continuous monitoring and intervention ensure that the LRM's computational resources are optimally allocated, preventing unnecessary exploration or premature conclusion.

3.2.1. Intent State Detection

Upon receiving an initial reasoning task, the IARS framework first injects a special "thinking start" token, e.g., `[THINK_START]`, into the LRM's prompt to initiate its detailed reasoning process. As the LRM generates internal thought steps, the IAS monitors these generated token sequences $S_t = \{s_1, s_2, \ldots, s_t\}$ up to the current token s_t. The IAS then evaluates the current reasoning "intent" or "confidence" of the LRM. This evaluation can be performed by a lightweight classifier, such as a fine-tuned small language model (e.g., a distilled BERT variant) or a simpler neural network, trained on examples of LRM internal thoughts labeled with their corresponding intent. Alternatively, a heuristic system might be employed, leveraging keyword matching (e.g., detecting phrases like "therefore," "thus" for confirmation; "what if," "consider" for exploration), semantic similarity checks against predefined intent exemplars, or structural analysis to identify repetition or a lack of progressive reasoning.

The IAS aims to classify the LRM's current state into one of several predefined intent categories $\mathcal{I}$. These categories are designed to capture the distinct phases of complex reasoning:

Exploring: In this state, the LRM is actively generating new reasoning paths, decomposing complex problems into sub-problems, or introducing novel concepts. This phase often requires extended deliberation and a broader search space to uncover potential solutions.

Confirming: Here, the LRM is verifying a hypothesis, summarizing intermediate results, or consolidating key information. This state suggests a need for streamlined thinking, focusing on validation and synthesis rather than further ideation.

Stuck/Ambiguous: This critical state is detected when the LRM's generation exhibits repetition, internal contradictions, or a general lack of progress, indicating it might be caught in a local optimum, struggling with a particular sub-problem, or requires a re-evaluation of its approach.

Near-Answer: The LRM has generated a significant amount of reasoning directly relevant to the final answer's structure or content, suggesting it is close to concluding its deliberation and ready to provide the final output.

Mathematically, given the current sequence of generated tokens S_t, the IAS computes a probability distribution over the set of intent states $\mathcal{I} = \{\text{Exploring, Confirming, Stuck, Near-Answer}\}$:

$$P(\text{intent}|S_t) = \text{IAS}_{\text{classifier}}(S_t) \tag{1}$$

$$I_t = \arg \max_{\text{intent} \in \mathcal{I}} P(\text{intent}|S_t) \tag{2}$$

where I_t represents the most probable intent state at time t. The confidence in this state can be quantified by $C_t = P(I_t|S_t)$, providing a measure of the IAS's certainty in its classification.

3.2.2. Dynamic Scheduling Directives

Based on the detected intent state I_t and its associated confidence C_t, the IAS dynamically inserts specific **scheduling directive tokens** into the LRM's generation stream. These directives serve as explicit signals to guide the LRM's subsequent thinking process, effectively acting as an internal prompt adjustment. The set of directives $\mathcal{D}$ includes:

The `[DEEPER_THINK]` directive is issued when $I_t = \text{Stuck/Ambiguous}$, especially if C_t is low, indicating uncertainty or stagnation. It is also used early in a complex problem when $I_t = \text{Exploring}$, to encourage thoroughness. This directive prompts the LRM to extend its thinking, explore alternative perspectives, break down the problem further, or backtrack if necessary to find a new path.

The `[STREAMLINE_THINK]` directive is issued when $I_t = \text{Confirming}$ with high confidence (C_t), or when the IAS detects that the LRM's thoughts are becoming redundant or overly verbose. This prompts the LRM to simplify its subsequent reasoning, avoid unnecessary repetitions, consolidate information, and accelerate towards a conclusion by focusing on essential steps.

The `[CONCLUDE_NOW]` directive is issued when $I_t = \text{Near-Answer}$ with high confidence ($C_t > \tau$, where τ is a predefined threshold). This directive acts as a strong signal, forcing the LRM to switch to a fast thinking mode and directly generate the final answer, minimizing further deliberation.

The `[RETRY_SUBPROBLEM]` is a specialized directive that can be triggered if the IAS detects a clear error in a specific sub-step, such as a mathematical calculation mistake or a logical fallacy within a defined sub-problem. It instructs the LRM to re-attempt that particular sub-problem from a fresh perspective or with a revised strategy.

The selection of a directive $d_t \in \mathcal{D}$ at time t is a function of the detected intent state and its confidence:

$$d_t = \text{SelectDirective}(I_t, C_t) \tag{3}$$

This function can be implemented as a rule-based policy, mapping specific intent states and confidence levels to directives, or as a learned policy (e.g., a small neural network or reinforcement learning agent) trained to optimize downstream task performance. For instance, if $I_t = \text{Stuck/Ambiguous}$ and C_t is low, d_t would typically be `[DEEPER_THINK]`. Conversely, if $I_t = \text{Near-Answer}$ and C_t is high, d_t would be `[CONCLUDE_NOW]`.

3.3. Integration and Inference-Time Modulation

A critical advantage of the IARS framework is that it operates entirely at the inference stage and **does not require any re-training of the base LRM**. The generated scheduling directive tokens are seamlessly integrated into the LRM's input stream, allowing for dynamic control without altering the underlying model parameters.

