TSNBench: Benchmarking LLM Proficiency in Time-Sensitive Networking

Rubi Debnath[1*] **Daniel Bujosa Mateu**[2] **Luxi Zhao**[3]
Silviu S. Craciunas[2,4] **Paul Pop**[2] **Sebastian Steinhorst**[1]
[1]Technical University of Munich, Munich, Germany
[2]Technical University of Denmark, Kongens Lyngby, Denmark
[3]Beihang University, Beijing, China
[4]NXP Semiconductors, Vienna, Austria

Abstract

We present TSNBench, the first benchmark for evaluating large language model (LLM) proficiency in Time-Sensitive Networking (TSN), a suite of IEEE 802.1 standards for deterministic communication with bounded latency in safety-critical domains such as autonomous vehicles, aviation, defense, and industrial automation. While LLMs have been extensively evaluated on general knowledge tasks, their capabilities in safety-critical networking domains remain largely unexplored. TSNBench comprises 939 expert-validated multiple-choice questions (MCQs) covering diverse TSN mechanisms, along with 100 open-ended Worst-Case Delay (WCD) computation tasks for Credit-Based Shaper (CBS) and Cyclic Queuing and Forwarding (CQF) across varying network topologies and traffic conditions. MCQ answers are validated by domain experts, and open-ended ground truth WCD values are computed using a verified Network Calculus (NC) solver for CBS and closed-form mathematical upper bounds for CQF. We evaluate 16 LLMs and find that although models achieve 67 to 95% accuracy on MCQs, they fail substantially on open-ended WCD computation. For CBS, only GPT-5 achieves a Mean Absolute Percentage Error (MAPE) of 36.2%, meaning its predicted WCD deviates by 36.2% of the actual TSN flow delay on average, while most models exceed 80%. For CQF, the best model achieves 41.8% MAPE, with most models clustering between 80% and 100%. Such errors are large relative to TSN latency budgets and can lead to violations of real-time constraints and unsafe configurations. TSNBench demonstrates that MCQ benchmarks may overestimate LLM capabilities in safety-critical networking domains.

1 Introduction

Recent advances in large language models (LLMs) across different domains such as engineering [Jackson et al., 2025, Guo et al., 2025], medicine [Xie et al., 2025, Liu et al., 2023, Li et al., 2024], clinical practice [Kweon et al., 2024], computer networking [Sharma and Yegneswaran, 2023], telecommunications [Maatouk et al., 2026, Ferrag et al., 2026, Oluwaseyi et al., 2025, Gajjar et al., 2025], and automation [Shen et al., 2024] have shown groundbreaking performance in assisting engineers, practitioners, researchers [Huang et al., 2023, Sun et al., 2024], and doctors in solving real-world problems. System engineers are increasingly using LLMs to design and configure networks [Wang et al., 2024a], generate code, and analyze network logs. With this, they are entering new territory: safety-critical application domains such as autonomous vehicles, aerospace [Fiori et al., 2024, Sanchez-Garrido et al., 2021], defense [Elliott, 2023], and industrial communication [Zhang et al.,

*Corresponding Author.

Preprint.

2024]. In these contexts, the accuracy, reliability, and consistency of LLMs become far more than leaderboard metrics, as they become engineering requirements.

Time-Sensitive Networking (TSN) [802, 2018], standardized by the IEEE 802.1 Working Group (WG), is a layer-2 Ethernet technology that provides deterministic communication guarantees for safety-critical applications. TSN deployments typically separate traffic based on timing criticality. Safety-critical periodic communication with guaranteed latency and bounded jitter is categorized as time-triggered (TT) [Ademaj et al., 2019] traffic and is served using the IEEE 802.1Qbv timed-gate mechanism. TT transmissions are controlled by a Gate-Control List (GCL), computed offline using exact methods such as SMT-based synthesis [Craciunas et al., 2016] or heuristic approaches [Pop et al., 2016, Gavriluţ et al., 2018, Bujosa et al., 2022]. In contrast, periodic or sporadic communication requiring bounded end-to-end latency but less stringent jitter control is classified as Audio Video Bridging (AVB) stream traffic [Böhm and Wermser, 2021, Bruckner et al., 2019]. Consequently, Worst-Case Delay (WCD) estimation errors of tens or hundreds of microseconds are significant, as they can consume timing margins, violate deadlines, or lead to infeasible TSN configurations. In mission-critical deployments, such errors can have severe consequences. A misconfigured TSN network can cause, for example, a robotic arm to miss a critical assembly step, a brake system to fail on a highway, an aircraft control system to respond incorrectly, a defense mechanism to collapse, or a spacecraft to miss a vital signal. These failures may result from sub-millisecond timing violations caused by a single misconfiguration. These risks highlight the importance of accurate analysis and configuration in TSN systems, especially as LLMs are increasingly integrated into network management workflows. Therefore, their domain proficiency must be rigorously evaluated. However, to the best of our knowledge, no existing benchmark evaluates LLM proficiency in TSN.

To fill this gap, we introduce TSNBench, the first benchmark for evaluating LLM proficiency in TSN, comprising two complementary evaluation components. The first is a 939-question expert-validated multiple-choice question and answer (MCQA) dataset, generated from 83 peer-reviewed research papers using three LLMs from distinct model families and rigorously reviewed by five domain experts, each with over eight years of TSN research experience. The second is a set of open-ended questions requiring multi-step WCD computation for two widely deployed TSN mechanisms, namely Credit-Based Shaper (CBS) [802, 2010] and Cyclic Queuing and Forwarding (CQF) [802, 2017, Yan et al., 2020], across varying network topologies and traffic flows, with ground truth computed using a verified Network Calculus (NC) solver [Zhao et al., 2018] for CBS and closed-form mathematical upper bounds for CQF [Wang et al., 2023]. These open-ended WCD questions are intended as a closed-book stress test of standalone model capability, rather than as a deployment workflow for free-text LLM timing outputs. Detailed background on TSN, NC, CBS, and CQF is provided in Appendix 7, 8, 9, and 10, respectively.

While general-purpose benchmarks such as MMLU [Hendrycks et al., 2021] and MMLU-Pro [Wang et al., 2024b] evaluate broad subject knowledge spanning elementary mathematics, history, and law, they are fundamentally unsuited for safety-critical domain-specific evaluation. Answering a multiple-choice question about elementary school history is categorically different from answering TSN terminology questions and correctly computing a WCD under NC constraints for a given network topology. Without a benchmark that captures this distinction, there is no principled way to measure LLM progress in deterministic networking domains. TSNBench is designed precisely to expose this gap.

We evaluate 16 LLMs comprising open-source and closed-source models, as well as general-purpose and reasoning-specialized architectures. Our results reveal a striking dissociation, where models achieve 67 to 95% accuracy on MCQA yet fail substantially on open-ended WCD computation. The best-performing model, GPT-5, achieves a Mean Absolute Percentage Error (MAPE) of 36.2% on CBS, while most models exceed 80%. This is concerning in a domain where timing violations of tens of microseconds, even 1% of a 1000 μs deadline, may cause system failures.

Our key contributions are:

1. **First expert-validated TSN benchmark:** TSNBench evaluates LLM knowledge of TSN mechanisms through 939 expert-validated MCQs derived from peer-reviewed TSN literature.

2. **Open-ended timing-analysis tasks:** TSNBench includes open-ended WCD computation tasks for CBS and CQF with ground truth computed using a verified NC solver for CBS and closed-form mathematical bounds for CQF.

3. **Evaluation across 16 LLMs:** We evaluate both open-source and closed-source models, including general-purpose and reasoning-specialized models, and show that high MCQA accuracy does not reliably predict accurate WCD computation.

In summary, TSNBench provides the research community with the first rigorous evaluation resource for LLM proficiency in TSN, offering valuable insights to both the real-time networking community exploring LLM-assisted TSN management and the machine learning community seeking to understand the limits of LLMs in safety-critical, computationally demanding domains.

2 Related Work

General LLM Benchmarks: Benchmarking and datasets are essential for measuring LLM progress and identifying key gaps and limitations [Hendrycks et al., 2021, Wang et al., 2024b]. General knowledge benchmarks such as MMLU [Hendrycks et al., 2021] and MMLU-Pro [Wang et al., 2024b] evaluate broad subject knowledge including elementary mathematics, history, computer science, and law, using multiple-choice questions. Domain-specific benchmarks have extended this paradigm to medicine [Xie et al., 2025, Liu et al., 2023, Li et al., 2024], clinical practice [Kweon et al., 2024], law [Guha et al., 2023], code generation [Hua et al., 2025, Huang et al., 2024], and scientific research [Sun et al., 2024]. While these benchmarks have driven significant progress, they are not designed to evaluate safety-critical networking tasks. Most rely on multiple-choice evaluation, and none assess whether a model can perform the multi-step computational reasoning required in safety-critical networking domains. TSNBench addresses this gap by introducing MCQA and open-ended WCD computation questions with ground truth verified by state-of-the-art NC solvers, providing an evaluation of TSN that no existing general benchmark captures.

Networking and Telecommunications Benchmarks: In the last few years, several benchmarks have evaluated LLM proficiency in networking and telecommunications domains. TeleQnA [Maatouk et al., 2026] presents an MCQ dataset for telecommunications, generated from research documents and 3GPP standards and validated by domain experts. 6G-Bench [Ferrag et al., 2026] presents an MCQ-based dataset for 6G networks containing 3,722 difficult questions validated through automated filtering and expert human review. Beyond question-answering benchmarks, NetConfEval [Wang et al., 2024a] evaluates LLMs on network configuration tasks and demonstrates that LLMs can simplify and automate complex network management tasks.

LLMs for TSN and Real-Time Networks The application of LLMs to TSN management and orchestration is still at a very early stage, with only limited initial studies available. Windmann et al. [2025] explored the use of LLMs for configuring hybrid 5G/TSN networks by assisting users with manual configuration tasks and suggesting configurations in a 5G-TSN network. However, this work remains preliminary and does not provide experimental results. Overall, prior work does not provide a systematic benchmark or rigorous evaluation of LLM proficiency across TSN mechanisms, nor does it assess computational reasoning capabilities for WCD analysis. TSNBench fills this gap by providing the first structured benchmark covering both declarative TSN knowledge through MCQA and computational reasoning through open-ended WCD evaluation.

3 TSNBench

Unlike established domains such as medicine [Xie et al., 2025], 5G [Oluwaseyi et al., 2025, Maatouk et al., 2026], general human knowledge [Phan et al., 2026, Hendrycks et al., 2021, Wang et al., 2024b], coding [Hua et al., 2025, Huang et al., 2024], and law [Guha et al., 2023], no open-source TSN dataset exists for LLM evaluation [Zhang et al., 2024, Peng et al., 2023, Zanbouri et al., 2025, Adil et al., 2026]. As highlighted in [Liu et al., 2023], the data source determines the reliability of a dataset, and generating a high-quality dataset is a crucial prerequisite for meaningful benchmarking. We describe the TSNBench construction pipeline below, with full details provided in Appendix 11.

3.1 Dataset Source Selection

Published research papers and standards are among the most reliable sources for building domain-specific datasets [Liu et al., 2023]. Since TSN knowledge originates primarily from peer-reviewed

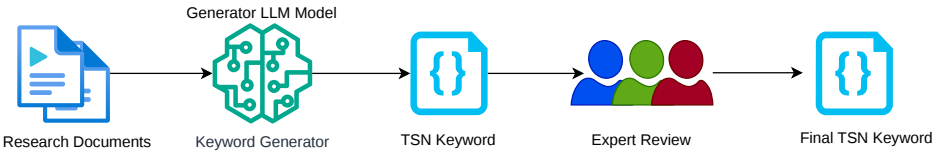

Figure 1: TSNBench keyword-generation pipeline. TSN keywords are extracted from research documents using an LLM, expert-verified, and used for MCQA generation as described in Section 3.3.

Table 1: Models used in the TSNBench keyword extraction and question generation pipeline. All models are used with default settings and last accessed in April 2026. Claude Sonnet 4 serves two distinct roles: keyword extraction and question generation. These roles use identical model configurations but operate on different inputs and prompts. Full dataset generation details are provided in Appendix 11.

Model	API	Model ID	Organization	Usage
Claude Sonnet 4	Anthropic API	claude-sonnet-4-20250514	Anthropic	Keyword extractor
Claude Sonnet 4	Anthropic API	claude-sonnet-4-20250514	Anthropic	Generator
GPT-4o mini	OpenAI API	gpt-4o-mini	OpenAI	Generator
Llama 3.1 70B	HF Router	Llama-3.1-70B-Instruct	Meta	Generator

research and IEEE 802.1 TSN standards, we curate a collection of open-access research documents as our source corpus. To avoid copyright issues and exclude papers with incorrect results or flawed methodologies, we include only published open-access papers. For papers not available in open-access form, we use arXiv versions that have been published or accepted, excluding unpublished preprints with unverified results. Where possible, we also collect author manuscript versions with proper attribution. To ensure quality, we prioritize highly cited papers from reputable venues while accounting for publication timeline, as recent papers naturally have fewer citations. In total, we collect 83 research papers covering a broad range of TSN mechanisms, including Time-Aware Shaper (TAS), CBS, CQF, NC-based schedulability analysis, performance evaluation, hardware experiments, combined shapers such as TAS+CBS [Zhao et al., 2022], and Multi-CQF [Alexandris et al., 2022]. Detailed background on TSN, related work, and its mechanisms is given in Appendix 7.

3.2 Keyword and Acronym Extraction

TSN employs specialized vocabulary, similar to other communication domains [Andrews et al., 2014, Saad et al., 2020, Ma et al., 2019]. A successful LLM that understands TSN should be able to reason correctly about TSN terminology. A model that cannot differentiate between TAS and CBS, or cannot correctly expand TSN-specific acronyms, cannot be considered proficient in TSN. To capture this dimension, we extract keywords and acronyms widely used in TSN literature and use them to guide MCQA generation. All terms are extracted from the 83 research documents using Claude Sonnet 4, as shown in Table 1, and stored in JSON format. Each document is preprocessed to remove non-relevant content, including author names, affiliations, figures, tables, URLs, and pseudocode. The model is instructed to extract only terms defined within the document, without relying on pretrained knowledge, and to provide each term's acronym, full form, and one-to-two-sentence definition from the source. The extracted set is then reviewed by domain experts to resolve duplicates, retaining the longer definition in cases of conflict. Figure 1 illustrates this pipeline.

3.3 MCQA Generation, Post-Processing, and Expert Review

Raw MCQA Generation: To optimize time and reduce manual effort, we use an LLM-based approach to generate MCQAs from research documents. The keyword file is provided alongside the research documents as additional input, serving as an independent source to complement research paper content during generation. We use three models from distinct families, namely Claude Sonnet 4, GPT-4o mini, and Llama 3.1 70B, as shown in Table 1. These models are deliberately selected to ensure diverse styles and reasoning capabilities, thereby reducing generative bias. The same

system prompt is used for all models, and each research paper is assigned to exactly one model in a round-robin manner. Non-relevant sections, such as author information, affiliations, references, URLs, figures, tables, and pseudocode, are removed from each document before generation.

Post-Processing: LLM-generated MCQAs cannot be used directly for benchmarking, as they may contain incorrectly formulated questions, incomplete options, or vague and incorrect answer choices. To address positional bias introduced by the generating model, answer options are shuffled randomly prior to human expert review, with the correct answer label updated to reflect the new ordering.

