WirelessMathBench: A Mathematical Modeling Benchmark for LLMs in Wireless Communications

Anonymous ACL submission

Abstract

Large Language Models (LLMs) have achieved impressive results across a broad array of tasks, yet their capacity for complex, domain-specific mathematical reasoning—particularly in wireless communications—remains underexplored. In this work, we introduce WirelessMathBench a novel benchmark specifically designed to evaluate LLMs on mathematical modeling challenges to wireless communications engineering. Our benchmark consists of 587 meticulously curated questions sourced from 40 state-of-the-art research papers, encompassing a diverse spectrum of tasks ranging from basic multiple-choice questions to complex equation completion tasks, including both partial and full completions, all of which rigorously adhere to physical and dimensional constraints. Through extensive experimentation with leading LLMs, we observe that while many models excel in basic recall tasks, their performance degrades significantly when reconstructing partially or fully obscured equations, exposing fundamental limitations in current LLMs. Even DeepSeek-R1, the best performer on our benchmark, achieves an average accuracy of only 38.05%, with a mere 7.83% success rate in full equation completion. By publicly releasing WirelessMathBench along with the evaluation toolkit, we aim to advance the development of more robust, domain-aware LLMs for wireless system analysis and broader engineering applications.

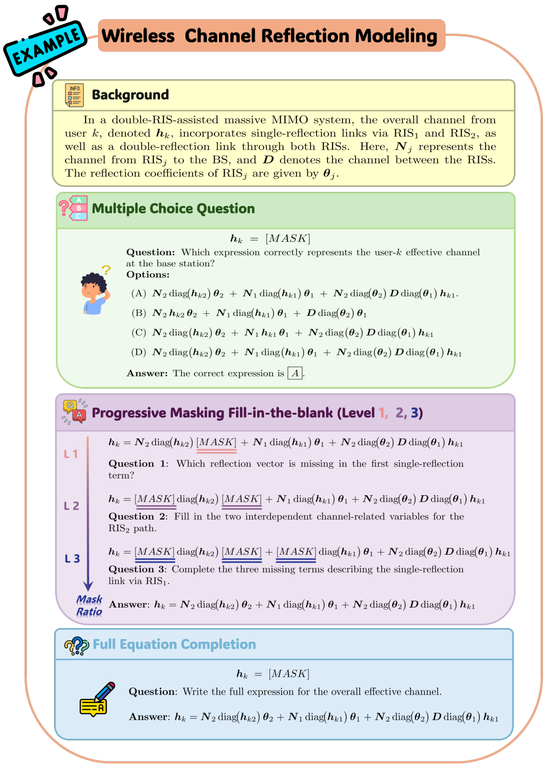

Figure 1: Example task from WirelessMathBench a system model derivation from wireless communications literature. The derivation progresses from a multiple-choice question to progressive mask completion questions, and finally to the full formula derivation, testing the model's ability to reason through complex channel reflections and matrix operations.

1 Introduction

Large Language Models (LLMs) have recently demonstrated groundbreaking performance across a diverse range of natural language tasks—from general language understanding (Brown et al., 2020; Wang et al., 2019b,a) and code generation (Lu et al., 2021) to elementary mathematical reasoning (Cobbe et al., 2021; Hendrycks et al., 2021a). Advanced models such as OpenAI-o1 (OpenAI, 2024) and DeepSeek-R1 (Guo et al., 2025) have further extended these capabilities, especially when supplemented with chain-of-thought strategies that enable clear, step-by-step solution processes. Nevertheless, despite these notable achievements, current state-of-the-art LLMs still encounter significant difficulties when tackling highly intricate problem statements. In particular, tasks that demand deep conceptual insights, rigorous validation of physical feasibility, and the careful management of tightly interrelated parameter sets continue to pose formidable challenges

(Mirzadeh et al., 2025; Zhang et al., 2023; He et al., 2024).

In many engineering fields—wireless communications in particular—mathematical modeling is indispensable. The design and analysis of modern wireless systems require not only accurate numerical computation but also precise symbolic derivations that honor strict physical and dimensional constraints. Tasks such as channel estimation (Yin et al., 2013; Liu et al., 2022; An et al., 2023), beamforming (Chu et al., 2023; Spencer et al., 2004), and multi-antenna system design (Huang et al., 2020; An et al., 2024; Zheng et al., 2024) involve intricate matrix operations, multi-stage derivations, and domain-specific lexicon. Even minor errors in symbolic manipulation can lead to significant performance degradation or non-compliance with industry standards (Bjornson et al., 2013).

Although recent work has leveraged LLMs for technical definition retrieval in wireless communications (Shao et al., 2024; Maatouk et al., 2023; Zou et al., 2024; Maatouk et al., 2024), few studies have directly addressed the challenges associated with multi-step derivations and symbolic manipulation in this specialized domain. This observation raises a broader question: *To what extent are LLMs capable of emulating the mathematical derivations and analytical typically by an engineer or researcher in the field of real wireless communications?*

To bridge this gap, we introduce WirelessMathBench a comprehensive benchmark specifically designed to test LLMs on the real-work wireless engineering orientation mathematical reasoning. WirelessMathBench comprises 587 high-quality questions sourced from 40 state-of-the-art papers, each carefully annotated and validated by domain experts to ensure accuracy. These questions span a variety of system models (e.g., Multiple-Input and Multiple-Output (MIMO), Non-orthogonal multiple access (NOMA), Reconfigurable Intelligent Surfaces (RIS)) and problem settings (e.g., channel estimation, beamforming), encompassing multiple-choice, fill-in-the-blank, and open-ended questions at various levels. Table1 highlights key differences between WirelessMathBench and other math benchmarks, ours is the only dataset of expert difficulty level and contains real-world engineering problems. Figure 1 illustrates how a single math formula escalates from a basic multiple-choice query to a fully masked equation derivation, reflecting the complexity of real-world wireless system analysis.

Our extensive experiments show that while leading LLMs perform well on simpler tasks (e.g., multiple-choice questions with over 75% accuracy), their performance drops dramatically on advanced derivation tasks (progressive masking and complete equations). Even the strongest model we evaluated, DeepSeek-R1 (Guo et al., 2025), only manages a 7.83% success rate in fully masked derivations, underscoring a fundamental gap between current LLM capabilities and the complex demands of wireless systems analysis.

By publicly releasing WirelessMathBench along with its evaluation toolkit, we aim to spur progress toward LLMs that are not only fluent in natural language but also capable of rigorous, domain-specific mathematical reasoning. We envision that WirelessMathBench will serve as a catalyst for innovation in mathematical reasoning capabilities, domain-adaptive pre-training techniques, and advanced thought-chaining strategies, ultimately propelling LLMs toward more robust scientific and engineering problem-solving.

2 Related Work

General-Purpose LLM Benchmarks. In recent years, rapid advancements in LLMs—exemplified by models such as GPT-3 (Brown et al., 2020), GPT-4 (Achiam et al., 2023), LLaMA (Touvron et al., 2023), Gemini (Team et al., 2023), and DeepSeek-R1 (Guo et al., 2025)—have spurred extensive evaluations on benchmarks like GLUE (Wang et al., 2019b), SuperGLUE (Wang et al., 2019a), and GSM8K (Cobbe et al., 2021). However, despite covering a broad spectrum of linguistic tasks, they typically lack the depth and specificity required to evaluate rigorous mathematical modeling or the domain-specific symbolic reasoning needed for complex technical applications.

Mathematical Reasoning Benchmarks. A parallel research stream has focused on the mathematical and symbolic reasoning abilities of LLMs. Early mathematical benchmarks (Amini et al., 2019; Cobbe et al., 2021; Koncel-Kedziorski et al., 2016; Ling et al., 2017; Hendrycks et al., 2021b) evaluate models on elementary arithmetic, algebra, and calculus problems. Recently, as the complexity of the problem increases, some benchmarks introduce competition-level problems that combine mathematical logic and background knowledge (Yu et al., 2024;

Benchmark	Diffuculty Level	QuestionType	Real Engineering Tasks	#Test Size
GSM8K (Cobbe et al., 2021)	Elementary School	OE	No	1,319
MATH (Hendrycks et al., 2021b)	High School	OE	No	5,000
OCWCourses (Lewkowycz et al., 2022)	University	OE	No	272
MMMU (Yue et al., 2024)	University	MC,OE	No	1983
OlympiadBench (He et al., 2024)	Competition	OE	No	8,476
WirelessMathBench	**Expert**	**MC, FB, OE**	**Yes**	**587**

Table 1: Comparison of representative mathematical benchmarks with **WirelessMathBench**. Existing datasets largely focus on elementary, high school, or Olympiad-level problems in purely theoretical contexts, while **WirelessMathBench** targets real-world, expert-level engineering tasks under strict dimensional and physical constraints. We note that open-ended (OE) tasks typically require free-form answers, MC indicates multiple-choice, and FB refers to fill-in-the-blank.