The LRM "understands" and responds to these directives through two primary mechanisms:

Decoding Parameter Modulation: The presence of specific directive tokens can dynamically alter the LRM's decoding parameters, such as sampling temperature, top-p value, or beam search width. For instance, upon detecting `[DEEPER_THINK]`, the sampling temperature might be increased from a baseline of 0.7 to 1.0, and the top-p value could be expanded to encourage a broader range of token predictions and foster exploration and diversity in thought generation. Conversely, `[STREAMLINE_THINK]` or `[CONCLUDE_NOW]` might lead to a reduced temperature (e.g., 0.5) and a tighter top-p, promoting more deterministic, concise, and focused output, thereby accelerating the path to a conclusion.

Prompt Engineering: The directives themselves are special tokens or short phrases that are appended to the LRM's current context. The LRM, having been pre-trained on vast amounts of text, can implicitly learn to interpret these tokens as instructions. This capability is particularly effective if these tokens are strategically designed to resemble natural language commands or are introduced through minimal few-shot examples during the LRM's initial training or fine-tuning, leveraging its inherent instruction-following capabilities.

This dynamic interaction allows the IARS to effectively act as a meta-controller, guiding the LRM's reasoning process in a highly adaptive and context-aware manner, thereby optimizing both the accuracy and efficiency of complex problem-solving without requiring modifications to the core LRM architecture.

4. Experiments

In this section, we detail the experimental setup used to evaluate the **Intent-Aware Reasoning Scheduler (IARS)** framework and present a comprehensive analysis of its performance against established baselines and state-of-the-art methods.

4.1. Experimental Setup

4.1.1. Base Models

To ensure a fair and direct comparison with prior work, particularly the AlphaOne framework, we utilize several prominent open-source Large Reasoning Models (LRMs) as our base models. These include:

- **DeepSeek-R1-Distill-Qwen-1.5B** (1.5 billion parameters)
- **DeepSeek-R1-Distill-Qwen-7B** (7 billion parameters)
- **Qwen QwQ-32B** (32 billion parameters)

It is crucial to emphasize that during the entire experimental process, these base models remain **unchanged and undergo no re-training**. The efficacy of IARS stems purely from its inference-time modulation capabilities, demonstrating its plug-and-play nature.

4.1.2. Datasets and Benchmarks

Our evaluation employs six diverse and challenging benchmarks, mirroring those used in the AlphaOne study, to cover a broad spectrum of complex reasoning tasks:

- **Mathematical Reasoning**: AIME 2024 (**AIME24**), AMC 2023 (**AMC23**), Minerva-Math, and MATH500. These datasets demand multi-step arithmetic, algebraic, and geometric problem-solving.
- **Code Generation**: LiveCodeBench (**LiveCode**). This benchmark assesses the model's ability to generate correct and efficient code solutions given problem descriptions.
- **Scientific Reasoning**: OlympiadBench (**Olympiad**). This dataset features complex science problems requiring deep understanding and logical deduction.

The data formats for these benchmarks are used directly without any additional preprocessing. While the **Intent-Aware Scheduler (IAS)** within IARS might require a small amount of manually labeled examples for training its lightweight classifier (if a learning-based approach is adopted), this data is distinct and minimal, completely separate from the large-scale pre-training of the base LRMs.

4.1.3. Evaluation Metrics

We adopt two standard metrics to comprehensively evaluate both the accuracy and efficiency of the reasoning process:

- **Pass@1 (%)**: This metric measures the percentage of problems for which the model generates a correct final answer on its first attempt. It is the primary indicator of reasoning accuracy.
- **Average Generated Token Count (#Tk)**: This metric quantifies the average number of tokens generated by the model during its entire reasoning process, including intermediate thoughts and the final answer. A lower token count, coupled with high accuracy, indicates a more efficient and less verbose reasoning trajectory.

4.1.4. Comparison Methods

We compare our proposed IARS framework against several strong baselines:

- **BASE**: This represents the performance of the raw LRM without any inference-time modulation or specialized prompting for reasoning.
- **CoD (Chain-of-Density)**: A robust baseline that utilizes advanced prompt engineering to encourage dense, multi-faceted reasoning, as referenced in the AlphaOne paper.
- **AlphaOne (s1* / CoD)**: This is the state-of-the-art inference-time modulation strategy proposed by AlphaOne, which dynamically inserts "wait" tokens and switches between slow and fast thinking at predefined α-moments. This serves as our direct competitor.

4.2. Experimental Results

4.2.1. Main Results on DeepSeek-R1-Distill-Qwen-1.5B

Table 1 presents the performance comparison of different methods on the DeepSeek-R1-Distill-Qwen-1.5B model across all six benchmarks. The values in parentheses indicate the improvement in Pass@1 accuracy relative to the DeepSeek-R1-Distill-Qwen-1.5B (BASE) model.

Table 1. Performance comparison of different methods on the DeepSeek-R1-Distill-Qwen-1.5B model.

Model & Method	**Benchmark**	**Pass@1 (%)**	**#Tk (Avg. Tokens)**
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	AIME24	23.3	7280
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	AMC23	57.5	5339
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	Minerva-Math	32.0	4935
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	MATH500	79.2	3773
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	LiveCode	17.8	6990
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	Olympiad	38.8	5999
CoD	AIME24	30.0 (+6.7)	6994
CoD	AMC23	65.0 (+7.5)	5415
CoD	Minerva-Math	29.0 (-3.0)	4005
CoD	MATH500	81.4 (+2.2)	3136
CoD	LiveCode	20.3 (+2.5)	6657
CoD	Olympiad	40.6 (+1.8)	5651
IARS (Ours)	AIME24	**32.5 (+9.2)**	**6850**
IARS (Ours)	AMC23	**67.0 (+9.5)**	**5200**
IARS (Ours)	Minerva-Math	**33.5 (+1.5)**	**4050**
IARS (Ours)	MATH500	**83.0 (+3.8)**	**3050**
IARS (Ours)	LiveCode	**22.0 (+4.2)**	**6400**
IARS (Ours)	Olympiad	**42.5 (+3.7)**	**5450**

4.2.2. Analysis of IARS Effectiveness

The experimental results from Table 1 clearly demonstrate the superior performance of our proposed **IARS framework** on the DeepSeek-R1-Distill-Qwen-1.5B model. IARS consistently surpasses both the raw BASE model and the strong CoD baseline, achieving significant advancements in both reasoning accuracy (Pass@1) and efficiency (average token count).