Human-Based Domain Expert Review: Given the safety-critical nature of TSN, rigorous human validation is essential. We engage five TSN domain experts: three senior professors with more than 15 years of research experience and two postdoctoral researchers with more than 8 years of expertise. Each question is independently evaluated with four outcomes: (i) *accept* - correct and clear; (ii) *revise* - requires modification for clarity or correctness; (iii) *reject* - the question is incorrect, misleading, or irrelevant; or (iv) *doubtful* - the expert is uncertain and passes it to remaining reviewers for consensus. Questions without consensus are discarded. Full review criteria are provided in Appendix 11.1 and Table 5. Table 2 summarizes the dataset statistics and Figure 2 illustrates the full pipeline.

Table 2: TSNBench dataset construction statistics. Full generation details are in Appendix 11.

Type	Category	Count
MCQA	Total raw questions generated by models	1326
	Questions removed after expert review	387
	Questions revised by domain experts	185
	Questions in the final dataset (used for benchmarking)	**939**
Open-ended questions	Credit-Based Shaper (CBS)	100
	Cyclic Queuing and Forwarding (CQF)	100

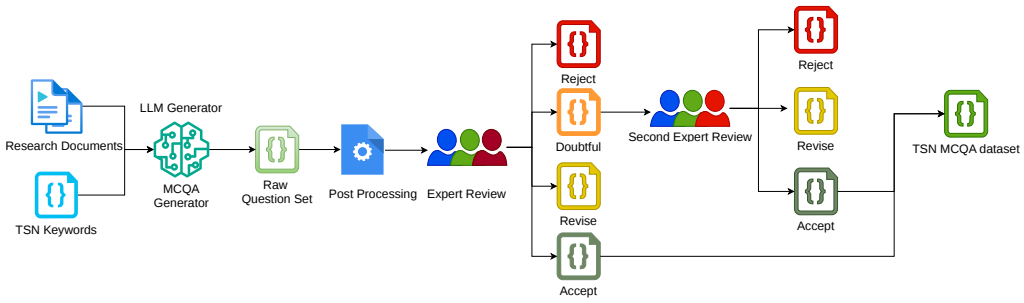

Figure 2: Pipeline of our TSNBench MCQA dataset generator, showing all steps from raw generation to the final validated dataset.

3.4 Open-Ended Question Formulation

While MCQA evaluates declarative TSN knowledge, open-ended questions assess whether LLMs can perform the multi-step mathematical reasoning required in real TSN deployment. We evaluate WCD computation, as WCD is a central key performance indicator (KPI) in TSN network design and directly determines whether a network meets its stringent timing requirements. We select two TSN mechanisms for this evaluation: CBS and CQF. CBS is widely deployed for audio-video traffic and requires NC-based analysis, making it mathematically demanding. CQF is a more recently standardized TSN mechanism whose WCD can be computed from a closed-form equation given routing and cycle duration (T), providing a complementary evaluation that isolates formula application from NC complexity. Together, these two mechanisms span a meaningful range of WCD computation difficulty. Ground truth WCD values are computed using a verified state-of-the-art NC tool [Zhao et al., 2018] for CBS and closed-form mathematical upper bound for CQF. We release all ground truth WCD values alongside the questions to support future open-source community evaluations. Each open-ended question is formulated by domain experts, as shown in Figure 3, and comprises three

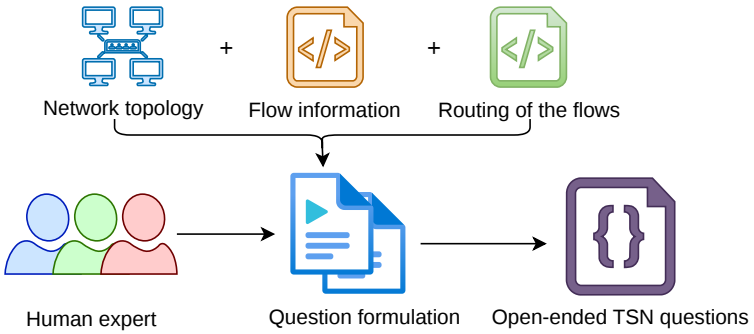

Figure 3: Pipeline for TSNBench open-ended question formulation by domain experts. Each question comprises three components: network topology, flow information, and flow routing.

components: network topology, flow information, and flow routing. In TSNBench, three topologies are used to cover a broad range of scenarios: (i) one-switch topology (Figure 15), (ii) medium-mesh topology (Figure 16), and (iii) ring topology, representing industrial networks (Figure 17). Each topology consists of end nodes and switches connected via Ethernet links, with unicast traffic flows transmitted from a sender to a single receiver. Flows consist of Ethernet frames whose maximum payload is bounded by the Maximum Transmission Unit (MTU). Further topology, flow, and routing details are provided in Appendix 11.4.

3.5 Prompt Design

For both MCQA and open-ended evaluations, each prompt defines the model's role as a TSN expert. For MCQA, we use zero-shot prompting with no in-context examples, representing a conservative approach that measures inherent TSN proficiency, ensuring that the output performance reflects the model's domain knowledge rather than in-context pattern matching. For open-ended questions, we also use a zero-shot setting, providing no example WCD calculations or NC or CQF equations, ensuring the model independently recalls and applies the correct computational methodology. For both question types, the model is asked to provide a confidence score alongside its answer.

The open-ended prompt comprises three variable components: network topology, flow parameters, and pre-computed shortest path routes. The same prompt template is used across all 100 open-ended evaluation instances per mechanism, with only these three components varying. Fixed network constants are maintained throughout to ensure comparability across models and instances. A detailed discussion of the open-ended prompt design is provided in Appendix 11.3.

3.6 Model Scoring and Ground Truth

For the MCQA dataset, performance is measured as the percentage of questions answered correctly, reported as accuracy. For the open-ended questions, we evaluate the computational reasoning capability of each model by comparing its predicted WCD values against ground truth values. For CBS, ground truth WCD values are derived using NC-based Total Flow Analysis (TFA). Specifically, the worst-case delay upper bound D_f^h for flow $f \in \mathcal{F}_{M_i}^h$ at h equals the worst-case delay upper bound $D_{M_i}^h$ for all flows with the same priority M_i aggregating at h,

$$D_f^h = D_{M_i}^h = hDev(\alpha_{M_i}^h, \beta_{M_i}^h) = \sup_{t \geq 0} \left\{ \inf \left\{ \tau \geq 0 \mid \alpha_{M_i}^h(t) \leq \beta_{M_i}^h(t+\tau) \right\} \right\}, \tag{1}$$

where $\alpha_{M_i}^h(t)$ represents the arrival curve of aggregate flows of priority M_i passing through h, and $\beta_{M_i}^h(t)$ represents the service curve for these corresponding flows. The end-to-end WCD for a flow is obtained by summing per-port delay bounds along its route. Full NC methodology and proofs are provided in Appendix 8.

For CQF, the worst-case end-to-end delay is given by the closed-form expression

$$\mathrm{WCD} = f_i.\phi + (\mathrm{SW}_{\mathrm{num}} + 1) \cdot \mathrm{T} + \xi, \tag{2}$$

Figure 4: TSNBench MCQA results across 16 models. Accuracy is the percentage of correct answers out of 939 questions, and consistency measures whether the model gives the same response across three runs. All models are evaluated at temperature 0.0 for deterministic performance; models without temperature support use their default setting and are marked with †. Full model details are given in Table 6, and extended results with temperature comparisons are provided in Table 7 in Appendix 12.

Model	Accuracy (%)	Avg. Consistency	Avg. Latency (ms)	Avg. Conf.	ECE↓	Brier↓	CW Rate↓
Grok 4.1 Fast†	93.2	0.9858	6673	0.9509	0.0151	0.0599	99.0
Grok 4.1 Fast (Non-Reasoning)	91.7	0.9986	515	0.9760	0.0328	0.0764	100.0
DeepSeek-V3.2 (Non-thinking)	94.0	0.9993	804	0.9312	**0.0105**	0.0526	96.4
GPT-4o	91.8	0.9957	729	0.8782	0.0354	0.0765	99.2
GPT-4o mini	88.3	0.9950	799	0.9004	0.0538	0.0974	77.8
Llama 3.3	88.9	0.9950	365	0.9082	0.0450	0.0918	100.0
Mistral Medium 3.1	92.1	0.9965	653	**0.9779**	0.0295	0.0750	100.0
Mistral Large 3	92.8	0.9975	5498	0.9476	0.0214	0.0646	100.0
Claude Sonnet 4.5	**95.3**	0.9993	1842	0.9374	0.0181	**0.0429**	86.6
o3†	94.7	0.9840	3845	**0.7524**	**0.1874**	0.0852	**3.4**
GPT-5†	95.0	0.99	5630	0.8773	0.0569	0.0475	51.7
DeepSeek-V3.2 (Thinking)†	94.7	0.9819	4400	0.9202	0.0224	0.0487	78.1
Gemini 2.5 Flash	90.1	0.9847	6744	0.9674	0.0539	0.0942	95.4
Llama 3.2 1B	**67.4**	1.0	669	0.8529	0.1859	**0.2544**	99.0
Qwen3 8B	83.7	0.9897	**15103**	0.8616	0.0351	0.1322	100.0
Ministral 3 8B	86.9	0.9954	**345**	0.9649	0.0822	0.1230	100.0

† Temperature parameter not supported. Evaluated with default settings. ↓ lower is better.

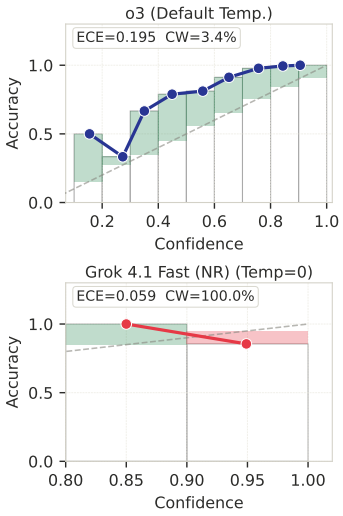

Figure 5: Reliability plot for o3 and Grok 4.1 Fast (NR). Full reliability analysis are in Figure 6.

where $f_i.\phi$ is the flow offset at the source node in μs, $\mathrm{SW}_{\mathrm{num}}$ is the number of switches along the flow route, T is the cycle duration in μs, and ξ denotes the network specific delays including processing delay, propagation delay, switching delay, and time synchronization error. The derivation and proof of this bound are provided in Appendix 10.

4 Experiments

We evaluate 16 state-of-the-art LLMs spanning open-source and closed-source models across general-purpose and reasoning-specialized architectures. Table 6 in Appendix 12 provides the full list of models with their model IDs and organizations. All models are accessed via their respective official vendor APIs with no fine-tuning applied: GPT (OpenAI API), DeepSeek (DeepSeek API), Mistral (Mistral AI API), Claude (Anthropic API), Gemini (Google AI API), Grok (xAI API), and Llama and Qwen (Hugging Face inference router). All client-side operations, including prompt construction, API handling, response parsing, and metric computation, are performed on a standard workstation. To assess repeatability and stochasticity, each MCQA and open-ended question is evaluated three times under two temperature settings: deterministic (T = 0.0) and stochastic (T = 0.7). Since TSN is widely used in safety-critical domains, deterministic responses are essential, as non-determinism would undermine the reliability of LLM-based TSN reasoning. For models that do not expose a temperature parameter, evaluations use the vendor default configuration, as noted in Table 4. Full cost and latency details are provided in Appendix 12, Table 8.

4.1 MCQA Evaluation

Contamination Analysis: Since the MCQs were generated using models from families included in the evaluation, as shown in Table 1, contamination is a potential concern. We therefore separate the evaluated models into generator families (Claude, GPT, Llama) and non-generator families (all remaining models) and compare their average MCQA accuracy. Generator-family models achieve an average accuracy of 88.8%, whereas non-generator-family models achieve 91.0%. The generator-family models do not perform better than the non-generator-family models, so we do not observe evidence of a systematic advantage. This analysis does not rule out all possible contamination pathways, but it addresses this specific concern. The open-ended timing tasks are less likely to be affected because their topology, flow, and routing inputs were constructed specifically for TSNBench.

Evaluation Metrics: Model performance on the MCQA dataset is measured using accuracy, defined as the percentage of correctly answered questions out of 939, averaged across three runs. We additionally report Expected Calibration Error (ECE) [Pavlovic, 2025] and Brier score [Hoessly,

2026] to evaluate the alignment between the model's expressed confidence and its actual correctness. Calibration is particularly critical in safety-critical domains such as TSN, where high-confidence incorrect answers may lead to misleading configuration decisions, deadline violations, or network instability in industrial and automotive systems. We therefore also evaluate the Confidently Wrong (CW) rate to determine the fraction of incorrect answers where the model expresses high confidence ($\geq$0.8). All calibration metrics are computed on the full 939-MCQA dataset across three runs per model.

Results and Discussion: Table 4 reports accuracy, average (avg.) consistency, calibration, and average latency for all 16 models. The top performers are Claude Sonnet 4.5 (95.3%) and GPT-5 (95.0%), with Claude Sonnet 4.5 also achieving the lowest Brier score (0.0429), indicating strong accuracy and calibration. Llama 3.2 1B achieves the lowest accuracy (67.4%), consistent with its substantially smaller parameter count compared with the other models.

A notable finding emerges from the reasoning models. Despite their stronger general reasoning capabilities, o3, GPT-5, and DeepSeek-V3.2 (Thinking) do not outperform the best non-reasoning models on MCQA, all scoring below Claude Sonnet 4.5. This suggests that TSN MCQA performance is primarily driven by domain knowledge rather than general reasoning, and that reasoning-specialized architectures offer limited advantage on declarative knowledge retrieval tasks.

The calibration results reveal key differences across models. While most models are well-calibrated (ECE < 0.06), o3 has the highest ECE (0.1874) despite 94.7% accuracy, yet achieves the lowest CW rate (3.4%), rarely assigning high confidence to incorrect answers (refer to Figure 5). In contrast, many non-reasoning models have CW rates of 100%, assigning high confidence to incorrect answers. Mistral Medium 3.1 has the highest average confidence (0.9779) while maintaining 92.1% accuracy. All models have zero refusal rate, indicating that the MCQA dataset does not trigger response refusals.

4.2 Reliability Analysis

Figure 6 presents the reliability plot for all 16 evaluated models on the MCQA dataset.

Each diagram shows the observed accuracy against the model's expressed confidence, binned across the confidence range. A perfectly calibrated model would fall on the gray dashed diagonal line. This means the model's confidence would perfectly align with its actual accuracy. The red shaded region indicates overconfidence, meaning the model's confidence exceeds its actual accuracy. The green shaded region indicates underconfidence, meaning the model is more accurate than its expressed confidence suggests.

In safety-critical TSN deployments, overconfidence is significantly more dangerous than underconfidence. A model that is incorrect but expresses high confidence may mislead a network engineer with an erroneous WCD estimate or misconfigured scheduling parameters. By contrast, an underconfident model that expresses uncertainty on correct answers prompts additional verification.

The majority of the evaluated models sit in the high-confidence region (0.8 to 1.0) regardless of their actual accuracy. This indicates that the models tend to exhibit overconfidence.