Hendrycks et al., 2021a; Arora et al., 2023; Frieder et al., 2024). For more advanced mathematical reasoning, datasets like MMMU (Yue et al., 2024), OCWCourses (Lewkowycz et al., 2022) and U-MATH (Chernyshev et al., 2024) focuses on university-level mathematics problems. MiniF2F (Zheng et al., 2022), AlphaGeometry (Trinh et al., 2024), OlympiadBench (He et al., 2024), and MathOdyssey (Fang et al., 2024) go further to Olympiad-level problems that require more advanced mathematical reasoning. Yet, these datasets do not capture the unique constraints or specialized notations found in applied domains like wireless communications.

Domain-Specific Benchmarks. To overcome the limitations of general-purpose evaluations, several domain-specific benchmarks have been developed for tasks that demand technical precision and specialized reasoning. For example, customized benchmarks have been developed for legal document analysis (Guha et al., 2024), chemical property inference (Guo et al., 2023), and scientific reasoning (Lu et al., 2022; Wang et al., 2024; Sun et al., 2024). To evaluate LLMs in more specialized domain tasks, recent works have introduced benchmarks like MLAgentBenchmark (Huang et al., 2024), which evaluates LLMs' ability to solve machine learning tasks, AI-Researcher (Si et al., 2025) evaluate can LLMs generate research ideas, and SWE-Bench (Jimenez et al., 2024) evaluate LLMs' ability to solve real-world software engineering tasks. These studies highlight that an in-depth evaluation of LLMs in specialized fields reveals that LLMs have strong potential in different professional fields.

LLMs in Wireless Communications. Wireless communications impose stringent requirements on mathematical precision, particularly for tasks such as channel estimation, interference management, and beamforming (Cadambe and Jafar, 2008; Shi et al., 2011; Gesbert et al., 2010). Some preliminary works have explored the use of LLMs in wireless contexts, focusing on domain-specific knowledge extraction and basic recall of technical standards (Maatouk et al., 2023; Shao et al., 2024; Maatouk et al., 2024). Notably, TelecomGPT (Zou et al., 2024) has extended LLM capabilities to higher-level tasks like wireless-specific code generation and formula completion. However, these early works usually emphasize knowledge retrieval or summarization, without considering testing what tasks LLMs can accomplish in actual wireless communication engineering systems.

In this work, we introduce WirelessMathBench to address these gaps. Unlike existing wireless or purely mathematical benchmarks, WirelessMathBench offers tasks that systematically combine multiple-choice questions with progressively masked formula derivations, all drawn from state-of-the-art research papers. The goal is to evaluate both symbolic reasoning and domain knowledge under realistic conditions, capturing the nuanced interplay of mathematical derivations and physical feasibility inherent in wireless communications. By providing a diverse set of tasks and domain-informed evaluation metrics, WirelessMathBench aims to facilitate collaborative advances in both LLMs and wireless communication engineering, ultimately enabling more powerful AI-assisted solutions for next-generation wireless networks.

3 The WirelessMathBench Benchmark

In this section, we present WirelessMathBench, a new benchmark specifically designed to evaluate LLMs on mathematical modeling tasks within wireless communications. We begin by discussing the rationale behind our benchmark design (Sec-

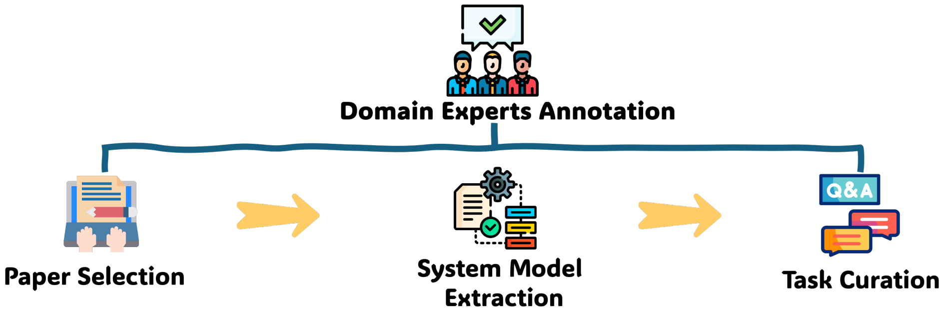

Figure 2: Overview of the data collection and annotation pipeline for WirelessMathBenchṪhe process involves selecting high-quality research papers, extracting system models from papers, curating tasks of varying complexity levels, and reviewing each task for clarity and correctness.

tion 3.1), followed by the details of our data collection and annotation pipeline (Section 3.2). We then explain how we construct questions of varying complexity levels, as well as our progressive masking methodology (Section 3.3).

3.1 Design Principles

The creation of WirelessMathBench is motivated by two core observations. First, recent work shows that LLMs can effectively assist humans in highly specialized tasks (Guha et al., 2024; Guo et al., 2023; Lu et al., 2022), underscoring their potential when provided with sufficient domain context. Second, LLMs have demonstrated the capacity to handle increasingly difficult mathematics, including Olympiad-level challenges (He et al., 2024; Fang et al., 2024). These findings suggest a substantial opportunity to push the limits of LLMs in areas where complex, domain-specific mathematics—such as wireless communications—plays a central role.

Building on these insights, WirelessMathBench is designed around two key principles:

1. **Real-World Complexity.** Each task is sourced directly from peer-reviewed research, reflecting the authentic modeling challenges faced in wireless systems.

2. **Multi-Tiered Progression.** Tasks range from basic multiple-choice questions to fully masked derivations, providing graduated levels of difficulty that capture both foundational knowledge and advanced reasoning.

3.2 Data Collection and Annotation

As illustrated in Figure 2, the data collection and annotation process for WirelessMathBench involves four main steps: paper selection, system model extraction, task curation, and domain expert review.

Paper Selection and Coverage. To capture the authentic complexities of wireless communications, we begin by identifying high-impact papers from top-tier publication venues that are freely accessible on arXiv. Table 2 summarizes the coverage of model-based and problem-based categories; in total, we select 40 papers spanning core techniques (e.g., MIMO, NOMA, RIS) and research focuses (e.g., channel estimation, beamforming). We emphasize works that feature nontrivial mathematical derivations—such as optimization formulations and multi-stage channel modeling—over those limited to empirical or simulation-based heuristics. A summary of the high-frequency keywords across the dataset is shown in Figure 3, highlighting the diverse wireless communication topics covered in the benchmark. We aim to include tasks that reflect the symbolic depth and physical constraints that are indispensable for real-world wireless engineering and state-of-the-art wireless research.

System Model Extraction. Our pipeline starts by applying a specially designed LLM template that systematically scans each research paper, isolating key mathematical expressions and relevant contextual descriptions. This initial extraction step is semi-automated: LLMs produce a structured draft containing a concise overview of the system model, assumptions, and principal formulas. Subsequently, domain experts review and refine these drafts, ensuring that the extracted material is both accurate (i.e., symbolically consistent with the original text) and self-contained (i.e., providing enough background to be understood independently). This hybrid method combines the scalability of automated extraction with meticulous expert checks,

Category Type	Topic Category	Number of Papers
Model-based	RIS	19
	MIMO	12
	UAV	6
	ISAC	6
	Satellite	4
	SIM	3
	NOMA	2
Problem-based	Beamforming	18
	Channel Estimation	12
	Performance Analysis	8
	Trajectory Design	5
	Power Allocation	5
	Resource Management	4
Total		**40**

Table 2: Distribution of the WirelessMathBench benchmark papers according to model-based and problem-based categories, along with their respective topic areas. A total of 40 papers are included, covering key themes in wireless communications. Note that some papers may span multiple topic categories.