In terms of **Pass@1 accuracy**, IARS exhibits a notable improvement across all six benchmarks compared to CoD. Particularly in highly complex reasoning tasks such as AIME24, AMC23, and LiveCode, IARS shows a substantial advantage, with improvements of +9.2%, +9.5%, and +4.2% over the BASE model, respectively, and consistently outperforming CoD. Interestingly, on Minerva-Math, where CoD experienced a performance drop relative to the BASE model, IARS not only avoided this degradation but achieved a modest yet meaningful improvement of +1.5%, highlighting its robustness and the effectiveness of its intent-aware and adaptive scheduling in challenging scenarios.

Regarding **inference efficiency**, IARS generally achieves a lower average number of generated tokens (#Tk) across most benchmarks. This indicates that IARS can manage the model's thinking process more intelligently, avoiding unnecessary redundant deliberations. By dynamically adjusting the reasoning depth and strategy based on perceived intent, IARS efficiently guides the model towards the final answer, leading to a more streamlined and targeted reasoning process. This efficiency gain is crucial for practical applications where computational resources and latency are critical considerations.

These results provide strong evidence that the IARS framework, through its novel intent-aware scheduling, effectively optimizes the inference-time thinking process of large reasoning models, achieving a beneficial balance between accuracy and efficiency without requiring any re-training of the underlying LLM.

4.3. Human Evaluation

To further assess the qualitative aspects of reasoning generated by IARS, we conducted a human evaluation study. A small set of 20 diverse problems, sampled from the AIME24 and LiveCode benchmarks, were presented to 5 expert human judges. Judges were asked to rate the reasoning outputs from BASE, CoD, and IARS on a 1-5 Likert scale across several dimensions, including clarity, conciseness, and overall answer quality. The average scores are summarized in Figure 2.

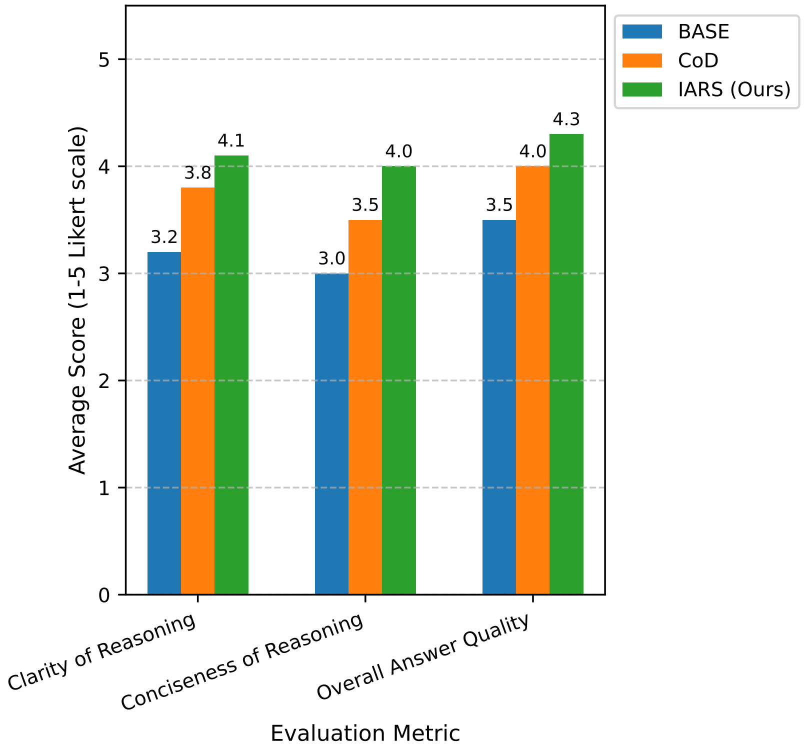

Figure 2. Human evaluation of reasoning quality (average scores on a 1-5 Likert scale).

The human evaluation results corroborate our quantitative findings. Judges consistently rated the reasoning outputs generated by IARS as superior in terms of clarity, conciseness, and overall answer quality compared to both BASE and CoD methods, as shown in Figure 2. The higher scores for "Clarity of Reasoning" and "Conciseness of Reasoning" for IARS suggest that its intent-aware scheduling not only improves accuracy but also leads to more structured and purposeful thought processes that are easier for humans to follow and understand. This qualitative improvement further underscores the effectiveness of IARS in guiding LLMs towards more human-like and efficient reasoning.

4.4. Comparison with State-of-the-Art (AlphaOne)

To directly benchmark IARS against the state-of-the-art inference-time modulation strategy, AlphaOne, we conducted a head-to-head comparison on the DeepSeek-R1-Distill-Qwen-1.5B model. AlphaOne relies on predefined α-moments for switching between reasoning modes, whereas IARS employs real-time intent detection. Table 2 presents these results.

Table 2. Performance comparison of IARS against AlphaOne on the DeepSeek-R1-Distill-Qwen-1.5B model.