Grok 4.1 Fast (NR), Mistral Medium 3.1, Mistral Large 3, and Ministral 3 8B achieve CW rates of 100%, meaning all incorrect answers fall in the high-confidence range. This represents the most critical calibration behavior for TSN deployment. GPT-4o, Gemini 2.5 Flash, Llama 3.2 1B, and Qwen3 8B similarly exhibit CW rates exceeding 95%. A notable exception is o3, which is the only model that falls predominantly in the green underconfident zone, with a CW rate of just 3.4%. Despite having the highest ECE (0.1874) among all evaluated models, o3 is the safest among the evaluated models from a calibration perspective, as it rarely expresses high confidence on incorrect MCQA answers. This highlights an important distinction between aggregate calibration metrics and safety-relevant calibration behavior. DeepSeek-V3.2 (NT) achieves the lowest ECE (0.0105), suggesting strong overall calibration, yet maintains a CW rate of 96.4%, demonstrating that a low ECE does not guarantee safe and realistic confidence behavior.

4.3 Open-Ended Question Evaluation

Evaluation Metrics: For the open-ended questions, we report two widely used metrics: Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE), computed per test case (TC).

Figure 6: Reliability diagram representing the performance of all 16 state-of-the-art models evaluated on the MCQA dataset in TSNBench. The gray dashed line represents the perfect calibration where confidence is equal to the accuracy. A model which is 100% confident and has 100% accuracy will fall on this gray dashed line. The red shaded region represents the over-confidence of the model (model confidence exceeds the actual accuracy of the model), and the green shaded region represents the under-confidence of the model (actual accuracy of the model exceeds it's given confidence).

Each TC consists of n flows, denoted f_i where $i = 1 \cdots n$. For each flow f_i, $\hat{y}_{TC_x,f_i}$ denotes the WCD predicted by the model for TC_x and y_{TC_x,f_i} denotes the ground truth WCD of flow f_i for TC number x, computed using a verified NC solver for CBS and using Eq. 22 for CQF. The MAE for each TC is defined as:

$$\text{MAE}_{TC_x} = \frac{1}{n}\sum_{i=1}^{n} |\hat{y}_{TC_x,f_i} - y_{TC_x,f_i}|, \tag{3}$$

where x denotes the TC index and $x \in \{1, \ldots, 100\}$. The MAPE for each TC is defined as:

$$\text{MAPE}_{TC_x} = \frac{1}{n}\sum_{i=1}^{n} \frac{|\hat{y}_{TC_x,f_i} - y_{TC_x,f_i}|}{y_{TC_x,f_i}} \times 100 \tag{4}$$

The overall MAE and MAPE for a model are obtained by averaging across all 100 TCs:

$$\text{MAE} = \frac{1}{100}\sum_{x=1}^{100}\text{MAE}_{TC_x}, \qquad \text{MAPE} = \frac{1}{100}\sum_{x=1}^{100}\text{MAPE}_{TC_x} \tag{5}$$

Table 3: Open-ended WCD estimation results for CBS and CQF across 100 test cases (TCs). MAE and MAPE are reported as mean ± standard deviation across all TCs. Median MAE is a robust measure against outlier TCs. A model is excluded ("–") if: (i) it responded to fewer than 50 TCs, (ii) fewer than 80% of flows per TC received a WCD estimate, or (iii) all predicted WCD values were zero (trivial failure).

Model	CBS WCD Accuracy			CQF WCD Accuracy		
	MAE (µs) ↓	MAPE (%) ↓	Median (µs) ↓	MAE (µs) ↓	MAPE (%) ↓	Median (µs) ↓
Grok 4.1 Fast†	174.6 ± 314.5	127.9 ± 514.1	107.0	139.6 ± 90.0	83.2 ± 56.6	137.7
Grok 4.1 Fast (Non-Reasoning)	3246.3 ± 3762.8	1102.5 ± 1112.9	**2185.4**	168.3 ± 69.0	90.6 ± 23.2	167.7
DeepSeek-V3.2 (Non-thinking)	–	–	–	172.3 ± 72.8	94.1 ± 40.4	178.2
GPT-4o⋆	–	–	–	82.2 ± 264.7	61.9 ± 193.6	**1.2**
GPT-4o mini	378.5 ± 189.7	97.2 ± 13.9	337.7	180.5 ± 82.0	99.2 ± 44.8	175.3
Llama 3.3 70B	313.3 ± 174.0	84.2 ± 39.0	273.3	160.9 ± 83.7	99.0 ± 77.2	147.0
Mistral Medium 3.1	337.4 ± 225.7	102.7 ± 93.8	258.0	166.8 ± 92.6	96.2 ± 82.6	141.0
Mistral Large 3	240.1 ± 152.3	62.7 ± 27.5	205.2	$\mathbf{59.5 \pm 27.2}$	$\mathbf{41.8 \pm 27.1}$	50.0
Claude Sonnet 4.5	292.8 ± 173.9	71.7 ± 15.3	264.4	211.5 ± 1057.3	116.2 ± 607.6	60.7
o3†	262.5 ± 319.4	84.4 ± 106.0	142.4	102.2 ± 76.0	60.4 ± 46.0	81.1
GPT-5†	$\mathbf{150.2 \pm 198.2}$	$\mathbf{36.2 \pm 36.4}$	**92.4**	107.0 ± 69.0	62.4 ± 42.1	107.0
DeepSeek-V3.2 (Thinking)†	–	–	–	–	–	–
Gemini 2.5 Flash	552.7 ± 1821.0	277.8 ± 1417.7	225.6	112.0 ± 89.9	60.6 ± 46.5	92.5
Llama 3.2 1B	–	–	–	–	–	–
Qwen3 8B§	–	–	–	–	–	–
Ministral 3 8B	$\mathbf{70287.8 \pm 403636.4}$	$\mathbf{25498.1 \pm 164932.0}$	879.1	$\mathbf{2918.5 \pm 4017.6}$	$\mathbf{1705.5 \pm 2382.1}$	**1046.0**

† Temperature parameter not supported. Evaluated with default settings.
⋆ GPT-4o returned all-zero WCD values for all CBS test cases (trivial failure) but produced efficient WCD response for CQF.
§ **Qwen3 8B** evaluation failed due to repeated API timeout errors. No valid responses recorded for any TC. **Llama 3.2 1B** provided WCDs for fewer than 5 TCs and furthermore provided insufficient valid response for both CBS and CQF. **DeepSeek-V3.2 (Thinking)** provided empty response for all TCs. lower ↓ is better for MAE, MAPE, and Median.

We additionally report the median MAE across TCs as a robust measure against outlier TCs. Further example and details on the evaluation metrics are provided in Appendix 13 and Table 9.

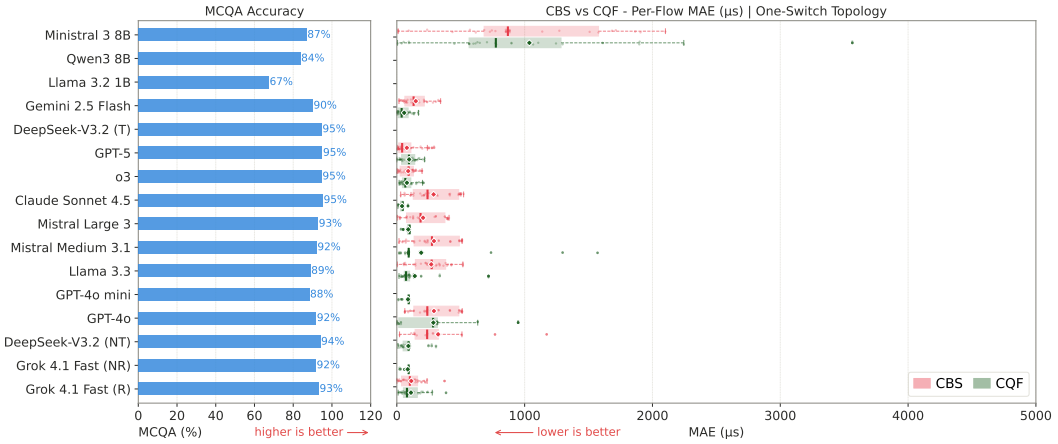

Figure 7: Performance comparison across MCQA and open-ended WCD computation for all 16 evaluated models in TSNBench. (*Left*) MCQA accuracy (%) per model. (*Right*) Per-TC MAE distribution (in μs) for CBS and CQF open-ended questions, shown as box plots over **One-Switch topology** test cases.

Results and Discussion: Table 3 presents the WCD computation results for both CBS and CQF across all 100 TCs. The central finding is a striking dissociation between MCQA accuracy and computational reasoning performance. Models that achieve above 90% accuracy on MCQA still fail substantially on open-ended WCD computation, with the best-performing model, GPT-5, achieving a median MAE of 92.4 μs on CBS, which is concerning because industrial TSN traffic can have strict timing requirements [Ekrad et al., 2025]. Detailed per-TC results are provided in Appendix 13.

For CBS, most models produce large errors, with many exceeding 200 μs MAE and 70% MAPE. Several models exhibit distinct failure modes. On CBS, Llama 3.2 1B responds to fewer than 50 evaluated TCs, returning all-zero WCD values for few TCs and partially incorrect values for some

TCs, with incomplete flow coverage in all responses. Grok 4.1 Fast (Reasoning) returns truncated JSON, providing flow profile metadata but no WCD values, suggesting that the model hit an output length limit. DeepSeek-V3.2 (Thinking) returns empty responses for more than 70 TCs across both mechanisms. The NC-based computation required for CBS is mathematically demanding and complex, and the zero-shot setting reveals that most models cannot independently recall or correctly apply the full NC methodology. Among models that produce valid CBS responses, GPT-5 achieves the best performance (MAE 150.2 μs, MAPE 36.2%). Notably, OpenAI reasoning models and Grok 4.1 Fast perform better on CBS than non-reasoning models, with GPT-5 achieving substantially lower MAE than all non-reasoning models, suggesting that multi-step mathematical reasoning capability provides an advantage for NC-based WCD computation even when it does not improve MCQA accuracy.

For CQF, performance is more varied, with median MAE ranging from 1.2 μs (GPT-4o) to 1,046 μs (Ministral 3 8B), and MAPE ranging from 41.8% (Mistral Large 3) to 1705.5% (Ministral 3 8B). GPT-4o achieves the lowest median MAE on CQF (1.2 μs, MAPE 61.9%) despite failing completely on CBS, suggesting it can correctly apply the CQF closed-form equation. Mistral Large 3 achieves the lowest MAPE on CQF (41.8%), indicating the most accurate relative WCD estimation across all evaluated models. Llama 3.2 1B exhibits the most severe hallucination failure, fabricating up to 1,013 flows (flow 0–1012) instead of predicting WCD for the actual flows (fewer than 30 flows per TC), and returning WCD = 0 for all. Qwen3 8B fails to produce any response for either CBS or CQF due to repeated API timeouts. Ministral 3 8B, despite being a small model, produces valid responses for both CBS and CQF but with large errors (MAPE 25498.1% for CBS and 1705.5% for CQF), demonstrating that context handling is necessary but not sufficient for correct WCD computation.

Comparison across MCQA and open-ended questions: Figure 7 illustrates the performance differences between models across two evaluation types, MCQA and open-ended questions. The right-hand figure shows the MAE for a one-switch topology across different models, while the left-hand figure presents the MCQA accuracy. The MCQA accuracy remains high, above 80%, for all models except Llama 3.2 1B. However, the MAE is still significant for TSN flows with deadlines in the range of 1000 to 5000 μs. Figure 18 further presents the performance differences between models for MCQA and open-ended questions in a ring topology.

5 Conclusion

We present TSNBench, the first benchmark for evaluating LLM proficiency in Time-Sensitive Networking (TSN), comprising 939 expert-validated multiple-choice questions (MCQs) and 100 open-ended questions per mechanism for Credit-Based Shaper (CBS) and Cyclic Queuing and Forwarding (CQF). The ground truth WCD values are computed using a verified Network Calculus (NC) solver for CBS and closed-form mathematical upper bounds for CQF. We evaluate 16 LLMs and find that models achieve 67-95% MCQA accuracy yet fail substantially on open-ended WCD computation, with the best model (GPT-5) still achieving a Mean Absolute Percentage Error (MAPE) of 36.2% on CBS. Despite CBS being extensively researched and an older mechanism, models cannot correctly apply NC, whereas CQF, with its simpler closed-form equation, is handled more successfully, confirming that WCD computation performance is governed by mathematical complexity rather than mechanism maturity. TSNBench demonstrates that MCQ benchmarks substantially overestimate LLM capability in safety-critical domains.

Limitations and Future Directions: TSNBench has three primary limitations. First, the MCQA dataset is generated from open-access research papers, limiting coverage of certain mechanisms. Second, the open-ended evaluation covers only CBS and CQF. Extending to TAS is a natural next step, though its NP-hard gate control list (GCL) synthesis problem poses additional challenges beyond CBS and CQF. Third, the open-ended tasks evaluate standalone zero-shot model behavior under a closed-book prompt and should not be interpreted as a recommended deployment workflow for safety-critical TSN systems. Evaluating LLMs in settings where they produce checkable artifacts verified by deterministic analysis tools, and evaluating whether providing NC equations in the prompt improves WCD computation accuracy, are important directions for future versions of TSNBench.

References

IEEE Standard for Local and Metropolitan Area Networks - Virtual Bridged Local Area Networks Amendment 12:

Forwarding and Queuing Enhancements for Time-Sensitive Streams. *IEEE Std 802.1Qav-2009 (Amendment to IEEE Std 802.1Q-2005)*, pages C1–72, 2010. doi: 10.1109/IEEESTD.2009.5375704.

IEEE Standard for Local and metropolitan area networks–Bridges and Bridged Networks–Amendment 29: Cyclic Queuing and Forwarding. *IEEE 802.1Qch-2017 (Amendment to IEEE Std 802.1Q-2014 as amended by IEEE Std 802.1Qca-2015, IEEE Std 802.1Qcd(TM)-2015, IEEE Std 802.1Q-2014/Cor 1-2015, IEEE Std 802.1Qbv-2015, IEEE Std 802.1Qbu-2016, IEEE Std 802.1Qbz-2016, and IEEE Std 802.1Qci-2017)*, pages 1–30, 2017. doi: 10.1109/IEEESTD.2017.7961303.

IEEE Standard for Local and Metropolitan Area Network–Bridges and Bridged Networks. *IEEE Std 802.1Q-2018 (Revision of IEEE Std 802.1Q-2014)*, pages 1–1993, 2018. doi: 10.1109/IEEESTD.2018.8403927.

A Ademaj, D Puffer, D Bruckner, G Ditzel, L Leurs, MP Stanica, P Didier, R Hummen, R Blair, and T Enzinger. Industrial automation traffic types and their mapping to QoS/TSN mechanisms. *TSN mechanisms*, 3, 2019.

Muhammad Adil, Tie Qiu, Xiaobo Zhou, Danish Javeed, Zhenrui Cao, and Dapeng Oliver Wu. Integrated 5G and Time Sensitive Networking for Emerging Applications: A Survey of Advancements, Challenges, and Future Directions. *IEEE Communications Surveys & Tutorials*, 28:4016–4050, 2026. doi: 10.1109/COMST.2025.3632286.

Konstantinos Alexandris, Paul Pop, and Tongtong Wang. Configuration and Evaluation of Multi-CQF Shapers in IEEE 802.1 Time-Sensitive Networking (TSN). *IEEE Access*, 10:109068–109081, 2022. doi: 10.1109/ACCESS.2022.3214007.