Figure 3: A word cloud illustrating the most frequent keywords in the WirelessMathBench benchmark, which reflects the range of wireless communication topics covered.

ensuring the resulting text is accurate, symbolically consistent, and sufficiently self-contained for subsequent tasks.

Task Curation. Once the system models have been extracted, we systematically transform them into precise question-answer pairs via a three-step process:

1. **Identify Core Equations:** We select representative formulas from each paper—such as channel gain expressions, optimization objectives, or multi-hop path-loss derivations—that encapsulate critical wireless engineering challenges.

2. **Construct Questions:** For each core equation, we generate questions at varying levels of difficulty. This includes: (a) multiple-choice questions targeting definitions or partial operations; (b) progressively masked fill-in-the-blank questions that require incremental reasoning; and (c) full equation completion tasks that demand derivation of the entire expression.

3. **Annotate and Review:** Each question is accompanied by contextual notes, and domain experts validate correctness and clarity. When necessary, we refine notation or provide short explanations to ensure the questions are self-contained and can be tackled without external references

Domain Expert Review. Lastly, all questions undergo a multi-round review by senior wireless researchers. They verify notation accuracy and domain applicability (e.g., check for appropriate dimensionality, and coherent modeling assumptions), and remove any ambiguous or misleading content. The remaining problems constitute the final WirelessMathBench dataset: a set of carefully selected tasks that embody typical mathematical derivations in advanced wireless communication engineering and research.

3.3 Task Design and Masking Strategies

To full evaluate the capabilities of LLMs at different levels of difficulty in mathematical modeling of wireless communications, WirelessMathBench incorporates three distinct task types. Each question leverages real-world system equations derived from state-of-the-art research papers, ensuring that the benchmark reflects both conceptual diversity and practical engineering relevance. At the same time, each independent question is accompanied by a brief description of the relevant wireless scenario (e.g., UAV relay or multi-antenna beamforming), providing the necessary domain and scenario background information.

Multiple-Choice Questions (MCQs). These questions require the solver to select the correct mathematical expression from a set of closely related distractors, with each MCQ carefully designed to test the model's ability to recognize and recall key elements of wireless system modeling. For example, a typical MCQ may present several equations for a wireless channel, of which only one formula satisfies both the correct dimensions and the physical constraints of the system under consideration.

Progressively masked fill-in-the-blank questions. In this task, a system model formula is progressively presented in a partially masked form across

Model	Source	Size
OpenAI-o1	(OpenAI, 2024)	unknown
DeepSeek-R1	(Guo et al., 2025)	671B
GPT-Family	(OpenAI, 2024; Achiam et al., 2023)	unknown
DeepSeek-V3	(DeepSeek-AI et al., 2024)	671B
Gemini-Famliy	(Google DeepMind, 2024)	unknown
Qwen2.5-Math	(Yang et al., 2024)	7B, 72B
LLaMA	(Grattafiori et al., 2024)	8B, 70B
LLaMA-3-8B-Tele	(Maatouk et al., 2024)	8B
Mistral-7B	(Jiang et al., 2023)	7B

Table 3: LLMs evaluated on WirelessMathBench.

three different masking levels. Each progressively masked instance is treated as an independent subproblem, requiring the model to infer and reconstruct the missing information at each stage. The masking levels range from isolated single-variable omissions to multi-variable occlusions, with varying degrees of accompanying prompt text to provide contextual guidance.

Full Equation Completion (FEC) question. For the most challenging question, the full equation is entirely hidden. The solver is provided with only a succinct description of the wireless scenario (for example, a base station-relay-user link with specific path loss characteristics) and must derive the complete expression from first principles. This task assesses the model's ability to reconstruct the entire derivation—from fundamental definitions (like channel gain or fading coefficients) to the final expression—while ensuring dimensional accuracy and adherence to domain-specific constraints (such as path loss exponent and transmit power limits). It represents the level of performance expected from a human expert in wireless communications.

In summary, by combining MCQs, progressively masked tasks, and full equation completions, WirelessMathBench offers a comprehensive, fine-grained evaluation of a model's capability to perform both symbolic reasoning and domain-specific derivations in wireless communications.

4 Experiments

We now present a comprehensive evaluation of WirelessMathBench, focusing on how leading LLMs handle wireless communications–specific mathematical modeling. We begin by detailing our experimental setup (Section 4.1), including prompt design and model baselines, then discuss our main results (Section 4.2), and conclude with an error analysis that highlights key challenges (Section 4.3).

4.1 Experiment Setup

Evaluation Workflow. All experiments are conducted in a zero-shot setting using unified prompt templates across different question types for consistent evaluation. For each task in WirelessMathBench, we provide the corresponding prompt to each model and collect the answers it generates. Our evaluation pipeline is now completed in two main ways. For multiple-choice questions, we directly extract the output answers and compare their consistency with the annotation results. For Progressive Masking Filling and Fully maded questions, since polynomials may have a certain number of possible answers, we use the help of LLMs (GPT-4o is selected in our experiment) to complete the evaluation, similar to(Fang et al., 2024; Chernyshev et al., 2024). The overall performance is reported as the average accuracy of all tasks. Detailed prompt examples and scoring criteria are provided in the Appendix B.

Baselines. Table 3 (in the main text) lists the principal models tested. We include leading reasoning models(e.g., DeepSeek-R1, OpenAI-o1), large-scale general-purpose LLMs (e.g., GPT-4, Gemini), and specialized models (e.g., Qwen2.5-Math) to capture a broad range of capabilities. For open-source models like LLaMA, we also explore domain-specific variants trained on a telecom corpus (e.g., LLaMA-3-8B-Tele) to gauge the benefit of targeted adaptation. All hyperparameters follow each model's respective default or recommended settings, and no additional chain-of-thought prompting is provided beyond the standard instructions above.

4.2 Main Results

Table 4 presents the performance of sixteen LLMs across five metrics in WirelessMathBench: (1) Multiple-choice Question (MCQ) accuracy, (2–4) progressive masking fill-in at three difficulty levels (Level 1, Level 2, Level 3), (5) Full Equation Completions (FEC), and the overall average accuracy (Avg. Acc). Our key findings are summarized below:

Reasoning-Oriented Models Show Advantages. Models that incorporate explicit chain-of-thought or advanced reasoning techniques—like DeepSeek-R1 and OpenAI-o1—consistently outperform simpler large-scale baselines. The average accuracy of **DeepSeek-R1** is 38.05%, and the average accuracy

Model	MCQ	Progressive Masking Filling			FEC	Avg. Acc
		Level 1	Level 2	Level 3		
DeepSeek-R1	76.00%	**60.00%**	**34.91%**	**12.50%**	**7.83%**	**38.05%**
OpenAI-o1	66.40%	59.17%	32.17%	8.04%	6.96%	34.55%
OpenAI-o1-mini	66.40%	53.33%	29.57%	10.71%	4.35%	32.87%
GPT-4o	72.80%	42.50%	28.70%	6.25%	4.35%	30.92%
GPT-4	53.60%	38.33%	18.26%	3.57%	4.35%	23.62%
GPT-3.5-turbo	45.60%	7.50%	10.43%	1.79%	1.74%	13.41%
DeepSeek-V3	**78.40%**	50.00%	24.35%	6.25%	6.96%	33.19%
Gemini-2.0-flash	71.20%	40.83%	24.35%	5.36%	4.35%	29.22%
Gemini-1.5-pro	65.60%	43.33%	29.57%	9.82%	6.09%	30.88%
Gemini-1.5-flash	66.40%	37.50%	13.91%	2.68%	4.35%	24.97%
Qwen2.5-Math-72B	70.40%	37.50%	26.09%	7.14%	6.09%	29.44%
LLaMA-3.3-70B	65.60%	38.33%	17.39%	2.68%	6.09%	26.02%
Qwen2.5-Math-7B	58.40%	21.67%	6.96%	4.46%	1.74%	18.82%
LLaMA-3-8B-Tele	40.80%	11.67%	4.35%	2.68%	0.87%	12.07%
LLaMA-3-8B	45.60%	10.83%	7.83%	2.68%	2.61%	13.91%
Mistral 7B	38.40%	20.00%	4.35%	0.89%	0.87%	12.90%

Table 4: Experimental results of state-of-the-art LLMs on WirelessMathBench. The table shows the performance of each model on MCQ, progressively masked filling and full equation completion tasks.

of **OpenAI-o1** is 34.55%, while the accuracy of other large-parameter models hovers around 30%. This suggests that explicit reasoning strategies contribute substantially to managing multi-step symbolic derivations in wireless communications tasks.