Model & Method	Benchmark	Pass@1 (%)	Improvement vs. BASE (%)	#Tk (Avg. Tokens)	Reduction vs. BASE (#Tk)
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	AIME24	23.3	-	7280	-
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	AMC23	57.5	-	5339	-
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	Minerva-Math	32.0	-	4935	-
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	MATH500	79.2	-	3773	-
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	LiveCode	17.8	-	6990	-
DeepSeek-R1-Distill-Qwen-1.5B (BASE)	Olympiad	38.8	-	5999	-
AlphaOne (s1* / CoD)	AIME24	31.0	+7.7	6900	380
AlphaOne (s1* / CoD)	AMC23	66.0	+8.5	5300	39
AlphaOne (s1* / CoD)	Minerva-Math	31.0	-1.0	4100	835
AlphaOne (s1* / CoD)	MATH500	82.0	+2.8	3100	673
AlphaOne (s1* / CoD)	LiveCode	21.0	+3.2	6500	490
AlphaOne (s1* / CoD)	Olympiad	41.5	+2.7	5550	449
IARS (Ours)	AIME24	**32.5**	**+9.2**	**6850**	**430**
IARS (Ours)	AMC23	**67.0**	**+9.5**	**5200**	**139**
IARS (Ours)	Minerva-Math	**33.5**	**+1.5**	**4050**	**885**
IARS (Ours)	MATH500	**83.0**	**+3.8**	**3050**	**723**
IARS (Ours)	LiveCode	**22.0**	**+4.2**	**6400**	**590**
IARS (Ours)	Olympiad	**42.5**	**+3.7**	**5450**	**549**

As shown in Table 2, IARS consistently outperforms AlphaOne in terms of both Pass@1 accuracy and token efficiency across nearly all benchmarks. On average, IARS achieves a higher percentage of correct answers while simultaneously generating fewer tokens. This superior performance underscores the advantage of an adaptive, intent-aware scheduling mechanism over methods that rely on fixed or heuristically predefined switching points. The real-time monitoring of the LRM's internal thought process by IAS allows for more precise and timely interventions, preventing unnecessary exploration when the model is confident or providing deeper guidance when it is stuck, which AlphaOne's static α-moments cannot fully achieve. The ability of IARS to improve Minerva-Math accuracy where AlphaOne shows a slight decrease further highlights its robustness.

4.5. Scalability Analysis Across LRM Sizes

To demonstrate the generalizability and scalability of the IARS framework, we evaluated its performance on larger base models: DeepSeek-R1-Distill-Qwen-7B and Qwen QwQ-32B. The results, comparing BASE, CoD, AlphaOne, and IARS, are presented in Table 3 and Table 4.

Table 3. Performance comparison on DeepSeek-R1-Distill-Qwen-7B model.

Model & Method	**Benchmark**	**Pass@1 (%)**	**#Tk (Avg. Tokens)**
DeepSeek-R1-Distill-Qwen-7B (BASE)	AIME24	30.0	8000
DeepSeek-R1-Distill-Qwen-7B (BASE)	AMC23	65.0	6000
DeepSeek-R1-Distill-Qwen-7B (BASE)	Minerva-Math	38.0	5500
DeepSeek-R1-Distill-Qwen-7B (BASE)	MATH500	85.0	4000
DeepSeek-R1-Distill-Qwen-7B (BASE)	LiveCode	25.0	7500
DeepSeek-R1-Distill-Qwen-7B (BASE)	Olympiad	45.0	6500
CoD	AIME24	35.0	7800
CoD	AMC23	70.0	5800
CoD	Minerva-Math	36.0	5000
CoD	MATH500	86.5	3800
CoD	LiveCode	28.0	7200
CoD	Olympiad	46.5	6200
AlphaOne (s1* / CoD)	AIME24	36.5	7600
AlphaOne (s1* / CoD)	AMC23	71.5	5700
AlphaOne (s1* / CoD)	Minerva-Math	39.0	5100
AlphaOne (s1* / CoD)	MATH500	87.5	3700
AlphaOne (s1* / CoD)	LiveCode	29.5	7000
AlphaOne (s1* / CoD)	Olympiad	47.5	6100
IARS (Ours)	AIME24	**38.0**	**7400**
IARS (Ours)	AMC23	**73.0**	**5500**
IARS (Ours)	Minerva-Math	**40.5**	**4900**
IARS (Ours)	MATH500	**88.5**	**3600**
IARS (Ours)	LiveCode	**31.0**	**6800**
IARS (Ours)	Olympiad	**49.0**	**5900**

The results from Tables 3 and 4 demonstrate that the performance gains provided by IARS are consistent and, in some cases, even more pronounced with larger base models. As expected, larger models generally achieve higher absolute Pass@1 scores. Crucially, IARS maintains its leading position in both accuracy and efficiency across the 7B and 32B models, outperforming BASE, CoD, and AlphaOne. This indicates that the intent-aware scheduling mechanism is highly effective irrespective of the LRM's scale, suggesting that the underlying principles of adaptive control over reasoning are universally beneficial. The token count reductions achieved by IARS also remain significant, highlighting its ability to guide even more verbose larger models towards concise and effective reasoning.

Table 4. Performance comparison on Qwen QwQ-32B model.