Jeffrey G. Andrews, Stefano Buzzi, Wan Choi, Stephen V. Hanly, Angel Lozano, Anthony C. K. Soong, and Jianzhong Charlie Zhang. What will 5g be? *IEEE Journal on Selected Areas in Communications*, 32(6):1065–1082, 2014. doi: 10.1109/JSAC.2014.2328098.

Dietmar Bruckner, Marius-Petru Stănică, Richard Blair, Sebastian Schriegel, Stephan Kehrer, Maik Seewald, and Thilo Sauter. An introduction to OPC UA TSN for industrial communication systems. *Proceedings of the IEEE*, 107(6):1121–1131, 2019. doi: 10.1109/JPROC.2018.2888703.

Daniel Bujosa, Mohammad Ashjaei, Alessandro V Papadopoulos, Thomas Nolte, and Julián Proenza. HERMES: Heuristic multi-queue scheduler for TSN time-triggered traffic with zero reception jitter capabilities. In *Proc. RTNS*, 2022. doi: 10.1145/3534879.3534906.

Daniel Bujosa Mateu. *Improved Configuration and Analysis Solutions for Time-Sensitive Networks with Support for Legacy Systems*. Malardalen University (Sweden), 2024.

Martin Böhm and Diederich Wermser. Multi-domain time-sensitive networks—control plane mechanisms for dynamic inter-domain stream configuration. *Electronics*, 10(20), 2021. ISSN 2079-9292. doi: 10.3390/electronics10202477.

Silviu S. Craciunas, Ramon Serna Oliver, Martin Chmelík, and Wilfried Steiner. Scheduling Real-Time Communication in IEEE 802.1Qbv Time Sensitive Networks. In *Proceedings of the 24th International Conference on Real-Time Networks and Systems*, RTNS '16, page 183–192, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450347877. doi: 10.1145/2997465.2997470. URL `https://doi.org/10.1145/2997465.2997470`.

Rubi Debnath, Mustafa Selman Akinci, Devika Ajith, and Sebastian Steinhorst. 5GTQ: QoS-Aware 5G-TSN Simulation Framework. In *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*, pages 1–7, 2023a. doi: 10.1109/VTC2023-Fall60731.2023.10333533.

Rubi Debnath, Philipp Hortig, Luxi Zhao, and Sebastian Steinhorst. Advanced Modeling and Analysis of Individual and Combined TSN Shapers in OMNeT++. In *2023 IEEE 29th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*, pages 176–185, 2023b. doi: 10.1109/RTCSA58653.2023.00029.

Rubi Debnath, Philipp Hortig, Luxi Zhao, and Sebastian Steinhorst. Quantifying the Impact of Frame Preemption on Combined TSN Shapers. In *NOMS 2024-2024 IEEE Network Operations and Management Symposium*, pages 1–9, 2024. doi: 10.1109/NOMS59830.2024.10575564.

Rubi Debnath, Mohammadreza Barzegaran, and Sebastian Steinhorst. Toward an optimized multi-cyclic queuing and forwarding in time-sensitive networking with time injection. *IEEE Internet of Things Journal*, 12(20):43034–43051, 2025a. doi: 10.1109/JIOT.2025.3597560.

Rubi Debnath, Luxi Zhao, Mohammadreza Barzegaran, and Sebastian Steinhorst. CyclicSim: Comprehensive Evaluation of Cyclic Shapers in Time-Sensitive Networking. In *2025 IEEE 22nd Consumer Communications & Networking Conference (CCNC)*, pages 01–09, 2025b. doi: 10.1109/CCNC54725.2025.10975975.

Rubi Debnath, Luxi Zhao, and Sebastian Steinhorst. Learning-Based Traffic Classification for Mixed-Critical Flows in Time-Sensitive Networking. In *ICC 2025 - IEEE International Conference on Communications*, pages 5926–5932, 2025c. doi: 10.1109/ICC52391.2025.11161468.

Kasra Ekrad, Inés Alvarez Vadillo, Bjarne Johansson, Saad Mubeen, and Mohammad Ashjaei. A Methodology to Map Industrial Automation Traffic to TSN Traffic Classes. In *2025 IEEE 30th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–8, 2025. doi: 10.1109/ETFA65518.2025.11205571.

Leonard Elliott. Time-sensitive networking (tsn) in military ground vehicle architectures. In *2024 NDIA Michigan Chapter Ground Vehicle Systems Engineering and Technology Symposium*. National Defense Industrial Association Michigan Chapter, August 2023. doi: https://doi.org/10.4271/2024-01-4122. URL `https://doi.org/10.4271/2024-01-4122`.

Mohamed Amine Ferrag, Abderrahmane Lakas, and Mérouane Debbah. 6G-Bench: An Open Benchmark for Semantic Communication and Network-Level Reasoning With Foundation Models in AI-Native 6G Networks. *IEEE Open Journal of the Communications Society*, 7:3305–3330, 2026. doi: 10.1109/OJCOMS.2026.3680457.

Norman Finn. Introduction to Time-Sensitive Networking. *IEEE Communications Standards Magazine*, 2(2): 22–28, 2018. doi: 10.1109/MCOMSTD.2018.1700076.

Tiziana Fiori, Francesco Giacinto Lavacca, Francesco Valente, and Vincenzo Eramo. Proposal and Investigation of a Lite Time Sensitive Networking Solution for the Support of Real Time Services in Space Launcher Networks. *IEEE Access*, 12:10664–10680, 2024. doi: 10.1109/ACCESS.2024.3353466.

Pranshav Gajjar, Cong Shen, and Vijay K Shah. Tele-LLM-hub: Building context-aware multi-agent LLM systems for telecom networks. In *NeurIPS 2025 Workshop: AI and ML for Next-Generation Wireless Communications and Networking*, 2025. URL `https://openreview.net/forum?id=AencYkmJtl`.

Voica Gavriluţ and Paul Pop. Traffic-type Assignment for TSN-based Mixed-criticality Cyber-physical Systems. 4(2), January 2020. ISSN 2378-962X. doi: 10.1145/3371708. URL `https://doi.org/10.1145/3371708`.

Voica Gavriluţ, Luxi Zhao, Michael L. Raagaard, and Paul Pop. Avb-aware routing and scheduling of time-triggered traffic for tsn. *IEEE Access*, 6:75229–75243, 2018. doi: 10.1109/ACCESS.2018.2883644.

Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Aditya K, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory Dickinson, Haggai Porat, Jason Hegland, Jessica Wu, Joe Nudell, Joel Niklaus, John Nay, Jonathan Choi, Kevin Tobia, Margaret Hagan, Megan Ma, Michael Livermore, Nikon Rasumov-Rahe, Nils Holzenberger, Noam Kolt, Peter Henderson, Sean Rehaag, Sharad Goel, Shang Gao, Spencer Williams, Sunny Gandhi, Tom Zur, Varun Iyer, and Zehua Li. LegalBench: A Collaboratively Built Benchmark for Measuring Legal Reasoning in Large Language Models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 44123–44279. Curran Associates, Inc., 2023. URL `https://proceedings.neurips.cc/paper_files/paper/2023/file/89e44582fd28ddfea1ea4dcb0ebbf4b0-Paper-Datasets_and_Benchmarks.pdf`.

Xingang Guo, Yaxin Li, XiangYi Kong, YILAN JIANG, Xiayu Zhao, Zhihua Gong, Yufan Zhang, Daixuan Li, Tianle Sang, Beixiao Zhu, Gregory Jun, Yingbing Huang, Yiqi Liu, Yuqi Xue, Rahul Dev Kundu, Qi Jian Lim, Yizhou Zhao, Luke Alexander Granger, Mohamed Badr Younis, Darioush Keivan, Nippun Sabharwal, Shreyanka Sinha, Prakhar Agarwal, Kojo Vandyck, Hanlin Mai, Zichen Wang, Aditya Venkatesh, Ayush Barik, Jiankun Yang, Chongying Yue, Jingjie He, Libin Wang, Licheng Xu, Hao Chen, Jinwen Wang, Liujun Xu, Rushabh Shetty, Ziheng Guo, Dahui Song, Manvi Jha, Weijie Liang, Weiman Yan, Bryan Zhang, Sahil Bhandary Karnoor, Jialiang Zhang, Rutva Pandya, Xinyi Gong, Mithesh Ballae Ganesh, Feize Shi, Ruiling Xu, Yifan Zhang, Yanfeng Ouyang, Lianhui Qin, Elyse Rosenbaum, Corey Snyder, Peter Seiler, Geir Dullerud, Xiaojia Shelly Zhang, Zuofu Cheng, Pavan Kumar Hanumolu, Jian Huang, Mayank Kulkarni, Mahdi Namazifar, Huan Zhang, and Bin Hu. Toward engineering AGI: Benchmarking the engineering design capabilities of LLMs. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025. URL `https://openreview.net/forum?id=Wmsnx7EPel`.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=d7KBjmI3GmQ`.

Linard Hoessly. On misconceptions about the brier score in binary prediction models. *Glob. Epidemiol.*, 11 (100242):100242, June 2026.

Tianyu Hua, Harper Hua, Violet Xiang, Benjamin Klieger, Sang T. Truong, Weixin Liang, Fan-Yun Sun, and Nick Haber. Researchcodebench: Benchmarking LLMs on implementing novel machine learning research code. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025. URL https://openreview.net/forum?id=3k70Vt0YFS.

Dong Huang, Yuhao Qing, Weiyi Shang, Heming Cui, and Jie M. Zhang. EffiBench: Benchmarking the Efficiency of Automatically Generated Code. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 11506–11544. Curran Associates, Inc., 2024. doi: 10.52202/079017-0367. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/15807b6e09d691fe5e96cdecde6d7b80-Paper-Datasets_and_Benchmarks_Track.pdf.

Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. Benchmarking large language models as AI research agents. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023. URL https://openreview.net/forum?id=kXlTY0BmK3.

Jason J Jackson, Terry Huang, Henry Velasquez, Kevin Zhu, and Sunishchal Dev. Predicting Emergent Software Engineering Capabilities by Fine-tuning. In *NeurIPS 2025 Workshop on Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling*, 2025. URL https://openreview.net/forum?id=EwchHtwavV.

Sunjun Kweon, Jiyoun Kim, Heeyoung Kwak, Dongchul Cha, Hangyul Yoon, Kwanghyun Kim, Jeewon Yang, Seunghyun Won, and Edward Choi. EHRNoteQA: An LLM Benchmark for Real-World Clinical Practice Using Discharge Summaries. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 124575–124611. Curran Associates, Inc., 2024. doi: 10.52202/079017-3958. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/e15c4afff22f12c4986c1fcb4e941e03-Paper-Datasets_and_Benchmarks_Track.pdf.

Shuyue Stella Li, Vidhisha Balachandran, Shangbin Feng, Jonathan S. Ilgen, Emma Pierson, Pang Wei Koh, and Yulia Tsvetkov. MediQ: Question-Asking LLMs and a Benchmark for Reliable Interactive Clinical Reasoning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 28858–28888. Curran Associates, Inc., 2024. doi: 10.52202/079017-0908. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/32b80425554e081204e5988ab1c97e9a-Paper-Conference.pdf.

Junling Liu, Peilin Zhou, Yining Hua, Dading Chong, Zhongyu Tian, Andrew Liu, Helin Wang, Chenyu You, Zhenhua Guo, LEI ZHU, and Michael Lingzhi Li. Benchmarking Large Language Models on CMExam - A comprehensive Chinese Medical Exam Dataset. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 52430–52452. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a48ad12d588c597f4725a8b84af647b5-Paper-Datasets_and_Benchmarks.pdf.

Yongsen Ma, Gang Zhou, and Shuangquan Wang. WiFi Sensing with Channel State Information: A Survey. *ACM Comput. Surv.*, 52(3), June 2019. ISSN 0360-0300. doi: 10.1145/3310194. URL https://doi.org/10.1145/3310194.

Ali Maatouk, Fadhel Ayed, Nicola Piovesan, Antonio De Domenico, Merouane Debbah, and Zhi-Quan Luo. Teleqna: A benchmark dataset to assess large language models telecommunications knowledge. *IEEE Network*, 40(2):253–260, 2026. doi: 10.1109/MNET.2025.3576035.

Ahmed Nasrallah, Akhilesh S. Thyagaturu, Ziyad Alharbi, Cuixiang Wang, Xing Shao, Martin Reisslein, and Hesham Elbakoury. Performance Comparison of IEEE 802.1 TSN Time Aware Shaper (TAS) and Asynchronous Traffic Shaper (ATS). *IEEE Access*, 7:44165–44181, 2019. doi: 10.1109/ACCESS.2019.2908613.

Giwa Oluwaseyi, Michael Adewole, Tobi Awodumila, and Pelumi Aderinto. The LLM as a network operator: A vision for generative AI in the 6g radio access network. In *NeurIPS 2025 Workshop: AI and ML for Next-Generation Wireless Communications and Networking*, 2025. URL https://openreview.net/forum?id=81mgAfsFJv.

Maja Pavlovic. Understanding Model Calibration - A gentle introduction and visual exploration of calibration and the expected calibration error (ECE). In *The Fourth Blogpost Track at ICLR 2025*, 2025. URL https://openreview.net/forum?id=BxBeCjQd2y.

Yifei Peng, Boxin Shi, Tigang Jiang, Xiaodong Tu, Du Xu, and Kun Hua. A Survey on In-Vehicle Time-Sensitive Networking. *IEEE Internet of Things Journal*, 10(16):14375–14396, 2023. doi: 10.1109/JIOT.2023.3264909.