Strong MCQ Performance but Rapid Decline in Derivations. Several models, including DeepSeek R1, V3, GPT-4, and Gemini-2.0, exceed 70% accuracy on MCQs, showing that they can find the correct formula given background knowledge and given error options, indicating that they can understand the modeling process and matrix operations in the communication domain to some extent. However, these MCQ gains generally do not extend to more complex derivation tasks, where most models' accuracy falls dramatically. For instance, **DeepSeek-V3** achieves the highest MCQ score at 78.40%, drop to 6.25% in Level 3 masking filling, and 6.96% in FEC.

Progressive Masking Emphasizes Multi-Step Reasoning Gaps. When forced to reconstruct partially hidden expressions, model performance declines in proportion to the level of masking. When forced to reconstruct partially hidden expressions, model performance degrades with increasing levels of masking. Models with implicit reasoning logic significantly outperform the others, with **DeepSeek-R1** in particular leading on these tasks—achieving 60.00% at Level 1 and 33.91% at Level 2, suggesting more robust chaining of thoughts. However, even DeepSeek-R1 struggles at level 3 (12.50%), highlighting the difficulty of maintaining symbolic coherence under heavily ambiguous conditions.

Fully Masked Equation Completion Remains Challenging. Most models attain only single-digit accuracy (2–7%) in the FEC task, where the entire equation is hidden. **DeepSeek-R1**'s 7.83% and **OpenAI-o1**'s 6.96% are the best in this category, but both remain low in absolute terms, indicating that fully reconstructing multi-step derivations without partial clues poses a significant challenge.

Domain-Focused Models Show Improvements. Models that are specifically tuned for mathematical reasoning—such as Qwen2.5-Math—demonstrate improved performance over other models with a similar parameter count, both in terms of overall average accuracy and on individual subtasks. In particular, **Qwen2.5-Math-72B** achieves an average accuracy of 29.44%, which is on par with the average performance of most commercial models. However, fine-tuning general-purpose models like

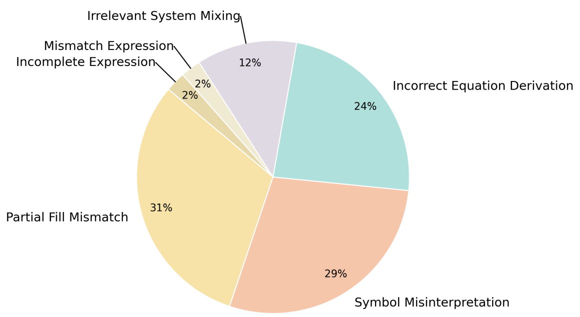

Figure 4: Error distribution among 40 annotated DeepSeek-R1 errors.

LLaMA to telecom-specific data (*e.g.*, LLaMA-3-8B-Tele) yields only limited benefits. This is likely because the telecom fine-tuning data predominantly consists of wireless protocols, whereas the problems in WirelessMathBench require handling long contexts and performing high-level mathematical reasoning.

4.3 Error Analysis

To better understand the limitations of the model-generated solutions, we randomly sampled and reviewed 40 failure answers by DeepSeek-R1 on the progressively masked filling and fully masked equation tasks, grouping them into several recurring categories. Figure 4 summarizes the distribution of these errors.

Partial Fill Mismatch (31%). A prevalent challenge in progressive masking tasks is that the model often merges multiple placeholders into a single bracket or places the correct term in the wrong mask position. For instance, it may combine $\sqrt{pK}\beta_{mk}\mathbf{y}_{pm}^{H}\boldsymbol{\phi}_k$ into one placeholder even though the prompt explicitly requests separate terms ($\sqrt{pK}\beta_{mk}$ and $\mathbf{y}_{pm}^{H}\boldsymbol{\phi}_k$) across two different masks.

Symbol Misinterpretation (29%). This type of error arises when the model chooses the wrong symbol or omits key symbolic elements in the final equation. An example is substituting $\mathbf{H}_{\mathrm{BR}}$ instead of $\mathbf{H}_{\mathrm{BR}}^{H}$ in a channel derivation.

Incorrect Equation Derivation (24%). Several solutions fail to follow the correct derivation path, either missing crucial intermediate steps or injecting extraneous components. In longer sequences, a single early mistake (e.g., confusing pilot power p with user transmit power ρ_k) tends to propagate, causing the final expression to be structurally flawed despite appearing superficially similar.

Irrelevant System Mixing (11%). We also observe instances in which the model introduces extraneous terms or assumes mismatched system settings. For example, it might inject NOMA-based interference factors into an RIS–MIMO scenario with no multi-user superposition, or switch to an entirely different beamforming constraint unconnected to the original problem statement.

Other Errors (4%). A small fraction of errors are too context-specific to classify under the main categories. These include incomplete expressions—where the answer stops abruptly without filling the entire formula—and mismatched expressions that repeat known placeholders without substituting the correct variables.

Overall, while the majority of failures fall into coherent error patterns, it is evident that the model struggles when tasks require nuanced symbol-to-mask matching or integrative domain reasoning. Therefore, strengthening the model's ability to accurately derive reasoning and align domain knowledge is a key challenge for future improvements.

5 Conclusion

We introduced **WirelessMathBench**, the first benchmark that comprehensively evaluates LLMs' abilities to domain-specific mathematical engineering tasks in wireless communications. By presenting a broad range of tasks—from multiple-choice questions to progressively masked fill-in-the-blank and full equation completions—WirelessMathBench assesses how well models handle real-world wireless communications mathematical engineering challenges. Our experiments show that, although many leading LLMs perform well on simpler question types, their ability to reconstruct equations deteriorates significantly when partial or full derivation is required, exposing a critical shortfall in current LLM-assisted scientific innovation. Future work will expand the scope and complexity of these wireless challenges, with the aim of enhancing LLMs' mathematical reasoning and domain adaptation. By advancing their integration into the next-generation wireless systems, we ultimately strive toward the development of more capable, general-purpose AI solutions for scientific and engineering applications.

Ethical Considerations

This paper focuses on the development of a benchmark for evaluating language models on mathematical modeling tasks in wireless communications. The source data of WirelessMathBench is curated from open-access research papers, ensuring that the benchmark is built on publicly available information. Meanwhile, we resummarize the papers and anonymize the content to prevent any potential privacy concerns. In experiments, we follow all licensing agreements and terms of service for the models evaluated, ensuring that our work is conducted in compliance with ethical guidelines.

Limitations

While WirelessMathBench provides a comprehensive evaluation of LLMs on wireless mathematical modeling tasks, several limitations remain. First, it mainly covers text-based problems (e.g., symbolic derivations), missing other key data types like antenna diagrams, simulation plots, and Radio frequency (RF) measurements measurements, which are crucial for real-world wireless tasks. Second, while WirelessMathBench spans topics from MIMO to RIS, it may not cover all emerging areas, such as quantum communication or terahertz systems. Third, our automated evaluation checks the final symbolic equivalence and dimensionality plausibility but may miss incorrect reasoning at intermediate steps. Lastly, all tests were done in a zero-shot setting. While this reflects real-world use, it does not explore whether fine-tuning or retrieval-based methods could improve results. Future versions of WirelessMathBench may include training splits to support domain adaptation and wireless-specific fine-tuning.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

Jiancheng An, Chau Yuen, Chongwen Huang, Mérouane Debbah, H. Vincent Poor, and Lajos Hanzo. 2023. A tutorial on holographic MIMO communications—part i: Channel modeling and channel estimation. *IEEE Communications Letters*, 27(7):1664–1668.

Jiancheng An, Chau Yuen, Chao Xu, Hongbin Li, Derrick Wing Kwan Ng, Marco Di Renzo, Mérouane Debbah, and Lajos Hanzo. 2024. Stacked intelligent metasurface-aided MIMO transceiver design. *IEEE Wireless Communications*, 31(4):123–131.