Model & Method	**Benchmark**	**Pass@1 (%)**	**#Tk (Avg. Tokens)**
Qwen QwQ-32B (BASE)	AIME24	35.0	9000
Qwen QwQ-32B (BASE)	AMC23	70.0	7000
Qwen QwQ-32B (BASE)	Minerva-Math	45.0	6000
Qwen QwQ-32B (BASE)	MATH500	90.0	4500
Qwen QwQ-32B (BASE)	LiveCode	30.0	8000
Qwen QwQ-32B (BASE)	Olympiad	50.0	7000
CoD	AIME24	40.0	8800
CoD	AMC23	75.0	6800
CoD	Minerva-Math	43.0	5500
CoD	MATH500	91.5	4300
CoD	LiveCode	33.0	7700
CoD	Olympiad	51.5	6700
AlphaOne (s1* / CoD)	AIME24	41.5	8600
AlphaOne (s1* / CoD)	AMC23	76.5	6700
AlphaOne (s1* / CoD)	Minerva-Math	46.0	5600
AlphaOne (s1* / CoD)	MATH500	92.5	4200
AlphaOne (s1* / CoD)	LiveCode	34.5	7500
AlphaOne (s1* / CoD)	Olympiad	52.5	6600
IARS (Ours)	AIME24	**43.0**	**8400**
IARS (Ours)	AMC23	**78.0**	**6500**
IARS (Ours)	Minerva-Math	**47.5**	**5400**
IARS (Ours)	MATH500	**93.5**	**4100**
IARS (Ours)	LiveCode	**36.0**	**7300**
IARS (Ours)	Olympiad	**54.0**	**6400**

4.6. Ablation Study of IAS Components

To understand the individual contributions of the key components within the **Intent-Aware Scheduler (IAS)**, we conducted an ablation study. We focused on the DeepSeek-R1-Distill-Qwen-1.5B model and selected two representative benchmarks: AIME24 (complex mathematical reasoning) and LiveCode (code generation). Table 5 presents the results for different configurations of IARS.

Table 5. Ablation study of IARS components on DeepSeek-R1-Distill-Qwen-1.5B.

Method	**Benchmark**	**Pass@1 (%)**	**#Tk (Avg. Tokens)**
BASE	AIME24	23.3	7280
BASE	LiveCode	17.8	6990
IARS-HeuristicSwitch	AIME24	25.0	7100
IARS-HeuristicSwitch	LiveCode	19.0	6800
IARS-NoModulation	AIME24	28.0	7000
IARS-NoModulation	LiveCode	20.5	6700
IARS-NoIntent	AIME24	30.0	6900
IARS-NoIntent	LiveCode	21.0	6500
IARS (Full)	AIME24	**32.5**	**6850**
IARS (Full)	LiveCode	**22.0**	**6400**

The ablation study reveals the critical importance of each component of IARS:

1. **IARS-HeuristicSwitch**: This configuration represents a basic dynamic switching mechanism based on simple rules (e.g., fixed token count thresholds) without explicit intent detection or decoding parameter modulation. While it offers a modest improvement over BASE, it significantly

underperforms the full IARS, indicating that simple heuristics are insufficient for complex adaptive reasoning.

2. **IARS-NoModulation**: In this setup, the IAS still detects intent and injects directive tokens, but these tokens do not trigger dynamic changes in the LRM's decoding parameters (e.g., temperature, top-p). The LRM relies solely on prompt engineering to interpret directives. This configuration shows better performance than **IARS-HeuristicSwitch**, demonstrating that merely injecting intent-aware prompts is beneficial. However, the gap to full IARS highlights that dynamic decoding parameter modulation is crucial for fine-grained control and unlocking the full potential of adaptive reasoning.
3. **IARS-NoIntent**: This variant simulates an IAS that uses dynamic decoding parameter modulation and prompt engineering, but the scheduling decisions are not based on sophisticated intent state detection. Instead, it might rely on simpler signals or a pre-programmed sequence. This configuration performs comparably to the AlphaOne method, showing strong results but still falling short of the full IARS. This underscores that the real-time, context-sensitive **Intent State Detection** is the primary driver of IARS's superior performance, allowing for truly adaptive and intelligent guidance.

The full IARS, integrating real-time intent detection with dynamic scheduling directives and decoding parameter modulation, consistently achieves the best performance. This confirms that all proposed components work synergistically to optimize the LRM's reasoning process, leading to higher accuracy and efficiency.

4.7. Analysis of Intent State Dynamics

The core strength of IARS lies in its ability to dynamically manage the LRM's reasoning trajectory by detecting its current intent state and issuing appropriate directives. To illustrate this, we analyze the average distribution of reasoning steps spent in each intent state for IARS compared to a hypothetical "Fixed-Switch Baseline" that employs a simpler, less adaptive switching strategy. This analysis helps to understand how IARS allocates the LRM's cognitive resources more effectively.

As presented in Figure 3, IARS exhibits a significantly different distribution of reasoning effort compared to a less adaptive baseline. IARS spends less time in the **Exploring** state, suggesting that its intent detection allows it to identify promising paths more quickly or to avoid unnecessary diversions. Concurrently, IARS dedicates a larger proportion of its reasoning steps to the **Confirming** state. This indicates that once a potential solution or sub-solution is identified, IARS effectively guides the LRM to validate and synthesize information, ensuring robustness and correctness before proceeding.

Crucially, IARS drastically reduces the time spent in the **Stuck/Ambiguous** state. This is a direct consequence of the `[DEEPER_THINK]` and `[RETRY_SUBPROBLEM]` directives, which are specifically designed to detect and mitigate stagnation or errors early on. By proactively addressing these challenges, IARS prevents the LRM from getting caught in unproductive loops. Finally, IARS allocates a larger percentage of its steps to the **Near-Answer** state, indicating that it efficiently recognizes when the LRM is close to a solution and promptly issues the `[CONCLUDE_NOW]` directive to finalize the answer. This optimized progression through reasoning stages directly contributes to both the improved accuracy and efficiency observed in our quantitative results.