Long Phan, Alice Gatti, Nathaniel Li, Adam Khoja, Ryan Kim, Richard Ren, Jason Hausenloy, Oliver Zhang, Mantas Mazeika, Dan Hendrycks, Ziwen Han, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, Michael Choi, Anish Agrawal, Arnav Chopra, Aakaash Nattanmai, Gordon McKellips, Anish Cheraku, Asim Suhail, Ethan Luo, Marvin Deng, Jason Luo, Ashley Zhang, Kavin Jindel, Jay Paek, Kasper Halevy, Allen Baranov, Michael Liu, Advaith Avadhanam, David Zhang, Vincent Cheng, Brad Ma, Evan Fu, Liam Do, Joshua Lass, Hubert Yang, Surya Sunkari, Vishruth Bharath, Violet Ai, James Leung, Rishit Agrawal, Alan Zhou, Kevin Chen, Tejas Kalpathi, Ziqi Xu, Gavin Wang, Tyler Xiao, Erik Maung, Sam Lee, Ryan Yang, Roy Yue, Ben Zhao, Julia Yoon, Xiangwan Sun, Aryan Singh, Clark Peng, Tyler Osbey, Taozhi Wang, Daryl Echeazu, Timothy Wu, Spandan Patel, Vidhi Kulkarni, Vijaykaarti Sundarapandiyan, Andrew Le, Zafir Nasim, Srikar Yalam, Ritesh Kasamsetty, Soham Samal, David Sun, Nihar Shah, Abhijeet Saha, Alex Zhang, Leon Nguyen, Laasya Nagumalli, Kaixin Wang, Aidan Wu, Anwith Telluri, Summer Yue, Alexandr Wang, Dmitry Dodonov, Tung Nguyen, Jaeho Lee, Daron Anderson, Mikhail Doroshenko, Alun Cennyth Stokes, Mobeen Mahmood, Oleksandr Pokutnyi, Oleg Iskra, Jessica P. Wang, John-Clark Levin, Mstyslav Kazakov, Fiona Feng, Steven Y. Feng, Haoran Zhao, Michael Yu, Varun Gangal, Chelsea Zou, Zihan Wang, Serguei Popov, Robert Gerbicz, Geoff Galgon, Johannes Schmitt, Will Yeadon, Yongki Lee, Scott Sauers, Alvaro Sanchez, Fabian Giska, Marc Roth, Søren Riis, Saiteja Utpala, Noah Burns, Gashaw M. Goshu, Mohinder Maheshbhai Naiya, Chidozie Agu, Zachary Giboney, Antrell Cheatom, Francesco Fournier-Facio, Sarah-Jane Crowson, Lennart Finke, Zerui Cheng, Jennifer Zampese, Ryan G. Hoerr, Mark Nandor, Hyunwoo Park, Tim Gehrunger, Jiaqi Cai, Ben McCarty, Alexis C. Garretson, Edwin Taylor, Damien Sileo, Qiuyu Ren, Usman Qazi, Lianghui Li, Jungbae Nam, John B. Wydallis, Pavel Arkhipov, Jack Wei Lun Shi, Aras Bacho, Chris G. Willcocks, Hangrui Cao, Sumeet Motwani, Emily de Oliveira Santos, Johannes Veith, Edward Vendrow, Doru Cojoc, Kengo Zenitani, Joshua Robinson, Longke Tang, Yuqi Li, Joshua Vendrow, Natanael Wildner Fraga, Vladyslav Kuchkin, Andrey Pupasov Maksimov, Pierre Marion, Denis Efremov, Jayson Lynch, Kaiqu Liang, Aleksandar Mikov, Andrew Gritsevskiy, Julien Guillod, Gözdenur Demir, Dakotah Martinez, Ben Pageler, Kevin Zhou, Saeed Soori, Ori Press, Henry Tang, Paolo Rissone, Sean R. Green, Lina Brüssel, Moon Twayana, Aymeric Dieuleveut, Joseph Marvin Imperial, Ameya Prabhu, Jinzhou Yang, Nick Crispino, Arun Rao, Dimitri Zvonkine, Gabriel Loiseau, Mikhail Kalinin, Marco Lukas, Ciprian Manolescu, Nate Stambaugh, Subrata Mishra, Tad Hogg, Carlo Bosio, Brian P. Coppola, Julian Salazar, Jaehyeok Jin, Rafael Sayous, Stefan Ivanov, Philippe Schwaller, Shaipranesh Senthilkumar, Andres M. Bran, Andres Algaba, Kelsey Van den Houte, Lynn Van Der Sypt, Brecht Verbeken, David Noever, Alexei Kopylov, Benjamin Myklebust, Bikun Li, Lisa Schut, Evgenii Zheltonozhskii, Qiaochu Yuan, Derek Lim, Richard Stanley, Tong Yang, John Maar, Julian Wykowski, Mart Oller, Anmol Sahu, Cesare Giulio Ardito, Yuzheng Hu, Ariel Ghislain Kemogne Kamdoum, Alvin Jin, Tobias Garcia Vilchis, Yuexuan Zu, Martin Lackner, James Koppel, Gongbo Sun, Daniil S. Antonenko, Steffi Chern, Bingchen Zhao, Pierrot Arsene, Joseph M. Cavanagh, Daofeng Li, Jiawei Shen, Donato Crisostomi, Wenjin Zhang, Ali Dehghan, Sergey Ivanov, David Perrella, Nurdin Kaparov, Allen Zang, Ilia Sucholutsky, Arina Kharlamova, Daniil Orel, Vladislav Poritski, Shalev Ben-David, Zachary Berger, Parker Whitfill, Michael Foster, Daniel Munro, Linh Ho, Shankar Sivarajan, Dan Bar Hava, Aleksey Kuchkin, David Holmes, Alexandra Rodriguez-Romero, Frank Sommerhage, Anji Zhang, Richard Moat, Keith Schneider, Zakayo Kazibwe, Don Clarke, Dae Hyun Kim, Felipe Meneguitti Dias, Sara Fish, Veit Elser, Tobias Kreiman, Victor Efren Guadarrama Vilchis, Immo Klose, Ujjwala Anantheswaran, Adam Zweiger, Kaivalya Rawal, Jeffery Li, Jeremy Nguyen, Nicolas Daans, Haline Heidinger, Maksim Radionov, Václav Rozhoň, Vincent Ginis, Christian Stump, Niv Cohen, Rafał Poświata, Josef Tkadlec, Alan Goldfarb, Chenguang Wang, Piotr Padlewski, Stanislaw Barzowski, Kyle Montgomery, Ryan Stendall, Jamie Tucker-Foltz, Jack Stade, T. Ryan Rogers, Tom Goertzen, Declan Grabb, Abhishek Shukla, Alan Givré, John Arnold Ambay, Archan Sen, Muhammad Fayez Aziz, Mark H. Inlow, Hao He, Ling Zhang, Younesse Kaddar, Ivar Ängquist, Yanxu Chen, Harrison K. Wang, Kalyan Ramakrishnan, Elliott Thornley, Antonio Terpin, Hailey Schoelkopf, Eric Zheng, Avishy Carmi, Ethan D. L. Brown, Kelin Zhu, Max Bartolo, Richard Wheeler, Martin Stehberger, Peter Bradshaw, JP Heimonen, Kaustubh Sridhar, Ido Akov, Jennifer Sandlin, Yury Makarychev, Joanna Tam, Hieu Hoang, David M. Cunningham, Vladimir Goryachev, Demosthenes Patramanis, Michael Krause, Andrew Redenti, David Aldous, Jesyin Lai, Shannon Coleman, Jiangnan Xu, Sangwon Lee, Ilias Magoulas, Sandy Zhao, Ning Tang, Michael K. Cohen, Orr Paradise, Jan Hendrik Kirchner, Maksym Ovchynnikov, Jason O. Matos, Adithya Shenoy, Michael Wang, Yuzhou Nie, Anna Sztyber-Betley, Paolo Faraboschi, Robin Riblet, Jonathan Crozier, Shiv Halasyamani, Shreyas Verma, Prashant Joshi, Eli Meril, Ziqiao Ma, Jérémy Andréoletti, Raghav Singhal, Jacob Platnick, Volodymyr Nevirkovets, Luke Basler, Alexander Ivanov, Seri Khoury, Nils Gustafsson, Marco Piccardo, Hamid Mostaghimi, Qijia Chen, Virendra Singh, Tran Quoc Khánh, Paul Rosu, Hannah Szlyk, Zachary Brown, Himanshu Narayan, Aline Menezes, Jonathan Roberts, William Alley, Kunyang Sun, Arkil Patel, Max Lamparth, Anka Reuel, Linwei Xin, Hanmeng Xu, Jacob Loader, Freddie Martin, Zixuan Wang, Andrea Achilleos, Thomas Preu, Tomek Korbak, Ida Bosio, Fereshteh Kazemi, Ziye Chen, Biró Bálint, Eve J. Y. Lo, Jiaqi Wang, Maria Inês S. Nunes, Jeremiah Milbauer, M. Saiful Bari, Zihao Wang, Behzad Ansarinejad, Yewen Sun, Stephane Durand, Hossam Elgnainy, Guillaume Douville, Daniel Tordera, George Balabanian, Hew Wolff, Lynna Kvistad, Hsiaoyun Milliron, Ahmad Sakor, Murat Eron, Andrew Favre, Shailesh Shah, Xiaoxiang Zhou, Firuz Kamalov, Sherwin Abdoli, Tim Santens, Shaul Barkan, Allison Tee, Robin Zhang, Alessandro Tomasiello, G. Bruno De Luca, Shi-Zhuo Looi, Vinh-Kha Le, Noam Kolt, Jiayi Pan, Emma Rodman, Jacob Drori, Carl J. Fossum, Niklas Muennighoff, Milind Jagota, Ronak Pradeep, Honglu Fan, Jonathan Eicher, Michael

Chen, Kushal Thaman, William Merrill, Moritz Firsching, Carter Harris, Stefan Ciobâcă, Jason Gross, Rohan Pandey, Ilya Gusev, Adam Jones, Shashank Agnihotri, Pavel Zhelnov, Mohammadreza Mofayezi, Alexander Piperski, David K. Zhang, Kostiantyn Dobarskyi, Roman Leventov, Ignat Soroko, Joshua Duersch, Vage Taamazyan, Andrew Ho, Wenjie Ma, William Held, Ruicheng Xian, Armel Randy Zebaze, Mohanad Mohamed, Julian Noah Leser, Michelle X. Yuan, Laila Yacar, Johannes Lengler, Katarzyna Olszewska, Claudio Di Fratta, Edson Oliveira, Joseph W. Jackson, Andy Zou, Muthu Chidambaram, Timothy Manik, Hector Haffenden, Dashiell Stander, Ali Dasouqi, Alexander Shen, Bita Golshani, David Stap, Egor Kretov, Mikalai Uzhou, Alina Borisovna Zhidkovskaya, Nick Winter, Miguel Orbegozo Rodriguez, Robert Lauff, Dustin Wehr, Colin Tang, Zaki Hossain, Shaun Phillips, Fortuna Samuele, Fredrik Ekström, Angela Hammon, Oam Patel, Faraz Farhidi, George Medley, Forough Mohammadzadeh, Madellene Peñaflor, Haile Kassahun, Alena Friedrich, Rayner Hernandez Perez, Daniel Pyda, Taom Sakal, Omkar Dhamane, Ali Khajegili Mirabadi, Eric Hallman, Kenchi Okutsu, Mike Battaglia, Mohammad Maghsoudimehrabani, Alon Amit, Dave Hulbert, Roberto Pereira, Simon Weber, Handoko, Anton Peristyy, Stephen Malina, Mustafa Mehkary, Rami Aly, Frank Reidegeld, Anna-Katharina Dick, Cary Friday, Mukhwinder Singh, Hassan Shapourian, Wanyoung Kim, Mariana Costa, Hubeyb Gurdogan, Harsh Kumar, Chiara Ceconello, Chao Zhuang, Haon Park, Micah Carroll, Andrew R. Tawfeek, Stefan Steinerberger, Daattavya Aggarwal, Michael Kirchhof, Linjie Dai, Evan Kim, Johan Ferret, Jainam Shah, Yuzhou Wang, Minghao Yan, Krzysztof Burdzy, Lixin Zhang, Antonio Franca, Diana T. Pham, Kang Yong Loh, Joshua Robinson, Abram Jackson, Paolo Giordano, Philipp Petersen, Adrian Cosma, Jesus Colino, Colin White, Jacob Votava, Vladimir Vinnikov, Ethan Delaney, Petr Spelda, Vit Stritecky, Syed M. Shahid, Jean-Christophe Mourrat, Lavr Vetoshkin, Koen Sponselee, Renas Bacho, Zheng-Xin Yong, Florencia de la Rosa, Nathan Cho, Xiuyu Li, Guillaume Malod, Orion Weller, Guglielmo Albani, Leon Lang, Julien Laurendeau, Dmitry Kazakov, Fatimah Adesanya, Julien Portier, Lawrence Hollom, Victor Souza, Yuchen Anna Zhou, Julien Degorre, Yiğit Yaln, Gbenga Daniel Obikoya, Rai Michael Pokorny, Filippo Bigi, M. C. Boscá, Oleg Shumar, Kaniuar Bacho, Gabriel Recchia, Mara Popescu, Nikita Shulga, Ngefor Mildred Tanwie, Thomas C. H. Lux, Ben Rank, Colin Ni, Matthew Brooks, Alesia Yakimchyk, Huanxu Quinn Liu, Stefano Cavalleri, Olle Häggström, Emil Verkama, Joshua Newbould, Hans Gundlach, Leonor Brito-Santana, Brian Amaro, Vivek Vajipey, Rynaa Grover, Ting Wang, Yosi Kratish, Wen-Ding Li, Sivakanth Gopi, Andrea Caciolai, Christian Schroeder de Witt, Pablo Hernández-Cámara, Emanuele Rodolà, Jules Robins, Dominic Williamson, Brad Raynor, Hao Qi, Ben Segev, Jingxuan Fan, Sarah Martinson, Erik Y. Wang, Kaylie Hausknecht, Michael P. Brenner, Mao Mao, Christoph Demian, Peyman Kassani, Xinyu Zhang, David Avagian, Eshawn Jessica Scipio, Alon Ragoler, Justin Tan, Blake Sims, Rebeka Plecnik, Aaron Kirtland, Omer Faruk Bodur, D. P. Shinde, Yan Carlos Leyva Labrador, Zahra Adoul, Mohamed Zekry, Ali Karakoc, Tania C. B. Santos, Samir Shamseldeen, Loukmane Karim, Anna Liakhovitskaia, Nate Resman, Nicholas Farina, Juan Carlos Gonzalez, Gabe Maayan, Earth Anderson, Rodrigo De Oliveira Pena, Elizabeth Kelley, Hodjat Mariji, Rasoul Pouriamanesh, Wentao Wu, Ross Finocchio, Ismail Alarab, Joshua Cole, Danyelle Ferreira, Bryan Johnson, Mohammad Safdari, Liangti Dai, Siriphan Arthornthurasuk, Isaac C. McAlister, Alejandro José Moyano, Alexey Pronin, Jing Fan, Angel Ramirez-Trinidad, Yana Malysheva, Daphiny Pottmaier, Omid Taheri, Stanley Stepanic, Samuel Perry, Luke Askew, Raúl Adrián Huerta Rodrguez, Ali M. R. Minissi, Ricardo Lorena, Krishnamurthy Iyer, Arshad Anil Fasiludeen, Ronald Clark, Josh Ducey, Matheus Piza, Maja Somrak, Eric Vergo, Juehang Qin, Benjámin Borbás, Eric Chu, Jack Lindsey, Antoine Jallon, I. M. J. McInnis, Evan Chen, Avi Semler, Luk Gloor, Tej Shah, Marc Carauleanu, Pascal Lauer, Tran Duc Huy, Hossein Shahrtash, Emilien Duc, Lukas Lewark, Assaf Brown, Samuel Albanie, Brian Weber, Warren S. Vaz, Pierre Clavier, Yiyang Fan, Gabriel Poesia Reis e Silva, Long Tony Lian, Marcus Abramovitch, Xi Jiang, Sandra Mendoza, Murat Islam, Juan Gonzalez, Vasilios Mavroudis, Justin Xu, Pawan Kumar, Laxman Prasad Goswami, Daniel Bugas, Nasser Heydari, Ferenc Jeanplong, Thorben Jansen, Antonella Pinto, Archimedes Apronti, Abdallah Galal, Ng Ze-An, Ankit Singh, Tong Jiang, Joan of Arc Xavier, Kanu Priya Agarwal, Mohammed Berkani, Gang Zhang, Zhehang Du, Benedito Alves de Oliveira Junior, Dmitry Malishev, Nicolas Remy, Taylor D. Hartman, Tim Tarver, Stephen Mensah, Gautier Abou Loume, Wiktor Morak, Farzad Habibi, Sarah Hoback, Will Cai, Javier Gimenez, Roselynn Grace Montecillo, Jakub Łucki, Russell Campbell, Asankhaya Sharma, Khalida Meer, Shreen Gul, Daniel Espinosa Gonzalez, Xavier Alapont, Alex Hoover, Gunjan Chhablani, Freddie Vargus, Arunim Agarwal, Yibo Jiang, Deepakkumar Patil, David Outevsky, Kevin Joseph Scaria, Rajat Maheshwari, Abdelkader Dendane, Priti Shukla, Ashley Cartwright, Sergei Bogdanov, Niels Mündler, Sören Möller, Luca Arnaboldi, Kunvar Thaman, Muhammad Rehan Siddiqi, Prajvi Saxena, Himanshu Gupta, Tony Fruhauff, Glen Sherman, Mátyás Vincze, Siranut Usawasutsakorn, Dylan Ler, Anil Radhakrishnan, Innocent Enyekwe, Sk Md Salauddin, Jiang Muzhen, Aleksandr Maksapetyan, Vivien Rossbach, Chris Harjadi, Mohsen Bahaloohoreh, Claire Sparrow, Jasdeep Sidhu, Sam Ali, Song Bian, John Lai, Eric Singer, Justine Leon Uro, Greg Bateman, Mohamed Sayed, Ahmed Menshawy, Darling Duclosel, Dario Bezzi, Yashaswini Jain, Ashley Aaron, Murat Tiryakioglu, Sheeshram Siddh, Keith Krenek, Imad Ali Shah, Jun Jin, Scott Creighton, Denis Peskoff, Zienab EL-Wasif, Ragavendran P, Michael Richmond, Joseph McGowan, Tejal Patwardhan, Hao-Yu Sun, Ting Sun, Nikola Zubić, Samuele Sala, Stephen Ebert, Jean Kaddour, Manuel Schottdorf, Dianzhuo Wang, Gerol Petruzella, Alex Meiburg, Tilen Medved, Ali ElSheikh, S. Ashwin Hebbar, Lorenzo Vaquero, Xianjun Yang, Jason Poulos, Vilém Zouhar, Sergey Bogdanik, Mingfang Zhang, Jorge Sanz-Ros, David Anugraha, Yinwei Dai, Anh N. Nhu, Xue Wang, Ali Anil Demircali, Zhibai Jia, Yuyin Zhou, Juncheng Wu, Mike He, Nitin Chandok, Aarush Sinha, Gaoxiang Luo, Long Le, Mickaël Noyé, Michał Perełkiewicz, Ioannis Pantidis, Tianbo Qi, Soham Sachin Purohit, Letitia Parcalabescu,