Daman Arora, Himanshu Gaurav Singh, and Mausam . 2023. Have LLMs advanced enough? a challenging problem solving benchmark for large language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Emil Bjornson, Michail Matthaiou, and Merouane Debbah. 2013. A new look at dual-hop relaying: Performance limits with hardware impairments. *IEEE Transactions on Communications*, 61(11):4512–4525.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Viveck R. Cadambe and Syed Ali Jafar. 2008. Interference alignment and degrees of freedom of the K-user interference channel. *IEEE Transactions on Information Theory*, 54(8):3425–3441.

Konstantin Chernyshev, Vitaliy Polshkov, Ekaterina Artemova, Alex Myasnikov, Vlad Stepanov, Alexei Miasnikov, and Sergei Tilga. 2024. U-MATH: A university-level benchmark for evaluating mathematical skills in llms. *arXiv preprint arXiv:2412.03205*.

Jinjin Chu, Rang Liu, Ming Li, Yang Liu, and Qian Liu. 2023. Joint secure transmit beamforming designs for integrated sensing and communication systems. *IEEE Transactions on Vehicular Technology*, 72(4):4778–4791.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, and et al. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Meng Fang, Xiangpeng Wan, Fei Lu, Fei Xing, and Kai Zou. 2024. MathOdyssey: Benchmarking mathematical problem-solving skills in large language models using odyssey math data. *arXiv preprint arXiv:2406.18321*.

Simon Frieder, Luca Pinchetti, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Petersen, and Julius Berner. 2024. Mathematical capabilities of chatgpt. *Advances in neural information processing systems*, 36.

David Gesbert, Stephen Hanly, Howard Huang, Shlomo Shamai Shitz, Osvaldo Simeone, and Wei Yu. 2010. Multi-cell MIMO cooperative networks: A new look at interference. *IEEE Journal on Selected Areas in Communications*, 28(9):1380–1408.

Google DeepMind. 2024. Introducing gemini 2.0: our new ai model for the agentic era. Accessed: 2024-12-11.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, and et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. 2024. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in Neural Information Processing Systems*, 36.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Taicheng Guo, Bozhao Nan, Zhenwen Liang, Zhichun Guo, Nitesh Chawla, Olaf Wiest, Xiangliang Zhang, et al. 2023. What can large language models do in chemistry? a comprehensive benchmark on eight tasks. *Advances in Neural Information Processing Systems*, 36:59662–59688.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. 2024. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3828–3850, Bangkok, Thailand. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021a. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Chongwen Huang, Sha Hu, George C. Alexandropoulos, Alessio Zappone, Chau Yuen, Rui Zhang, Marco Di Renzo, and Merouane Debbah. 2020. Holographic MIMO surfaces for 6G wireless networks: Opportunities, challenges, and trends. *IEEE Wireless Communications*, 27(5):118–125.

Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. 2024. Mlagentbench: Evaluating language agents on machine learning experimentation. In *Forty-first International Conference on Machine Learning*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R Narasimhan. 2024. SWE-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations*.

Rik Koncel-Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. 2016. Mawps: A math word problem repository. In *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 1152–1157.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. 2022. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada. Association for Computational Linguistics.

Mengbing Liu, Xin Li, Boyu Ning, Chongwen Huang, Sumei Sun, and Chau Yuen. 2022. Deep learning-based channel estimation for double-ris aided massive mimo system. *IEEE Wireless Communications Letters*, 12(1):70–74.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin Clement, Dawn Drain, Daxin Jiang, Duyu Tang, Ge Li, Lidong Zhou, Linjun Shou, Long Zhou, Michele Tufano, MING GONG, Ming Zhou, Nan Duan, Neel Sundaresan, Shao Kun Deng, Shengyu Fu, and Shujie LIU. 2021. CodeXGLUE: A machine learning benchmark dataset for code understanding and generation. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Ali Maatouk, Kenny Chirino Ampudia, Rex Ying, and Leandros Tassiulas. 2024. Tele-LLMs: A series of specialized large language models for telecommunications. *arXiv preprint arXiv:2409.05314*.

Ali Maatouk, Fadhel Ayed, Nicola Piovesan, Antonio De Domenico, Merouane Debbah, and Zhi-Quan Luo. 2023. TeleQnA: A benchmark dataset to assess large language models telecommunications knowledge. *arXiv preprint arXiv:2310.15051*.

Seyed Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. 2025. GSM-symbolic: Understanding the limitations of mathematical reasoning in large language models. In *The Thirteenth International Conference on Learning Representations*.

OpenAI. 2024. Hello gpt-4o. Accessed: 2024-10-01.

OpenAI. 2024. Learning to reason with llms. Accessed: 2025-09-12.

Jiawei Shao, Jingwen Tong, Qiong Wu, Wei Guo, Zijian Li, Zehong Lin, and Jun Zhang. 2024. Wirelessllm: Empowering large language models towards wireless intelligence. *arXiv preprint arXiv:2405.17053*.

Qingjiang Shi, Meisam Razaviyayn, Zhi-Quan Luo, and Chen He. 2011. An iteratively weighted mmse approach to distributed sum-utility maximization for a MIMO interfering broadcast channel. *IEEE Transactions on Signal Processing*, 59(9):4331–4340.

Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. 2025. Can LLMs generate novel research ideas? a large-scale human study with 100+ NLP researchers. In *The Thirteenth International Conference on Learning Representations*.

Q.H. Spencer, A.L. Swindlehurst, and M. Haardt. 2004. Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels. *IEEE Transactions on Signal Processing*, 52(2):461–471.

Liangtai Sun, Yang Han, Zihan Zhao, Da Ma, Zhennan Shen, Baocai Chen, Lu Chen, and Kai Yu. 2024. Scieval: A multi-level large language model evaluation benchmark for scientific research. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19053–19061.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. 2024. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019a. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019b. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

Xiaoxuan Wang, Ziniu Hu, Pan Lu, Yanqiao Zhu, Jieyu Zhang, Satyen Subramaniam, Arjun R Loomba, Shichang Zhang, Yizhou Sun, and Wei Wang. 2024. Scibench: Evaluating college-level scientific problem-solving abilities of large language models.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. 2024. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *Preprint*, arXiv:2409.12122.

Haifan Yin, David Gesbert, Miltiades Filippou, and Yingzhuang Liu. 2013. A coordinated approach to channel estimation in large-scale multiple-antenna systems. *IEEE Journal on Selected Areas in Communications*, 31(2):264–273.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng YU, Zhengying Liu, Yu Zhang, James Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2024. Metamath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations*.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567.

Cedegao E Zhang, Katherine M Collins, Adrian Weller, and Joshua B Tenenbaum. 2023. Ai for mathematics: A cognitive science perspective. *arXiv preprint arXiv:2310.13021*.

Jiakang Zheng, Jiayi Zhang, Hongyang Du, Dusit Niyato, Bo Ai, Mérouane Debbah, and Khaled B. Letaief. 2024. Mobile cell-free massive MIMO: Challenges, solutions, and future directions. *IEEE Wireless Communications*, 31(3):140–147.

Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. 2022. miniF2F: a cross-system benchmark for formal olympiad-level mathematics. In *International Conference on Learning Representations*.

Hang Zou, Qiyang Zhao, Yu Tian, Lina Bariah, Faouzi Bader, Thierry Lestable, and Merouane Debbah. 2024. TelecomGPT: A framework to build telecom-specfic large language models. *arXiv preprint arXiv:2407.09424*.

A Dataset Details

A.1 Topics and Papers Selection

Our study addresses modern wireless communication challenges by selecting topics based on three key criteria that ensure both the academic rigor and practical relevance of our work. First, we target topics that have undergone peer review and have been accepted by prestigious journals such as IEEE Transactions on Wireless Communications (TWC), IEEE Transactions on Communications (TCOM), and IEEE Journal on Selected Areas in Communications (JSAC), as well as by top-tier conferences including IEEE International Conference on Communications(ICC) and IEEE Global Communications Conference (Globalcom), and for which corresponding arXiv versions are available. Second, we focus on communication system modeling that entails complex, multi-step mathematical derivations. These derivations are designed to closely mimic the challenges encountered in real-world wireless communication scenarios, capturing the intricate interplay between theoretical constructs and practical system constraints. Third, we ensure topic diversity by covering a wide range of wireless communication scenarios and problem domains. Specifically, our study encompasses seven major communication scenarios and six key problem areas, including interference management, spectrum optimization, network coding, and energy efficiency.