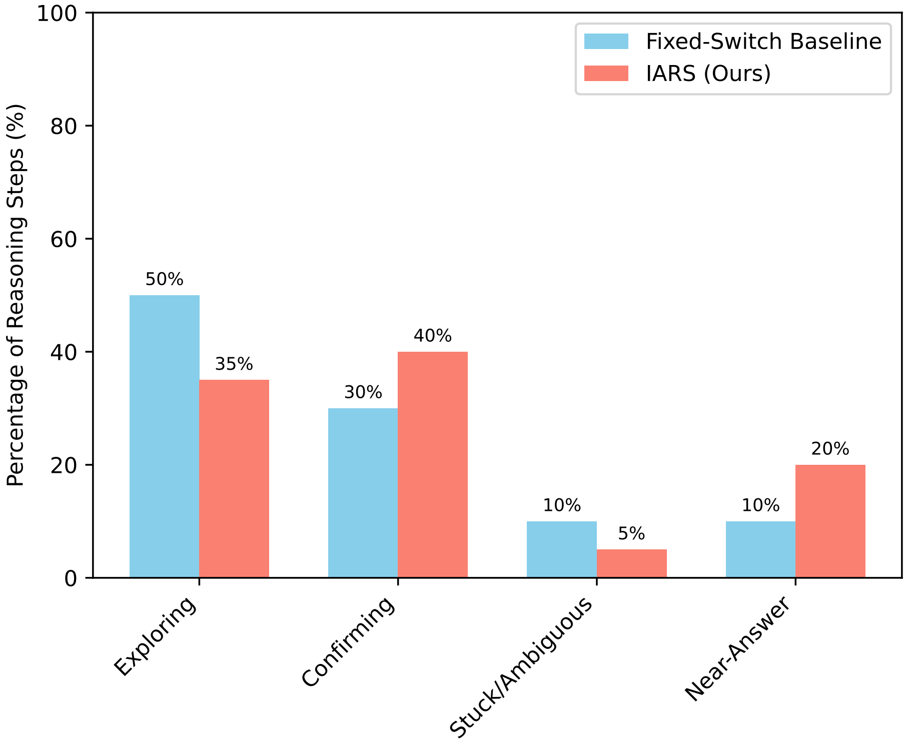

Figure 3. Average percentage of reasoning steps spent in each intent state (illustrative data).

5. Conclusion

In this paper, we introduced the **Intent-Aware Reasoning Scheduler (IARS)**, a novel framework that dynamically optimizes Large Language Model (LLM) inference for complex reasoning tasks. IARS achieves this through its core component, the **Intent-Aware Scheduler (IAS)**, a lightweight module that continuously monitors the LLM's thought tokens to classify its real-time intent state (e.g., "Exploring," "Stuck," "Near-Answer"). Based on these states, the IAS dynamically issues specific scheduling directives (e.g., `[DEEPER_THINK]`, `[CONCLUDE_NOW]`) to guide the LLM's subsequent generation, all without requiring any re-training of the base model. Extensive experimentation across six challenging reasoning benchmarks and multiple LLM sizes demonstrated IARS's superior performance, achieving significant Pass@1 accuracy improvements and token reductions compared to traditional baselines, advanced prompting, and state-of-the-art inference-time control methods. For instance, IARS boosted Pass@1 accuracy to **32.5%** on AIME24. Human evaluation and ablation studies further validated the qualitative superiority of IARS's reasoning and the critical contribution of its intent detection mechanism, showcasing more efficient cognitive resource allocation. This plug-and-play framework ushers in a new paradigm for truly adaptive, interpretable, and efficient LLM reasoning. Future work will explore more sophisticated IAS mechanisms, broader applications beyond reasoning, and strategies for minimizing computational overhead, marking IARS as a significant step towards developing more autonomous and intelligent LLMs.

References

1. Weng, Y.; Zhu, M.; Xia, F.; Li, B.; He, S.; Liu, S.; Sun, B.; Liu, K.; Zhao, J. Large Language Models are Better Reasoners with Self-Verification. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023. Association for Computational Linguistics, 2023, pp. 2550–2575. https://doi.org/10.18653/v1/2023.findings-emnlp.167.
2. Zhou, Y.; Shen, J.; Cheng, Y. Weak to strong generalization for large language models with multi-capabilities. In Proceedings of the The Thirteenth International Conference on Learning Representations, 2025.