Thai-Hoa Nguyen, Genta Indra Winata, Edoardo M. Ponti, Hanchen Li, Kaustubh Dhole, Jongee Park, Dario Abbondanza, Yuanli Wang, Anupam Nayak, Diogo M. Caetano, Antonio A. W. L. Wong, Maria del Rio-Chanona, Dániel Kondor, Pieter Francois, Ed Chalstrey, Jakob Zsambok, Dan Hoyer, Jenny Reddish, Jakob Hauser, Francisco-Javier Rodrigo-Ginés, Suchandra Datta, Maxwell Shepherd, Thom Kamphuis, Qizheng Zhang, Hyunjun Kim, Ruiji Sun, Jianzhu Yao, Franck Dernoncourt, Satyapriya Krishna, Sina Rismanchian, Bonan Pu, Francesco Pinto, Yingheng Wang, Kumar Shridhar, Kalon J. Overholt, Glib Briia, Hieu Nguyen, David Quod Soler Bartomeu, Tony CY Pang, Adam Wecker, Yifan Xiong, Fanfei Li, Lukas S. Huber, Joshua Jaeger, Romano De Maddalena, Xing Han Lù, Yuhui Zhang, Claas Beger, Patrick Tser Jern Kon, Sean Li, Vivek Sanker, Ming Yin, Yihao Liang, Xinlu Zhang, Ankit Agrawal, Li S. Yifei, Zechen Zhang, Mu Cai, Yasin Sonmez, Costin Cozianu, Changhao Li, Alex Slen, Shoubin Yu, Hyun Kyu Park, Gabriele Sarti, Marcin Briański, Alessandro Stolfo, Truong An Nguyen, Mike Zhang, Yotam Perlitz, Jose Hernandez-Orallo, Runjia Li, Amin Shabani, Felix Juefei-Xu, Shikhar Dhingra, Orr Zohar, My Chiffon Nguyen, Alexander Pondaven, Abdurrahim Yilmaz, Xuandong Zhao, Chuanyang Jin, Muyan Jiang, Stefan Todoran, Xinyao Han, Jules Kreuer, Brian Rabern, Anna Plassart, Martino Maggetti, Luther Yap, Robert Geirhos, Jonathon Kean, Dingsu Wang, Sina Mollaei, Chenkai Sun, Yifan Yin, Shiqi Wang, Rui Li, Yaowen Chang, Anjiang Wei, Alice Bizeul, Xiaohan Wang, Alexandre Oliveira Arrais, Kushin Mukherjee, Jorge Chamorro-Padial, Jiachen Liu, Xingyu Qu, Junyi Guan, Adam Bouyamourn, Shuyu Wu, Martyna Plomecka, Junda Chen, Mengze Tang, Jiaqi Deng, Shreyas Subramanian, Haocheng Xi, Haoxuan Chen, Weizhi Zhang, Yinuo Ren, Haoqin Tu, Sejong Kim, Yushun Chen, Sara Vera Marjanović, Junwoo Ha, Grzegorz Luczyna, Jeff J. Ma, Zewen Shen, Dawn Song, Cedegao E. Zhang, Zhun Wang, Gaël Gendron, Yunze Xiao, Leo Smucker, Erica Weng, Kwok Hao Lee, Zhe Ye, Stefano Ermon, Ignacio D. Lopez-Miguel, Theo Knights, Anthony Gitter, Namkyu Park, Boyi Wei, Hongzheng Chen, Kunal Pai, Ahmed Elkhanany, Han Lin, Philipp D. Siedler, Jichao Fang, Ritwik Mishra, Károly Zsolnai-Fehér, Xilin Jiang, Shadab Khan, Jun Yuan, Rishab Kumar Jain, Xi Lin, Mike Peterson, Zhe Wang, Aditya Malusare, Maosen Tang, Isha Gupta, Ivan Fosin, Timothy Kang, Barbara Dworakowska, Kazuki Matsumoto, Guangyao Zheng, Gerben Sewuster, Jorge Pretel Villanueva, Ivan Rannev, Igor Chernyavsky, Jiale Chen, Deepayan Banik, Ben Racz, Wenchao Dong, Jianxin Wang, Laila Bashmal, Duarte V. Gonçalves, Wei Hu, Kaushik Bar, Ondrej Bohdal, Atharv Singh Patlan, Shehzaad Dhuliawala, Caroline Geirhos, Julien Wist, Yuval Kansal, Bingsen Chen, Kutay Tire, Atak Talay Yücel, Brandon Christof, Veerupaksh Singla, Zijian Song, Sanxing Chen, Jiaxin Ge, Kaustubh Ponkshe, Isaac Park, Tianneng Shi, Martin Q. Ma, Joshua Mak, Sherwin Lai, Antoine Moulin, Zhuo Cheng, Zhanda Zhu, Ziyi Zhang, Vaidehi Patil, Ketan Jha, Qiutong Men, Jiaxuan Wu, Tianchi Zhang, Bruno Hebling Vieira, Alham Fikri Aji, Jae-Won Chung, Mohammed Mahfoud, Ha Thi Hoang, Marc Sperzel, Wei Hao, Kristof Meding, Sihan Xu, Vassilis Kostakos, Davide Manini, Yueying Liu, Christopher Toukmaji, Eunmi Yu, Arif Engin Demircali, Zhiyi Sun, Ivan Dewerpe, Hongsen Qin, Roman Pflugfelder, James Bailey, Johnathan Morris, Ville Heilala, Sybille Rosset, Zishun Yu, Peter E. Chen, Woongyeong Yeo, Eeshaan Jain, Sreekar Chigurupati, Julia Chernyavsky, Sai Prajwal Reddy, Subhashini Venugopalan, Hunar Batra, Core Francisco Park, Hieu Tran, Guilherme Maximiano, Genghan Zhang, Yizhuo Liang, Hu Shiyu, Rongwu Xu, Rui Pan, Siddharth Suresh, Ziqi Liu, Samaksh Gulati, Songyang Zhang, Peter Turchin, Christopher W. Bartlett, Christopher R. Scotese, Phuong M. Cao, Ben Wu, Jacek Karwowski, and Davide Scaramuzza. A benchmark of expert-level academic questions to assess ai capabilities. *Nature*, 649(8099):1139–1146, January 2026. ISSN 1476-4687. doi: 10.1038/s41586-025-09962-4. URL `http://dx.doi.org/10.1038/s41586-025-09962-4`.

Paul Pop, Michael Lander Raagaard, Silviu S. Craciunas, and Wilfried Steiner. Design optimization of cyber-physical distributed systems using IEEE time-sensitive networks (TSN). *IET-CPS*, 1(1):86–94, 2016. doi: https://doi.org/10.1049/iet-cps.2016.0021.

Laria Reynolds and Kyle McDonell. Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI EA '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380959. doi: 10.1145/3411763.3451760. URL `https://doi.org/10.1145/3411763.3451760`.

Walid Saad, Mehdi Bennis, and Mingzhe Chen. A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems. *IEEE Network*, 34(3):134–142, 2020. doi: 10.1109/MNET.001.1900287.

Jorge Sanchez-Garrido, Beatriz Aparicio, José Gabriel Ramírez, Rafael Rodriguez, Mariasole Melara, Lorenzo Cercós, Eduardo Ros, and Javier Diaz. Implementation of a Time-Sensitive Networking (TSN) Ethernet Bus for Microlaunchers. *IEEE Transactions on Aerospace and Electronic Systems*, 57(5):2743–2758, 2021. doi: 10.1109/TAES.2021.3061806.

Ramon Serna Oliver, Silviu S. Craciunas, and Wilfried Steiner. IEEE 802.1Qbv Gate Control List Synthesis Using Array Theory Encoding. In *2018 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pages 13–24, 2018. doi: 10.1109/RTAS.2018.00008.

Prakhar Sharma and Vinod Yegneswaran. PROSPER: Extracting Protocol Specifications Using Large Language Models. In *Proceedings of the 22nd ACM Workshop on Hot Topics in Networks*, HotNets '23, page 41–47,

New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400704154. doi: 10.1145/3626111.3628205. URL https://doi.org/10.1145/3626111.3628205.

Yongliang Shen, Kaitao Song, Xu Tan, Wenqi Zhang, Kan Ren, Siyu Yuan, Weiming Lu, Dongsheng Li, and Yueting Zhuang. TaskBench: Benchmarking Large Language Models for Task Automation. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 4540–4574. Curran Associates, Inc., 2024. doi: 10.52202/079017-0148. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/085185ea97db31ae6dcac7497616fd3e-Paper-Datasets_and_Benchmarks_Track.pdf.

Johannes Specht and Soheil Samii. Urgency-Based Scheduler for Time-Sensitive Switched Ethernet Networks. In *2016 28th Euromicro Conference on Real-Time Systems (ECRTS)*, pages 75–85, 2016. doi: 10.1109/ECRTS.2016.27.

Liangtai Sun, Yang Han, Zihan Zhao, Da Ma, Zhennan Shen, Baocai Chen, Lu Chen, and Kai Yu. SciEval: a multi-level large language model evaluation benchmark for scientific research. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. ISBN 978-1-57735-887-9. doi: 10.1609/aaai.v38i17.29872. URL https://doi.org/10.1609/aaai.v38i17.29872.

Changjie Wang, Mariano Scazzariello, Alireza Farshin, Simone Ferlin, Dejan Kostić, and Marco Chiesa. NetConfEval: Can LLMs Facilitate Network Configuration? *Proc. ACM Netw.*, 2(CoNEXT2), June 2024a. doi: 10.1145/3656296. URL https://doi.org/10.1145/3656296.

Xiaolong Wang, Haipeng Yao, Tianle Mai, Zehui Xiong, Fu Wang, and Yunjie Liu. Joint Routing and Scheduling With Cyclic Queuing and Forwarding for Time-Sensitive Networks. *IEEE Transactions on Vehicular Technology*, 72(3):3793–3804, 2023. doi: 10.1109/TVT.2022.3216958.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A More Robust and Challenging Multi-Task Language Understanding Benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024b. URL https://openreview.net/forum?id=y10DM6R2r3.

Stefan Windmann, Janis Albrecht, Maxim Friesen, and Jürgen Jasperneite. NetPilot - Towards LLM-Assisted Configuration of Hybrid TSN/5G Networks. In *2025 IEEE 30th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–4, 2025. doi: 10.1109/ETFA65518.2025.11205555.

Jiacheng Xie, Yang Yu, Ziyang Zhang, Shuai Zeng, Jiaxuan He, Ayush Vasireddy, Xiaoting tang, Congyu Guo, Lening Zhao, Congcong Jing, Guanghui An, and Dong Xu. TCM-ladder: A benchmark for multimodal question answering on traditional chinese medicine. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025. URL https://openreview.net/forum?id=ZDrT1eG54T.

Jinli Yan, Wei Quan, Xuyan Jiang, and Zhigang Sun. Injection time planning: Making cqf practical in time-sensitive networking. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, pages 616–625, 2020. doi: 10.1109/INFOCOM41043.2020.9155434.

Jialin Yang, Dongfu Jiang, Tony He, Sherman Siu, Yuxuan Zhang, Disen Liao, Zhuofeng Li, Huaye Zeng, Yiming Jia, Haozhe Wang, Benjamin Schneider, Chi Ruan, Wentao Ma, Zhiheng Lyu, Yifei Wang, Yi Lu, Quy Duc Do, Ziyan Jiang, Ping Nie, and Wenhu Chen. Structeval: Benchmarking LLMs' capabilities to generate structural outputs. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=buDwV7LUA7. J2C Certification.

Kouros Zanbouri, Md. Noor-A-Rahim, Jobish John, Cormac J. Sreenan, H. Vincent Poor, and Dirk Pesch. A Comprehensive Survey of Wireless Time-Sensitive Networking (TSN): Architecture, Technologies, Applications, and Open Issues. *IEEE Communications Surveys & Tutorials*, 27(4):2129–2155, 2025. doi: 10.1109/COMST.2024.3486618.

Tianyu Zhang, Gang Wang, Chuanyu Xue, Jiachen Wang, Mark Nixon, and Song Han. Time-Sensitive Networking (TSN) for Industrial Automation: Current Advances and Future Directions. *ACM Comput. Surv.*, 57(2), October 2024. ISSN 0360-0300. doi: 10.1145/3695248. URL https://doi.org/10.1145/3695248.

Luxi Zhao, Paul Pop, Zhong Zheng, and Qiao Li. Timing Analysis of AVB Traffic in TSN Networks Using Network Calculus. In *2018 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pages 25–36, 2018. doi: 10.1109/RTAS.2018.00009.

Luxi Zhao, Paul Pop, Zhong Zheng, Hugo Daigmorte, and Marc Boyer. Latency Analysis of Multiple Classes of AVB Traffic in TSN With Standard Credit Behavior Using Network Calculus. *IEEE Transactions on Industrial Electronics*, 68(10):10291–10302, 2021. doi: 10.1109/TIE.2020.3021638.

Luxi Zhao, Paul Pop, and Sebastian Steinhorst. Quantitative Performance Comparison of Various Traffic Shapers in Time-Sensitive Networking. *IEEE Transactions on Network and Service Management*, 19(3):2899–2928, 2022. doi: 10.1109/TNSM.2022.3180160.