A.2 ArXiv Data Processing

Our data processing pipeline is similar with (Maatouk et al., 2024). First begins with the removal of all comments from the LaTeX files using Google's arXiv LaTeX Cleaner[1]. We then parse the LaTeX source to extract the core technical content by separating the main text and mathematical expressions from non-essential elements such as comments, figures, and tables. For submissions comprising multiple files linked via `\input` commands, we utilize the `latexpand` tool[2] to flatten the document into a single file, ensuring all dependencies are resolved. To address the variability introduced by author-defined macros (e.g., via `\newcommand` or `\def`), we automatically expand these using the `de-macro`[3], replacing custom macros with their full definitions and normalizing all mathematical expressions to a consistent LaTeX format. Non-informative content such as acknowledgments and extensive bibliographies are removed to focus on technical material and to ensure anonymity in dataset construction by removing all author information from the articles.

B Prompt Templates

For clarity and reproducibility, we provide examples of our prompt templates. Figure 10 shows a template for a paper summary prompt, Figure 11 illustrates a question generation prompt, and Figures 12 andFigure 14 present templates for requesting LLMs to answer multiple-choice and fill-in-the-blank questions, respectively.

C Model Configurations and Hyperparameters

In our WirelessMathBench experiment, a total of 16 models were tested. All model tests followed the same template and set default parameters, and all results are the results of a single run.

C.1 Closed-source Models

For models such as OpenAI-o1, GPT-4o, GPT-4, GPT-3.5-turbo, Gemini-2.0-flash, Gemini-1.5-pro, and Gemini-1.5-flash, we utilize their official API interfaces. These models are invoked via their respective API endpoints with standardized default parameters to ensure consistency and reproducibility across all experiments.

[1] https://github.com/google-research/arxiv-latex-cleaner
[2] https://ctan.org/pkg/latexpand
[3] https://ctan.org/pkg/de-macro

C.2 Open-source Models

Our local experiments employ several open-source models deployed across different environments:

- **AliyunCloud Deployment[4]:** Qwen2.5-Math-72B and DeepSeek-V3 are deployed on the AliyunCloud platform.
- **NVIDIA NIM Deployment[5]:** DeepSeek-R1 and LLaMA-3.3-70B are run on NVIDIA NIM cloud platform.
- **HuggingFace Transformers[6]:** Other models—including Qwen2.5-Math-7B, LLaMA-3-8B-Tele, LLaMA-3-8B, and Mistral-7B—are run on local Nvidia A6000s, using the HuggingFace Transformers library to load pre-trained models.

D Example Output

To illustrate the diversity and complexity of the tasks within WirelessMathBench, we now present representative example outputs generated by various models. Figures 5, Figure 6, Figure 7, Figure 8, and Figure 9 showcase different task types along with the responses provided by leading models (OpenAI-o1 and DeepSeek-R1).

Disclosure

In the process of writing this paper, we partially utilized ChatGPT as a language polishing tool to improve the clarity and quality of the text. However, all research ideas, data analyses, and conclusions were independently conceived and confirmed by the authors.

[4] https://bailian.console.aliyun.com/

[5] https://build.nvidia.com/

[6] https://huggingface.co/docs/transformers/en/index

Q_id:liu2023detecting_q3_mc.

Backgroud	MRC combining yields SINR with coherent combining gain (numerator) and interference\/noise terms (denominator). Key components include LoS power $\|\bar{\mathbf{g}}_k\|^2$, estimation quality γ_{mk}, and covariance trace $\mathrm{tr}(\mathbf{\Theta}_k)$
Question	Which numerator represents coherent combining gain?
Optioins	A: $\left(\|\bar{\mathbf{g}}_k\|^2 + pK\mathrm{tr}(\mathbf{\Theta}_k)\right)^2$ B: $\left(\|\bar{\mathbf{g}}_k\|^2 + \sum_{m=1}^{M} \gamma_{mk}\right)^2$ C: $\left(\|\bar{\mathbf{g}}_k\|^2 + pK\mathrm{tr}(\mathbf{\Theta}_k) + \sum_{m=1}^{M} \gamma_{mk}\right)$ D: $\left(\|\bar{\mathbf{g}}_k\|^2 + pK\mathrm{tr}(\mathbf{\Theta}_k) + \sum_{m=1}^{M} \gamma_{mk}\right)^2$

Correct Answer D

OpenAI-o1 B

{
"explanation": "In MRC combining, the coherent combining gain in the numerator of the SINR expression represents the squared magnitude of the combined channel gains. The gain is proportional to the square of the sum of the deterministic LoS component and the estimated channel components.\n\nThe LoS power is given by \(\|\bar{\mathbf{g}}_k\|^2\). The estimation quality per antenna is given by \(\gamma_{mk}\) for antennas \(m = 1,\,\ldots,\,M\).\n\nTherefore, the total coherent combining gain is represented by:\n\[\n\left(\|\bar{\mathbf{g}}_k\|^2 + \sum_{m=1}^{M} \gamma_{mk} \right)^2\n\]\n\nThis expression corresponds to **Option B**.\n\nOptions A and D include \(pK\mathrm{tr}(\pmb{\Theta}_k)\) in the numerator, which typically represents interference or noise terms, not the coherent signal gain.\n\nOption C does not square the sum, which is inconsistent with the fact that coherent combining gain is proportional to the square of the combined channel gains.\n\nTherefore, Option B is the correct representation of the coherent combining gain in the numerator.",
"answer": "B"
}

DeepSeek-R1 D

{
"explanation": "The coherent combining gain in MRC arises from the constructive addition of signal components across antennas, leading to a squared term in the SINR numerator. The key components include the LoS power $\|\bar{\mathbf{g}}_k\|^2$, estimation quality $\sum_{m=1}^{M} \gamma_{mk}$, and covariance trace $pK\mathrm{tr}(\mathbf{\Theta}_k)$. These terms collectively contribute to the effective signal power after combining. Option D correctly incorporates all three components within a squared sum, reflecting the coherent gain from both deterministic (LoS) and statistical (estimation quality, covariance) channel properties. Other options either omit critical terms (e.g., covariance trace in B) or lack the necessary squaring operation (C), which is essential for coherent gain.",
"answer": "D"
}

Figure 5: An example question and the corresponding output from LLMs for a multiple-choice task.

Q_id:liu2021deep_q1_fillin_1.

Backgroud

In an IRS-assisted MU-MIMO system, the base station receives signals through both direct and IRS-reflected paths. The observation matrix $\boldsymbol{X}_k$ combines the channel matrix $\boldsymbol{H}_k$, IRS phase-shift matrix $\boldsymbol{P}$, and additive noise $\boldsymbol{Z}_k$. Here, \$M\$ is BS antennas, N IRS elements, and C pilot symbols.

Question

What noise term completes the received signal equation?

Equation

$$\boldsymbol{X}_k = \boldsymbol{H}_k\boldsymbol{P} + [MASK]$$

Correct Answer

$\boxed{\boldsymbol{Z}_k}$

OpenAI-o1

$\boxed{\boldsymbol{Z}_k}$

```
{
  "explanation": "The additive noise term \(\boldsymbol{Z}_k\) completes the received signal equation by
    accounting for random disturbances in the system.",
  "answer": "\boxed{\boldsymbol{Z}_k}"
}
```

DeepSeek-R1

$\boxed{\boldsymbol{Z}_k}$

```
{
  "explanation": "In the given IRS-assisted MU-MIMO system, the received signal equation combines the channel
    matrix $\boldsymbol{H}_k$, IRS phase-shift matrix $\boldsymbol{P}$, and additive noise
    $\boldsymbol{Z}_k$. The additive noise term $\boldsymbol{Z}_k$ is explicitly mentioned in the
    background context as part of the observation matrix $\boldsymbol{X}_k$. The equation structure
    $\boldsymbol{X}_k = \boldsymbol{H}_k \boldsymbol{P} + [MASK]$ directly implies that the missing term
    is the additive noise $\boldsymbol{Z}_k$, which aligns with standard wireless communication models where
    noise is added linearly to the signal component.",
  "answer": "\boxed{\boldsymbol{Z}_k}"
}
```

Figure 6: An example of a level 1 progressively masked fill-in-the-blank task. The figure shows both the question and the LLM's output as it begins to reconstruct the missing parts of the equation.