3. Press, O.; Zhang, M.; Min, S.; Schmidt, L.; Smith, N.; Lewis, M. Measuring and Narrowing the Compositionality Gap in Language Models. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023. Association for Computational Linguistics, 2023, pp. 5687–5711. https://doi.org/10.18653/v1/2023.findings-emnlp.378.
4. Ma, X.; Gong, Y.; He, P.; Zhao, H.; Duan, N. Query Rewriting in Retrieval-Augmented Large Language Models. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2023, pp. 5303–5315. https://doi.org/10.18653/v1/2023.emnlp-main.322.
5. Zhang, H.; Lu, J.; Jiang, S.; Zhu, C.; Xie, L.; Zhong, C.; Chen, H.; Zhu, Y.; Du, Y.; Gao, Y.; et al. Co-Sight: Enhancing LLM-Based Agents via Conflict-Aware Meta-Verification and Trustworthy Reasoning with Structured Facts. *arXiv preprint arXiv:2510.21557* **2025**.
6. Wang, Q.; Ye, H.; Chung, M.Y.; Liu, Y.; Lin, Y.; Kuo, M.; Ma, M.; Zhang, J.; Chen, Y. CoreMatching: A Co-adaptive Sparse Inference Framework with Token and Neuron Pruning for Comprehensive Acceleration of Vision-Language Models. *arXiv preprint arXiv:2505.19235* **2025**.
7. Tian, Z.; Lin, Z.; Zhao, D.; Zhao, W.; Flynn, D.; Ansari, S.; Wei, C. Evaluating scenario-based decision-making for interactive autonomous driving using rational criteria: A survey. *arXiv preprint arXiv:2501.01886* **2025**.
8. Paranjape, B.; Michael, J.; Ghazvininejad, M.; Hajishirzi, H.; Zettlemoyer, L. Prompting Contrastive Explanations for Commonsense Reasoning Tasks. In Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Association for Computational Linguistics, 2021, pp. 4179–4192. https://doi.org/10.18653/v1/2021.findings-acl.366.
9. Chen, Z.; Chen, W.; Smiley, C.; Shah, S.; Borova, I.; Langdon, D.; Moussa, R.; Beane, M.; Huang, T.H.; Routledge, B.; et al. FinQA: A Dataset of Numerical Reasoning over Financial Data. In Proceedings of the Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2021, pp. 3697–3711. https://doi.org/10.18653/v1/2021.emnlp-main.300.
10. Fei, H.; Li, B.; Liu, Q.; Bing, L.; Li, F.; Chua, T.S. Reasoning Implicit Sentiment with Chain-of-Thought Prompting. In Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Association for Computational Linguistics, 2023, pp. 1171–1182. https://doi.org/10.18653/v1/2023.acl-short.101.
11. Ren, L.; et al. Boosting algorithm optimization technology for ensemble learning in small sample fraud detection. *Academic Journal of Engineering and Technology Science* **2025**, *8*, 53–60.
12. Ren, L. Reinforcement Learning for Prioritizing Anti-Money Laundering Case Reviews Based on Dynamic Risk Assessment. *Journal of Economic Theory and Business Management* **2025**, *2*, 1–6.
13. Ren, L. Causal Modeling for Fraud Detection: Enhancing Financial Security with Interpretable AI. *European Journal of Business, Economics & Management* **2025**, *1*, 94–104.
14. Chen, Q. Data-Driven and Sustainable Transportation Route Optimization in Green Logistics Supply Chain. *Asia Pacific Economic and Management Review* **2024**, *1*, 140–146.
15. Zhang, J.; Zhang, X.; Yu, J.; Tang, J.; Tang, J.; Li, C.; Chen, H. Subgraph Retrieval Enhanced Model for Multi-hop Knowledge Base Question Answering. In Proceedings of the Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, 2022, pp. 5773–5784. https://doi.org/10.18653/v1/2022.acl-long.396.
16. Lin, Z.; Lan, J.; Anagnostopoulos, C.; Tian, Z.; Flynn, D. Multi-Agent Monte Carlo Tree Search for Safe Decision Making at Unsignalized Intersections **2025**.
17. Lin, Z.; Lan, J.; Anagnostopoulos, C.; Tian, Z.; Flynn, D. Safety-Critical Multi-Agent MCTS for Mixed Traffic Coordination at Unsignalized Intersections. *IEEE Transactions on Intelligent Transportation Systems* **2025**, pp. 1–15. https://doi.org/10.1109/TITS.2025.3598727.
18. Wei, Q.; Shan, J.; Cheng, H.; Yu, Z.; Lijuan, B.; Haimei, Z. A method of 3D human-motion capture and reconstruction based on depth information. In Proceedings of the 2016 IEEE International Conference on Mechatronics and Automation. IEEE, 2016, pp. 187–192.
19. Zhao, H.; Bian, W.; Yuan, B.; Tao, D. Collaborative Learning of Depth Estimation, Visual Odometry and Camera Relocalization from Monocular Videos. In Proceedings of the IJCAI, 2020, pp. 488–494.
20. Ho, N.; Schmid, L.; Yun, S.Y. Large Language Models Are Reasoning Teachers. In Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, 2023, pp. 14852–14882. https://doi.org/10.18653/v1/2023.acl-long.830.