Luxi Zhao, Yida Yan, and Xuan Zhou. Minimum Bandwidth Reservation for CBS in TSN With Real-Time QoS Guarantees. *IEEE Transactions on Industrial Informatics*, 20(4):6187–6198, 2024. doi: 10.1109/TII.2023.3342466.

6 Limitations and Broader Impact

6.1 Limitations

While TSNBench fills a significant research gap and proposes a step forward towards evaluating TSN capabilities in LLMs, it has several limitations:

Dataset scope: TSNBench currently only covers CBS and CQF in open-ended questions. Evaluating other TSN mechanisms is necessary to fully cover the entire TSN mechanism.

Prompt Design: TSNBench does not provide any mathematical equation to the model as input for NC WCD calculation for CBS or the upper bound delay calculation of CQF.

MCQA Scope: MCQs are solely developed using published research papers and the IEEE standards are not used to generate the MCQs. Solving the license issue and utilizing standards to include MCQs using IEEE 802.1 standard will enhance the entire MCQA dataset.

Topology coverage: TSNBench open-ended question currently covers three different topologies: one-switch, medium-mesh, and ring topology. Covering diverse topologies and flow parameters will present a comprehensive evaluation.

6.2 Improvement Strategies

To address the limitations of TSNBench, we propose the following additions and improvements in the future version of TSNBench.

1. **Larger and more diverse dataset:** Our current TSNBench dataset covers 100 TCs across three topology types. In future versions, we will include larger and more complex topologies with higher flow counts. As model performance improves, more complex open-ended evaluations should be integrated with complex topologies and combined TSN mechanisms.
2. **Additional scheduling mechanisms:** TSNBench currently evaluates CBS and CQF. Future versions should extend to TAS and ATS to cover a broader range of the TSN standard suite.
3. **Updated MCQA:** Our MCQA dataset was developed using open-source research documents. In future work, we will update the dataset with MCQAs formulated directly from TSN standards.
4. **Fine-tuned and domain-adapted models.** TSNBench currently evaluates general-purpose LLMs without any TSN-specific fine-tuning. Future versions should benchmark domain-adapted models trained on TSN standards and network calculus literature.

6.3 Broader Impact

TSNBench enables the real-time systems community and the machine learning community to objectively measure LLM performance and readiness for management and deployment assistance in safety-critical deterministic networks. By highlighting the critical aspects of TSN and the performance gap of the models between MCQA and computational reasoning, TSNBench alerts the incompetence of the models which may lead to misconfigurations and safety-critical issues. This benchmark provides a concrete direction to improve LLMs for deterministic networking. TSNBench further highlights the potential benefits of using LLMs thereby automating the management and deployment of TSN networks. Moreover, open-sourced ground truth WCD values computed by NC solvers provide a reliable resource for the entire community to further evaluate different benchmarking datasets.

6.4 Negative Impacts

While TSNBench is intended to advance research on LLM proficiency in TSN, we acknowledge the following potential negative impacts.

Overreliance on model outputs: Models trained on the open-access dataset provided by TSNBench may achieve high accuracy on WCD analysis tasks, which could lead practitioners to deploy such models directly in real-world deployments without independent verification. Any WCD values or

network configuration decisions produced by an LLM should be verified using formally verified solvers and NC tools before real-world deployment.

False confidence from MCQA performance: Our results demonstrate that strong MCQA performance does not transfer to open-ended WCD estimation. A practitioner or system engineer who evaluates an LLM solely on MCQA benchmarks may incorrectly conclude that the model is suitable for TSN configuration tasks, leading to unsafe deployments in systems where timing guarantees are required.

Data contamination and benchmark overfitting: As TSNBench is released as an open-access dataset, future models may be trained directly on the benchmark questions, leading to inflated performance that does not reflect genuine TSN reasoning capability. We recommend that researchers introduce randomization in the test cases to prevent bias in results. Researchers should be cautious when interpreting results from models whose training data may overlap with the TSNBench dataset.

Misuse of the dataset: The dataset can be used to train models to configure TSN networks. Owing to the safety-critical nature of TSN applications, such models could potentially be exploited by attackers to manipulate network configurations, introduce timing violations, or deliberately cause deadline misses in industrial and automotive systems.

7 Time-Sensitive Networking

Time-Sensitive Networking (TSN) [Finn, 2018] is a set of amendments and additions to the IEEE 802.1 standards that, since its inception in 2012, has become one of the most relevant technologies for enabling deterministic and real-time communications over Ethernet networks. TSN extends standard Ethernet by introducing mechanisms for bounded latency, low jitter, and high reliability, making it suitable for applications such as industrial automation, automotive systems, and professional audio-video networks. Figure 8 showcases a simple TSN network with flows.

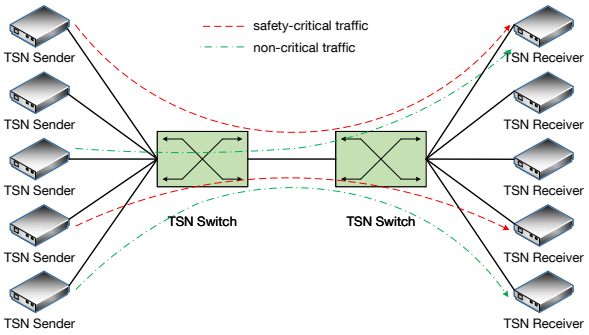

Figure 8: A sample TSN network with TSN senders, receivers, and TSN switches in the network. TSN senders are sending mixed-critical including safety-critical and non-critical traffic to the TSN receivers.

In TSN, communication between end-stations is based on the transmission of Ethernet frames across a network of interconnected Ethernet links and TSN switches. These switches, as well as the output ports of end-stations, implement a queuing architecture with up to eight First-In-First-Out (FIFO) queues, each associated with one of the eight traffic priorities defined in IEEE 802.1Q [802, 2018]. TSN is not just limited to wired domain. The growing necessity of deterministic communication has extended to wireless domain gaining a significant interest in wireless-TSN networks. Although TSN is fundamentally an IEEE 802.1 bridged Ethernet technology, wireless and 5G-TSN [Debnath et al., 2023a] integration requires additional adaptation or translation functions, together with time-synchronization mechanisms that preserve deterministic latency guarantees across heterogeneous network segments. We showcase a 5G-TSN system in Figure 9, where TSN senders are sending mixed criticality traffic types to wireless receiver nodes over a TSN switch and 5G system in the network. Some of the most commonly used abbreviations in TSN are given in Table 4.

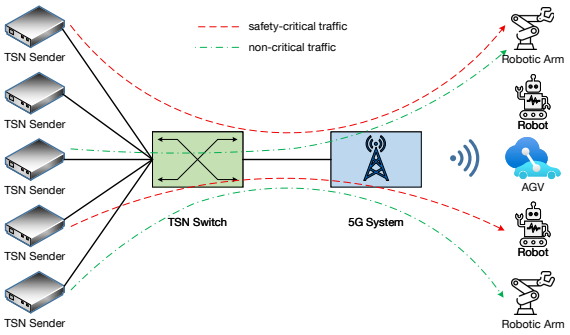

Figure 9: A sample wireless-TSN network with TSN senders, wireless receivers (such as robotic arm and automated guided vehicles (AGVs)), TSN switches, and 5G system in the network. TSN senders are sending mixed-critical including safety-critical and non-critical traffic to the wireless receivers.

Frames are classified into traffic classes and assigned to egress queues based on their priority, with transmission selection typically governed by strict priority. Industrial TSN traffic is commonly categorized into traffic types such as isochronous traffic, cyclic-synchronous traffic, cyclic-asynchronous traffic, network-control traffic, alarms and events, configuration and diagnostics, and best-effort traffic [Ademaj et al., 2019]. These traffic types require different timing guarantees: safety-critical isochronous traffic is typically mapped to time-triggered (TT) traffic, requiring guaranteed latency and bounded jitter, and is commonly handled by time-triggered mechanisms such as the Time-Aware Shaper (TAS) [Craciunas et al., 2016, Serna Oliver et al., 2018]. In contrast, cyclic-synchronous or cyclic-asynchronous traffic that requires bounded end-to-end latency but less stringent jitter control is commonly mapped to AVB stream traffic and is often supported by the Credit-Based Shaper (CBS) [Zhao et al., 2018]. TSN also defines mechanisms such as Asynchronous Traffic Shaping (ATS) [Specht and Samii, 2016, Debnath et al., 2023b, Nasrallah et al., 2019], Frame preemption (FP) [Debnath et al., 2024], and Cyclic Queuing and Forwarding (CQF) [Wang et al., 2023, Debnath et al., 2025a, Yan et al., 2020] to provide deterministic communication under different traffic and deployment assumptions.

These mechanisms regulate when and how frames are transmitted, allowing the network to provide guarantees such as bounded delay, jitter, and controlled bandwidth allocation. In the MCQA dataset of TSNBench, we covered the basics of different TSN mechanisms, including TAS, CBS, ATS, CQF, and CBS. The MCQAs are theoretical in nature and cover the basic understanding of the mechanisms without going into their mathematical or analytical details. In contrast, for the open-ended mechanisms, we evaluate the capability of the models to perform numerical analysis, formulate mathematical equations, and find the WCD values for the flows in the network. For this, we selected two TSN mechanisms: CBS and CQF. The WCD values of the flows using the CBS mechanism are calculated using NC analysis, which is mathematically complex. Therefore, we also evaluate the CQF mechanism as a simpler mechanism. The WCD values of the flows using the CQF mechanism can be directly calculated using the routing of the flow and the cycle duration. The detailed working mechanism and architecture of CQF and CBS are described in detail in the Appendix 9 and 10. The theory of NC is further explained along with the mathematical equations in Appendix 8.

8 Network Calculus Theory

Network Calculus (NC) is a theory for calculating worst-case bounds in communication networks based on min-plus algebra. Its basic paradigm involves two operators: convolution $\otimes$

$$(f \otimes g)(t) = \inf_{0 \le s \le t} \{f(t-s) + g(s)\}, \tag{6}$$

and deconvolution $\oslash$,

$$(f \oslash g)(t) = \sup_{s \ge 0} \{f(t+s) - g(s)\}. \tag{7}$$

Table 4: Abbreviations and mechanisms used in TSNBench.

Keyword	Abbreviations
TSN	Time-Sensitive Networking
TAS	Time Aware Shaper
CBS	Credit-Based Shaper
ATS	Asynchronous Traffic Shaper
CQF	Cyclic Queuing and Forwarding
NC	Network Calculus
WCD	Worst-Case Delay
AVB	Audio Video Bridging
TT	Time-Triggered

Based on this algebra, the arrival curve and the service curve are constructed to describe the maximum arrival traffic data and the minimum service capability over any time interval, respectively. In the hybrid TSN/TAS+CBS architecture, the service for ET traffic is constrained not only by the bandwidth reservation, but also by high-priority TT traffic. We adopt the state-of-the-art network calculus model [Zhao et al., 2021, 2024] to ensure deadline guarantees for ET flows with an arbitrary number of SR classes in the TSN/TAS+CBS architecture. Since, in our open-end CBS questions, we do not have any TAS mechanism, we use the TSN/TAS+CBS architecture without the TAS mechanism in it with only CBS mechanism for the AVB flows in the network.

As described in [Zhao et al., 2024], the service curve $\beta(t)$ is for constraining the minimum service capabilities, satisfying

$$\mathcal{R}^*(t) \geq (\mathcal{R} \otimes \beta)(t). \tag{8}$$

The function $\mathcal{R}(t)$ (resp. $\mathcal{R}^*(t)$) is the input (resp. output) cumulative function counting the total data bits of the flow that arrive at (resp. departure from) the server up to time t. A typical example of a service curve is the rate-latency form,

$$\beta_{R,T}(t) = R[t-T]^+ \tag{9}$$

with the service rate R and latency T. The notation $[x]^+$ equals x if $x \geq 0$, and 0 otherwise.

In the hybrid TSN/TAS+CBS architecture, the CBS service curve [Zhao et al., 2021] for the arbitrary SR Class M_i ($i \in [1, N_{SR}]$) with the impact of TT traffic at the output port h is,

$$\beta^h_{M_i}(t) = idSl^h_{M_i}\left[t - \frac{\alpha^h_{TAS}(t)}{C} - \frac{c^{h,\max}_{M_i}}{idSl^h_{M_i}}\right]^+_{\uparrow}, \tag{10}$$

where $c^{h,\max}_{M_i}$ is the credit upper bound for SR Class M_i,

$$c^{h,\max}_{M_i} = idSl^h_{M_i} \cdot \frac{\sum_{j=1}^{i-1} c^{h,\min}_{M_j} - l^{h,\max}_{>i}}{\sum_{j=1}^{i-1} idSl^h_{M_j} - C}, \tag{11}$$

where $l^{h,\max}_{>i} = \max_{j>i}\{l^{h,\max}_{M_j}, l^{h,\max}_{BE}\}$ is the maximum frame size with priority lower than Class M_i at h, $l^{h,\max}_{M_j}$ is the maximum frame size of Class M_i at h, and $c^{h,\min}_{M_i}$ is the lower credit bound of SR Class M_i,

$$c^{h,\min}_{M_i} = sdSl^h_{M_i} \cdot \frac{l^{h,\max}_{M_i}}{C}. \tag{12}$$

$\alpha^h_{TAS}(t)$ in Eq. (10) is the arrival curve of TT traffic scheduled by GCL.

The arrival curve $\alpha(t)$ is for constraining the arrival process of the flow, satisfying

$$\mathcal{R}(t) \leq (\mathcal{R} \otimes \alpha)(t). \tag{13}$$

A typical example of an arrival curve is the burst-rate form,

$$\alpha(t) = b + \rho \cdot t, \tag{14}$$

for $t > 0$ and 0 otherwise, with the parameters b as the maximum burst tolerance and ρ as the long-term rate of the flow.

For each ET flow f at its source ES h_0, the arrival curve can be modeled as,

$$\alpha_f^{h_0}(t) = b_f^{h_0} + \rho_f^{h_0} t, \tag{15}$$

where $b_f^{h_0} = l_f$, and $\rho_f^{h_0} = l_f / P_f$. The arrival curve of flow f at intermediate node h is the output arrival curve of f departing from the server h^-,

$$\alpha_f^h(t) = \alpha_f^{h^-} \oslash \delta_{D_f^{h^-}}(t), \tag{16}$$

where $D_f^{h^-}$ is the latency upper bound of flow f queuing at server h^-, and $\delta_D(t)$ is the pure-delay function.

The aggregate arrival curve for ET flows of SR Class M_i at h is obtained by summing the arrival curves of individual flows. It also incorporates the link shaping curve and the CBS shaping curve to improve the tightness of the analysis results.

$$\alpha_{M_i}^h(t) = \sum_{h^- \in \mathcal{H}} \sum_{f \in \mathcal{F}_{M_i}^{h^-,h}} \alpha_f^h(t) \wedge \sigma_{link}^{h^-,h}(t) \wedge \sigma_{M_i}^{h^-,h}(t), \tag{17}$$

where $x \wedge y = \min\{x, y\}$, $\sigma_{link}^{h^-,h}(t)$ is the link shaping curve from the preceding output h^- to the current output port h:

$$\sigma_{link}^{h^-,h}(t) = Ct + l_{M_i}^{h^-,h,\max}, \tag{18}$$

considering the packetization impact of the maximum frame size $l_{M_i}^{h^-,h,\max}$ of flows with Class M_i from h^- to h. $\sigma_{M_i}^{h^-,h}(t)$ is the CBS shaping curve of Class M_i from h^- to h:

$$\sigma_{M_i}^{h^-,h}(t) = idSl_{M_i}^{h^-} \left[t - \frac{\beta_{TAS}^{h^-}(t)}{C} + \frac{c_{M_i}^{h^-,\max} - c_{M_i}^{h^-,\min}}{idSl_{M_i}^{h^-}} \right] + l_{M_i}^{h^-,h,\max}, \tag{19}$$

$\beta_{TAS}^h(t)$ represents the minimum service supplied to TT traffic on the output port h.