Q_id:zhao2024dual_q1_fillin_2.

Backgroud	In RIS-aided ISAC systems, the effective channel combines direct BS-user links and RIS-reflected paths. The RIS phase matrix $\boldsymbol{\Theta}$ enables configurable signal reflections. Channel matrices maintain $N_U \times N_B$ dimensions through proper conjugation.
Question	What matrix configures RIS phase shifts?
Equation	$\mathbf{H}_{\mathrm{c},k} = \mathbf{H}_{\mathrm{BU},k} + [MASK]\boldsymbol{\Theta}[MASK]$

Correct Answer	$\boxed{\mathbf{H}_{\mathrm{RU},k}}$ and $\boxed{\mathbf{H}_{\mathrm{BR}}^{H}}$

OpenAI-o1

$\boxed{\mathbf{H}_{\mathrm{RU},k}}$ and $\boxed{\mathbf{H}_{\mathrm{BR}}}$


```
{
  "explanation": "The two missing channel matrices are \(\mathbf{H}_{\text{RU},k}\) and \
    (\mathbf{H}_{\text{BR}}\).",
  "answer": "\boxed{\mathbf{H}_{\text{RU},k}},\boxed{\mathbf{H}_{\text{BR}}}"
}
```


DeepSeek-R1

$\boxed{\mathbf{H}_{\mathrm{RU},k}}$ and $\boxed{\mathbf{H}_{\mathrm{BR}}}$


```
{
  "explanation": "In RIS-aided systems, the effective channel combines the direct BS-user path (H_{BU,k}) and
    the RIS-reflected path. The reflected path involves the channel from BS to RIS (H_{BR}) and RIS to user
    (H_{RU,k}), with the RIS phase matrix Θ in between. Matrix multiplication order ensures dimensions align:
    H_{RU,k} (N_U × N_R) · Θ (N_R × N_R) · H_{BR} (N_R × N_B) results in N_U × N_B, matching
    H_{BU,k}. Thus, the missing terms are H_{RU,k} and H_{BR}.",
  "answer": "\boxed{\mathbf{H}_{\text{RU},k}},\boxed{\mathbf{H}_{\text{BR}}}"
}
```


Figure 7: An example of a level 2 progressively masked fill-in-the-blank task.

Q_id:xu2021a_q1_fillin_3.

Backgroud	Satellite line-of-sight channels use planar antenna arrays with spatial correlation. The LoS component $\overline{\mathbf{g}}_k$ depends on Rician factor κ_k, path-loss β_k, and user angles θ_k (elevation), ω_k (azimuth). Array response vector $\pm b\{ell\}$ uses wavelength λ and element positions $\mathbf{c}_n$.
Question	Provide scaling factor and first element's phase
Equation	$bar\mathbf{g}_k = [MASK]\left[e^{j[MASK]}, \ldots, e^{j\boldsymbol{\ell}(\theta_k,\omega_k)^T\mathbf{c}_N}\right]^T$

Correct Answer	$\boxed{\sqrt{\dfrac{\kappa_k\beta_k}{\kappa_k+1}}}$ and $\boxed{\boldsymbol{\ell}(\theta_k,\omega_k)^T\mathbf{c}_1}$

OpenAI-o1

$\boxed{\sqrt{\dfrac{\kappa_k\beta_k}{\kappa_k+1}}}$ and $\boxed{0}$

{
"explanation": "The scaling factor for the LoS component is \sqrt{\dfrac{\kappa_k \beta_k}{\kappa_k + 1}} due to the Rician factor and path-loss. The first element's phase is zero because the array factor at the origin is zero, so the exponential term becomes e^{j0}=1.",
"answer": "\boxed{\sqrt{\dfrac{\kappa_k \beta_k}{\kappa_k + 1}}},\boxed{0}"
}

DeepSeek-R1

$\boxed{\sqrt{\dfrac{\kappa_k\beta_k}{\kappa_k+1}}}$ and $\boxed{\boldsymbol{\ell}(\theta_k,\omega_k)^T\mathbf{c}_1}$

{
"explanation": "The scaling factor for the LoS component $\bar{\mathbf{g}}_k$ combines the path-loss β_k and Rician factor κ_k as $\sqrt{\frac{\kappa_k \beta_k}{\kappa_k + 1}}$. The first element's phase is determined by the inner product of the array response vector function $\pmb{\ell}(\theta_k, \omega_k)$ with the first antenna element's position $\mathbf{c}_1$.",
"answer": "\boxed{\sqrt{\frac{\kappa_k \beta_k}{\kappa_k + 1}}}\, \boxed{\pmb{\ell}(\theta_k, \omega_k)^T \mathbf{c}_1}"
}

Figure 8: An example of a level 3 progressively masked fill-in-the-blank task. Notice that both OpenAI-o1 and DeepSeek-R1 encountered difficulties in correctly interpreting the conjugate condition required by the task.

Q_id:an2023stacked_q1_fillin_4.

Backgroud	In a stacked intelligent metasurface (SIM) with L layers and N meta-atoms per layer, each layer applies a diagonal phase-shift matrix to incoming signals. Denote the matrix of the l-th layer by $\mathbf{\Phi}^l$. The diagonal entries contain the phase terms $e^{j\theta^l_n}$, controlling how each meta-atom shifts the wave\u2019s phase.","question_text":"Write the full expression for the phase-shift matrix of layer l.
Question	"Write the full expression for the phase-shift matrix of layer l.
Equation	$\mathbf{\Phi}^l = [MASK]$

Correct Answer $\boxed{\mathrm{diag}\Bigl(e^{j\theta^l_1},\, e^{j\theta^l_2},\, \dotsc,\, e^{j\theta^l_N} \Bigr)}$

OpenAI-o1 $\boxed{\mathrm{diag}\Bigl(e^{j\theta^l_1},\, e^{j\theta^l_2},\, \dotsc,\, e^{j\theta^l_N} \Bigr)}$

```
{
  "explanation": "The phase-shift matrix \( \boldsymbol{\Phi}^l \) is a diagonal matrix with elements \( e^{j\theta_n^l} \) along the diagonal for each meta-atom \( n = 1, 2, \dotsc, N \). Using the \( \mathrm{diag} \) function and following the LaTeX protocols: functions in roman font, scaled parentheses via \( \Bigl( \) and \( \Bigr) \), commas with small spaces via \( ,\, \), and braced exponents \( e^{\{...\}} \), the expression becomes:",
  "answer": "\boxed{\mathrm{diag}\Bigl( e^{j\theta^l_1},\, e^{j\theta^l_2},\, \dotsc,\, e^{j\theta^l_N} \Bigr)}"
}
```

DeepSeek-R1 $\boxed{\mathrm{diag}\Bigl(e^{j\theta^l_1},\, e^{j\theta^l_2},\, \ldots,\, e^{j\theta^l_N} \Bigr)}$

{
"explanation": "The phase-shift matrix for layer l in a SIM is a diagonal matrix where each diagonal entry corresponds to the phase shift $e^{j\theta^l_n}$ applied by the n-th meta-atom in the l-th layer. Using the diag function with scaled parentheses and properly formatted arguments, the matrix is constructed as $\mathrm{diag}\Bigl(e^{j\theta^l_1},\, e^{j\theta^l_2},\, \ldots,\, e^{j\theta^l_N} \Bigr)$.",
"answer": "\boxed{\mathrm{diag}\Bigl(e^{j\theta_1^l},\, e^{j\theta_2^l},\, \ldots,\, e^{j\theta_N^l} \Bigr)}"
}

Figure 9: An example of a Full Equation Completion (FEC) task.