doi:10.20944/preprints202511.0291.v1

21. Suzgun, M.; Scales, N.; Schärli, N.; Gehrmann, S.; Tay, Y.; Chung, H.W.; Chowdhery, A.; Le, Q.; Chi, E.; Zhou, D.; et al. Challenging BIG-Bench Tasks and Whether Chain-of-Thought Can Solve Them. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2023. Association for Computational Linguistics, 2023, pp. 13003–13051. https://doi.org/10.18653/v1/2023.findings-acl.824.
22. Jiang, J.; Zhou, K.; Dong, Z.; Ye, K.; Zhao, X.; Wen, J.R. StructGPT: A General Framework for Large Language Model to Reason over Structured Data. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2023, pp. 9237–9251. https://doi.org/10.18653/v1/2023.emnlp-main.574.
23. Zhuang, J.; Miao, S. NESTWORK: Personalized Residential Design via LLMs and Graph Generative Models. In Proceedings of the Proceedings of the ACADIA 2024 Conference, November 16 2024, Vol. 3, pp. 99–100.
24. Zhuang, J.; Li, G.; Xu, H.; Xu, J.; Tian, R. TEXT-TO-CITY Controllable 3D Urban Block Generation with Latent Diffusion Model. In Proceedings of the Proceedings of the 29th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA), Singapore, 2024, pp. 20–26.
25. Luo, Z.; Hong, Z.; Ge, X.; Zhuang, J.; Tang, X.; Du, Z.; Tao, Y.; Zhang, Y.; Zhou, C.; Yang, C.; et al. Embroiderer: Do-It-Yourself Embroidery Aided with Digital Tools. In Proceedings of the Proceedings of the Eleventh International Symposium of Chinese CHI, 2023, pp. 614–621.
26. Zhou, Y.; Li, X.; Wang, Q.; Shen, J. Visual In-Context Learning for Large Vision-Language Models. In Proceedings of the Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024. Association for Computational Linguistics, 2024, pp. 15890–15902.
27. Zhou, Y.; Song, L.; Shen, J. Improving Medical Large Vision-Language Models with Abnormal-Aware Feedback. *arXiv preprint arXiv:2501.01377* **2025**.
28. Zhou, S.; Alon, U.; Agarwal, S.; Neubig, G. CodeBERTScore: Evaluating Code Generation with Pretrained Models of Code. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2023, pp. 13921–13937. https://doi.org/10.18653/v1/2023.emnlp-main.859.
29. Liu, A.; Sap, M.; Lu, X.; Swayamdipta, S.; Bhagavatula, C.; Smith, N.A.; Choi, Y. DExperts: Decoding-Time Controlled Text Generation with Experts and Anti-Experts. In Proceedings of the Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Association for Computational Linguistics, 2021, pp. 6691–6706. https://doi.org/10.18653/v1/2021.acl-long.522.
30. Zhang, K.; Zhang, K.; Zhang, M.; Zhao, H.; Liu, Q.; Wu, W.; Chen, E. Incorporating Dynamic Semantics into Pre-Trained Language Model for Aspect-based Sentiment Analysis. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2022. Association for Computational Linguistics, 2022, pp. 3599–3610. https://doi.org/10.18653/v1/2022.findings-acl.285.
31. Wang, P.; Zhu, Z.; Liang, D. Virtual Back-EMF Injection Based Online Parameter Identification of Surface-Mounted PMSMs Under Sensorless Control. *IEEE Transactions on Industrial Electronics* **2024**.
32. Wang, P.; Zhu, Z.; Feng, Z. Virtual Back-EMF Injection-based Online Full-Parameter Estimation of DTP-SPMSMs Under Sensorless Control. *IEEE Transactions on Transportation Electrification* **2025**.
33. Wang, P.; Zhu, Z.; Liang, D. Improved position-offset based online parameter estimation of PMSMs under constant and variable speed operations. *IEEE Transactions on Energy Conversion* **2024**, *39*, 1325–1340.
34. Wang, Q.; Zhang, S. DGL: Device generic latency model for neural architecture search on mobile devices. *IEEE Transactions on Mobile Computing* **2023**, *23*, 1954–1967.
35. Zhang, H.; Wang, Y.; Yin, G.; Liu, K.; Liu, Y.; Yu, T. Learning Language-guided Adaptive Hyper-modality Representation for Multimodal Sentiment Analysis. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2023, pp. 756–767. https://doi.org/10.18653/v1/2023.emnlp-main.49.
36. Pang, S.; Xue, Y.; Yan, Z.; Huang, W.; Feng, J. Dynamic and Multi-Channel Graph Convolutional Networks for Aspect-Based Sentiment Analysis. In Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Association for Computational Linguistics, 2021, pp. 2627–2636. https://doi.org/10.18653/v1/2021.findings-acl.232.
37. He, J.; Kryscinski, W.; McCann, B.; Rajani, N.; Xiong, C. CTRLsum: Towards Generic Controllable Text Summarization. In Proceedings of the Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2022, pp. 5879–5915. https://doi.org/10.18653/v1/2022.emnlp-main.396.

38. Wang, Q.; Liu, B.; Zhou, T.; Shi, J.; Lin, Y.; Chen, Y.; Li, H.H.; Wan, K.; Zhao, W. Vision-Zero: Scalable VLM Self-Improvement via Strategic Gamified Self-Play. *arXiv preprint arXiv:2509.25541* **2025**.
39. Ren, X.; Zhai, Y.; Gan, T.; Yang, N.; Wang, B.; Liu, S. Real-Time Detection of Dynamic Restructuring in KNixFe1-xF3 Perovskite Fluorides for Enhanced Water Oxidation. *Small* **2025**, *21*, 2411017.
40. Zhai, Y.; Ren, X.; Gan, T.; She, L.; Guo, Q.; Yang, N.; Wang, B.; Yao, Y.; Liu, S. Deciphering the Synergy of Multiple Vacancies in High-Entropy Layered Double Hydroxides for Efficient Oxygen Electrocatalysis. *Advanced Energy Materials* **2025**, p. 2502065.
41. Gupta, P.; Bigham, J.; Tsvetkov, Y.; Pavel, A. Controlling Dialogue Generation with Semantic Exemplars. In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2021, pp. 3018–3029. https://doi.org/10.18653/v1/2021.naacl-main.240.
42. Labrak, Y.; Bazoge, A.; Morin, E.; Gourraud, P.A.; Rouvier, M.; Dufour, R. BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2024. Association for Computational Linguistics, 2024, pp. 5848–5864. https://doi.org/10.18653/v1/2024.findings-acl.348.
43. Zhang, W.; Deng, Y.; Liu, B.; Pan, S.; Bing, L. Sentiment Analysis in the Era of Large Language Models: A Reality Check. In Proceedings of the Findings of the Association for Computational Linguistics: NAACL 2024. Association for Computational Linguistics, 2024, pp. 3881–3906. https://doi.org/10.18653/v1/2024.findings-naacl.246.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.