Paper Summary Prompt Template.

```
Task: Act as an expert in wireless communications and mathematical modeling.
You will be provided with a research paper in the field of wireless communications.
Your goal is to extract and summarize the mathematical system modeling contained in
the paper.
  <<STRUCTURE REQUIREMENTS>>
1. **Model Extraction**:
  a) Identify ALL system equations with context
  b) For each equation:
    i) List ALL variables with units/dimensions
    ii) Specify underlying assumptions
    iii) Note domain restrictions

2. **Summary Organization**:
  \paragraph{Background} (2-3 sentences contextualizing the model)
  \paragraph{Key Assumptions} (bullet points with $\bullet$)
  \paragraph{Parameter Definitions} (table-like structure using \quad)
  \paragraph{Core Equations} (numbered with original labels if available)

3. **Equation Formatting**:
  - Vectors: \boldsymbol{v}
  - Matrices: \mathbf{M}
  - Operators: \mathrm{diag}, \mathrm{tr}
  - Complex numbers: j for imaginary unit
  - Dirac notation: \delta(\cdot)
  - Alignment: \begin{align*}...\end{align*}

<<CONTENT GUIDELINES>>
• **Variable Explanations**:
 - For each symbol: $\theta$ \quad (Type: Phase shift; Domain: [0,2π); Unit: rad)
 - Matrix dimensions: $\mathbf{H} \in \mathbb{C}^{N\times M}$
 - Distinguish similar symbols: $h_{ij}$ vs $h_{i}^{(j)}$

• **Model Validation**:
 - Verify dimensional consistency
 - Check boundary conditions
 - Confirm parameter unit homogeneity

• **Special Notation**:
 - Estimated quantities: $\hat{\mathbf{x}}$
 - Time derivatives: $\dot{\mathbf{v}}$
 - Optimal values: $\mathbf{\Theta}^\ast$

<<EXAMPLE OUTPUT>>
\paragraph{Background}
Consider an RIS-assisted mmWave system with $K$ single-antenna users
communicating via $N$ RIS elements...
<Key Assumptions>

\textbf{1) Effective Channel Representation.}\\
The received signal at the BS from user \(k\) combines the single-reflection and
double-reflection links......
The equivalent channel from user \(k\) to the BS is
\begin{equation}\label{eq:EffectiveChannel}
..........
\end{equation}
<Parameter Definitions> such as $\boldsymbol{h}_{k1} \in \mathbb{C}^{N \times
1}$ is the channel from user \(k\) to \(\mathrm{RIS}_1\).
\paragraph{System model with equations.}
```

Figure 10: This template is used to generate concise summaries of research papers.

Question Generation Prompt Template.

```
Task: Act as an expert in wireless communications and mathematical modeling.
You will generate exam-style questions from research paper summaries with the
following strict requirements:

<<STRUCTURE REQUIREMENTS>>
1. **Per Equation Processing**:
   a) Identify ALL system model equations, explain each symbol in the equation in
background
   b) For EACH equation:
      i) Mask the RHS with [MASK]
      ii) Generate 1 MCQ with 4 plausible options
      iii) Create 4 progressive fill-in-the-blank subquestions with:
           - 25%, 50%, 75%, and 100% key symbols masked
           - Variable masking in different positions
      iv) Ensure each subquestions can be answered independently and full self-
containment (no external references)

2. **Question Components**:
   - For MCQs:
     * \textbf{Background}: Contextual info in 3-5 lines
     * \textbf{Equation}: Masked equation in display math mode
     * \textbf{Question}: Explicitly ask to replace [MASK]
     * \textbf{Options}: 4 LaTeX-formatted choices (A)-(D)
     * \textbf{Answer}: Detailed derivation walkthrough
   - For Fill-in-the-blanks:
     * \textbf{Masked Equation}: Progressive symbol replacement
     * \textbf{Question}: Specific term request
     * \textbf{Answer}: Complete equation with \boxed{} for solutions

<<CONTENT GUIDELINES>>
• **Distractor Design**:
  - Below are some common error types:
    1) Matrix dimension mismatches
    2) Incorrect sequence (θ1 vs θ2)
    3) Missing diag() operators
    4) Channel matrix transposition errors
    5) Incorrect matrix multiplication
    6) Incorrect vector/matrix dimensions
    7) Random mistakes

• **Mathematical Rigor**:
  - Use \mathrm{diag}() for diagonalization
  - Replace symbolic writing in the problem appropriately, but ensure correctness
  - Verify matrix multiplication compatibility

• **Pedagogical Elements**:
  - Add \underbrace annotations in answers
  - Use $;$; for proper equation spacing
  - Include \text{} descriptions for terms

• **Masking Strategy**:
  - Mask variables not operators
  - Progressively increase masked terms:
    1) 25%: Single critical variable
    2) 50%: Two interdependent terms
    3) 75%: Multiple components
    4) 100%: Full equation recall

• **Blank Positioning**:
  - 25% mask: Key parameter (e.g., θ values)
  - 50% mask: Combined terms (e.g., N_jdiag(h_kj))
  - 75% mask: Structural components
  - 100% mask: Entire RHS

<<FORMATTING RULES>>
• Strict LaTeX compliance:
  - Equations in $$ $$/equation* environments
  - \boldsymbol for vectors/matrices
  - \mathrm for operators (diag, etc)
  - \quad spacing between equation terms

• Section headers with:
  \textbf{\Large Question X (System Component)}
  \vspace{-0.5em}  % Compact vertical spacing

<<EXAMPLE TEMPLATE>>
…………
```

Figure 11: This prompt template converts paper summaries into detailed question–answer pairs.

Multiple-Choice Question Prompt Template.

You are a domain expert in Wireless Communication. Please answer the following multiple-choice question.

Background Context
{background}

Multiple-Choice Question
{question_text}

Relevant Equation
{equation or "No equation provided"}

{options_str}

Formatting Requirements
{latex_core_rules}
- Explanations requiring math must use . . . for inline equations
- Maintain consistent notation with question context

Response Instructions
1. Select ONE letter choice (A-D)
2. Provide technical reasoning in the explanation field
3. Present your final answer in a JSON format. This should include: answer and explanation

Your output should be formatted as a JSON object enclosed in Markdown code blocks tagged with 'json'. For example:
`json
{{
"explanation": "<explanation>"
"answer": "C",
}}
`

Figure 12: This template is designed for answering multiple-choice questions. The model is guided to choose the correct mathematical expression from a set of closely related options.

Fill-in-the-Blank and Full Eqaution Completion Question Prompt Template.

You are a domain expert in Wireless Communication. Please answer the following fill-in-the-blank question.

Background Context
{background}

Problem Statement (Blanks marked with [MASK])
{question_text}

Equation to Complete
{equation or "No equation provided"}

Strict LaTeX Protocol
{latex_core_rules}
5. Each [MASK] requires a separate \boxed{{...}}
6. Final answer line format:
 The final answer is \boxed{{...}},\boxed{{...}},...

Submission Requirements
1. Answer box number corresponds to the [MASK] number
2. Use EXACT formatting from the equation/question
3. No natural language in boxed answers
4. Technical explanation in the 'explanation' field

Your output should be formatted as a JSON object enclosed in Markdown code blocks tagged with 'json'. For example:
`json
{{
 "explanation": "<explanation>“,
 "answer": "\boxed{<answer1>}, \boxed{<answer2>}, \boxed{<answer3>}",

}}
`

Figure 13: This prompt template is used for fill-in-the-blank and full equation completion tasks. It directs the model to reconstruct missing parts of equations by using contextual cues and domain knowledge, simulating the process of step-by-step derivation.

LLM Evaluation Answer Prompt Template.

````
You are an expert in wireless communications and mathematical modeling. Your task
is to evaluate a student's answer against the correct answer.
Follow these evaluation criteria strictly:

1. **Mathematical Equivalence:** Check if the student's answer is mathematically
equivalent to the correct answer.
2. **Answer Format:** The student's answer should be in the same format as the
correct answer.
3. **All blanks should be filled.** All the blanks should be filled with the correct
answer.
4. **Scoring:** Output a score of "1" if the student's answer is correct (i.e.,
mathematically equivalent) or "0" if it is not.

Please provide your final output as a JSON object (enclosed in Markdown code blocks
tagged with 'json') with the following format:

```json
{{
  "score": "1"
}}
```
or
```json
{{
  "score": "0"
}}
```

Here is the background:

{background}

Here is the question:

{question}

Here is the question:

{question}

Here is the correct answer:

{true_answer}

Here is the student's answer:

{student_answer}

Evaluate the student's answer based solely on the above information and output only
the JSON object with the score.
"""
````

Figure 14: This evaluation prompt template standardizes the process of assessing model-generated answers. It ensures that responses are judged consistently based on their correctness, completeness, and adherence to the required domain-specific reasoning.