With NC-based Total Flow Analysis (TFA), the worst-case delay upper bound D_f^h for flow $f \in \mathcal{F}_{M_i}^h$ at h equals the worst-case delay upper bound $D_{M_i}^h$ for all flows with the same priority M_i aggregating at h,

$$D_f^h = D_{M_i}^h = hDev(\alpha_{M_i}^h, \beta_{M_i}^h) = \sup_{t \geq 0} \left\{ \inf \left\{ \tau \geq 0 \mid \alpha_{M_i}^h(t) \leq \beta_{M_i}^h(t+\tau) \right\} \right\} \tag{20}$$

where $\alpha_{M_i}^h(t)$ is the arrival curve of aggregate flows of Class M_i from Eq. (17), and $\beta_{M_i}^h(t)$ is the service curve for Class M_i from Eq. (10). The upper bound of the worst-case end-to-end delay for the flow f is then obtained by summing the per-port latency bounds along its route.

9 Credit-Based Shaper

Credit-Based Shaper (CBS) is a TSN mechanism designed to prevent starvation of lower-priority traffic while guaranteeing a reserved portion of bandwidth for higher-priority queues, thereby providing reliability through bounded end-to-end delays. Traffic assigned to queues using CBS is typically referred to as Audio Video Bridging (AVB) traffic.

Here, we build on the description from [Bujosa Mateu, 2024]. In CBS, each AVB queue is associated with a credit value. This credit increases over time when a frame is waiting to be transmitted or when the credit is negative, and decreases while a frame is being transmitted. Moreover, if the credit is positive and there are no AVB frames waiting to be transmitted, the credit is immediately reset to 0. The rates at which credit is increased and decreased are defined by the parameters *idleSlope* and *sendSlope*, respectively. Each queue implementing CBS is configured with its own *idleSlope*

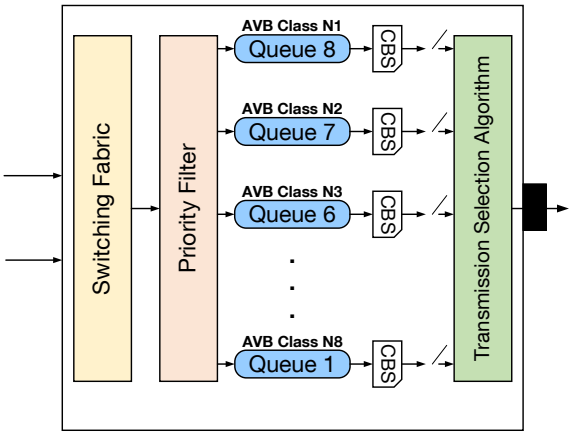

Figure 10: A simple CBS mechanism with eight queues in the egress port of the switch with different AVB class mapped to different queues.

and *sendSlope* values, which determine its allocated bandwidth share. In particular, the bandwidth reserved for a queue is expressed as Eq. (21). A queue is eligible for transmission only when its credit is zero or positive.

$$\text{Reserved BW} = \frac{\textit{idleSlope}}{\textit{idleSlope} + \textit{sendSlope}} \cdot BW \tag{21}$$

Consider the example illustrated in Figure 11, which includes two AVB queues and one Best Effort (BE) queue. Frames 1 and 4 are assigned to the higher-priority AVB queue, while frames 2 and 3 belong to the lower-priority AVB queue and the BE queue, respectively.

At time T0, both AVB queues are eligible for transmission. Due to strict priority scheduling, the higher-priority AVB queue (priority 2) is selected, and frame 1 is transmitted. During this transmission, its credit decreases, while the credit of the lower-priority AVB queue increases because it is waiting.

At time T1, the higher-priority AVB queue has accumulated negative credit and is therefore no longer eligible for transmission. As a result, the lower-priority AVB queue is selected, and frame 2 is transmitted, even though a higher-priority frame (frame 4) is waiting. During this time, the lower-priority queue's credit decreases, while the higher-priority queue's credit recovers.

By time T2, both AVB queues have negative credit, making them ineligible for transmission. Consequently, the BE queue is selected, and frame 3 is transmitted, despite the presence of a higher-priority AVB frame waiting.

Finally, at time T3, the credit of the higher-priority AVB queue has recovered to zero, making it eligible again. Therefore, frame 4 is transmitted.

10 Cyclic Queuing and Forwarding (CQF)

Cyclic Queuing and Forwarding (CQF) [Debnath et al., 2025b] is a TSN shaping mechanism which uses a single cycle duration, denoted as T, across the entire network. T is the minimum scheduling unit where we put the TSN flows. Furthermore, T defines the granularity of the end-to-end delay of the flows in the network. The unit of T is in μs in TSNBench. In a TSN switch, every egress port in the network has eight queues. TSN flows are stored in the queues depending on its priority. In CQF, for each egress port, two queues are used: an `even queue` and an `odd queue`. Figure 14 shows the basic working diagram of CQF with two queues (`even` and `odd`). As shown in Figure 14, CQF works by employing two queues, let's say, Q_8 and Q_7 for TT flows by operating them in a ping-pong manner where Q_7 receives and Q_8 transmits at the first cycle slot (T_1). During the second cycle slot (T_2), Q_8 receives and Q_7 transmits. Selecting or allotting a cycle slot for a flow means selecting the cycle slot number (within the hyperperiod H) and the queue for the flow.

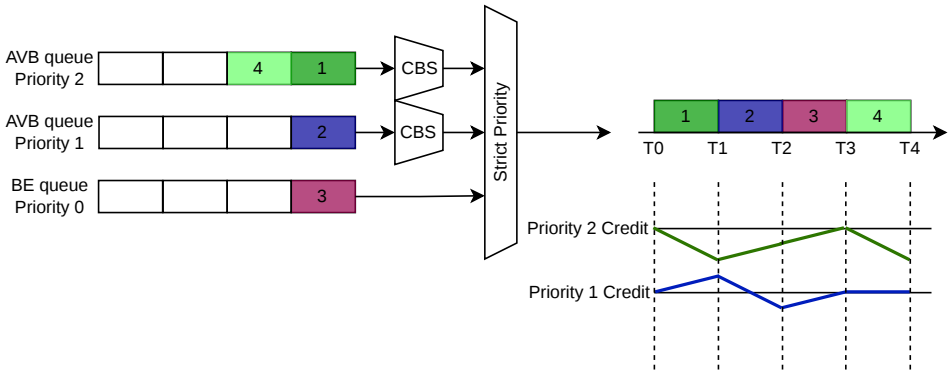

Figure 11: TSN output port with two AVB queues employing CBS and one BE queue.

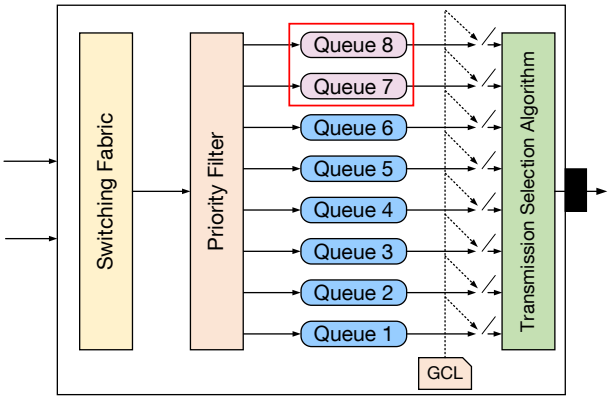

Figure 12: A simple CQF mechanism with eight queues in the egress port of the switch with two queues (Queue 8 and 7) operating as even and odd queue as shown in red.

In the CQF evaluation of TSNBench, we provide the cycle duration (T*) and network-specific delays to the model as input through the prompt.*

WCD CQF: The worst case end-to-end delay of the TT flows in the CQF network is quantified as follows:

$$\text{Max Delay} = f_i.\phi + (\mathrm{SW}_{\mathrm{num}} + 1) \cdot \mathrm{T} + \xi, \tag{22}$$

where $f_i \cdot \phi$ is the offset of the flow f_i in μs, $\mathrm{SW}_{\mathrm{num}}$ is the total number of switches in the route of the TT flow, T is the cycle duration in μs, and ξ denotes the network specific delays: processing delay, propagation delay, and time synchronization error ($\mathrm{sync}_{\mathrm{error}}$).

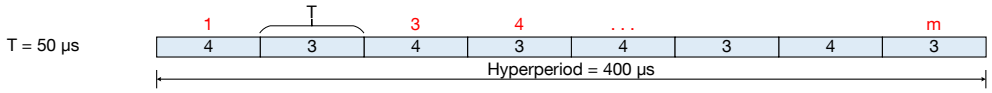

Figure 13: The Hypercyle also known as the scheduling cycle of the CQF (400 μs) with cycle duration (T) of 50μs. The different cycle slots are numbered as 1,2 $\cdots$ m in red.

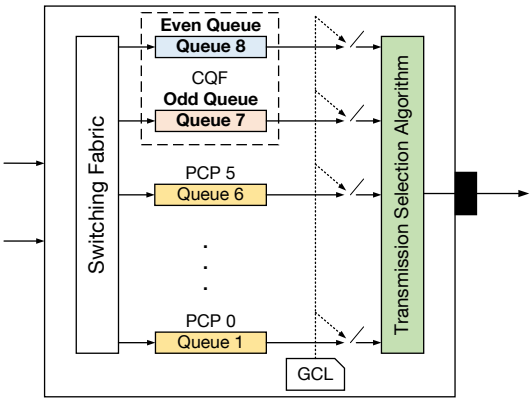

Figure 14: In this figure, we showcase the even and the odd queue in CQF architecture and during one cycle duration one queue receives the flows and another queue transmits the flows received in the previous cycle duration.

11 More on TSNBench

11.1 Human evaluation decision mechanism

To maintain the same standards across all human reviewers, we use the following rules to evaluate the MCQA dataset. There are four possible options for every question in the MCQA dataset.

1. **Accept:**
   i. Technically correct.
   ii. Clearly worded and self-contained.
   iii. Unambiguous options.
   iv. Accurate and sufficient explanation.
   v. The correct answer is actually the correct answer.
2. **Reject:**
   i. Incorrect or misleading.
   ii. Poorly constructed beyond revision.
   iii. Irrelevant to TSN.
   iv. Incomplete information.
   v. Too paper-dependent.
   vi. Duplicate questions.
3. **Revise:**
   i. Minor issues in grammar, clarity, or wording.
   ii. Options need improvement.
   iii. Explanation needs refinement.
4. **Doubtful:**
   i. Paper-specific or uncertain about the correctness of the question.
   ii. Explanation seems questionable.
   iii. Needs further clarification.

   For a doubtful multiple-choice question, we read the research paper and re-evaluate the question. Afterward, the decision can be accept, reject, or revise; if it is still doubtful, we send it to another expert reviewer for a consensus-based group decision.

Key principles followed while reviewing the dataset: We ensured that the MCQAs were technically accurate and aligned with TSN fundamentals. We avoided tricky questions and preferred clarity over

complexity. The same set of rules was given to all expert reviewers who worked on this dataset and served as human judges. After the review, 185 questions were revised by the domain experts, as shown below in Table 5.

Table 5: Statistical data of the MCQA dataset after domain expert human review.

Category	Count
Total questions revised by domain experts	185

11.2 Sample Questions

We present three representative sample questions from our MCQA dataset below.

Q1 *TSN Keyword*

What does TAS stand for in TSN traffic management?

A. Transmission Access Scheduler
B. Traffic Analysis System
C. Time-Aware Shaper
D. Traffic Admission Service

Correct Answer: **C**

Q2 *Research Paper*

In a Cyclic Queuing and Forwarding (CQF) network what fundamental limitation would prevent effective fault tolerance using Frame Replication and Elimination for Reliability (FRER) in a linear topology where each switch has maximum transmission unit (MTU) sized frames frequently queued?

A. CQF's ping-pong queue switching would create timing conflicts with FRER's frame elimination mechanism.
B. EMI interference would corrupt both original and replicated frames equally, making spatial redundancy ineffective.
C. FRER cannot detect bit errors caused by EMI since it lacks Cyclic Redundancy Check (CRC) verification capabilities.
D. Linear topologies cannot provide the disjoint paths required for FRER's spatial redundancy approach, forcing expensive hardware additions.

Correct Answer: **D**

Q3 *Research Paper*

What fundamental challenge makes the Time Aware Shaper (TAS) implementation complex despite its ability to provide guaranteed end-to-end delays?

A. The requirement to synchronize all network devices to a common time reference.
B. The need to maintain separate queues for each traffic class simultaneously.
C. The difficulty in estimating worst-case transmission times for variable-length frames.
D. The synthesis of the gate control list, which is an NP-complete problem.

Correct Answer: **D**

11.3 Prompt design of open-ended questions

For the CBS and CQF mechanisms, two different approaches are used for WCD calculation. NC is used to calculate the CBS WCD, whereas an analytical mathematical calculation is used to find the

WCD for the CQF mechanism. Since these two mechanisms work differently, we design prompts tailored to each mechanism.

Role: We start by defining the role of the model: "You are an expert Time-Sensitive Networking (TSN) orchestrator." We inject three network inputs: (i) network topology, (ii) TSN flow information, and (iii) the routes of the flows. We use the prompt-as-program [Reynolds and McDonell, 2021] approach to separate the network topology, flow information, and flow routes. All of these are provided in text format. However, to evaluate different topologies, flows, and routes, we separate them from the prompt logic. This ensures that the prompt remains the same across different network topologies and parameters.

Constants: To correctly calculate the WCD, information about the network parameters is required. To prevent the model from assuming these values and to keep the constant values consistent across all models, we provide this information in the prompt.

Constants for CBS open-ended questions:

$Bandwidth = 100$ Mbps,

$Propagation\ delay = 1\ \mu$s,

$Switching\ delay = 1\ \mu$s,

$Time\ synchronization\ error = 1\ \mu$s,

The switches of the network are cut-through switches,

$IdleSlope = 75\%$

By controlling these network parameters, we directly mitigate hallucinations and assumptions about numerical values.

Architecture Restriction: TSN supports multiple architectures that affect the Quality of Service (QoS) and the WCD of the flows. The prompt restricts the model to using only one TSN mechanism through the following directive.

For the CBS mechanism, we use:

TSN Mechanism:

Only Credit-Based Shaper (CBS, IEEE 802.1Qav) is allowed;

All flows are AVB Class A, PCP = 6, using queue 6 only.

For the CQF mechanism, we use:

TSN Mechanism:

Only Cyclic Queuing and Forwarding (CQF, IEEE 802.1Qch) is allowed;

All flows are TT, PCP = 7, using queue 7 (odd) and 6 (even) only.

Our reasoning is that letting the model select the TSN architecture or mechanism is a separate benchmarking problem, where the model is evaluated on architecture design performance. In TSNBench, our goal is to benchmark LLMs in TSN. Without an explicit restriction, the model may select an incorrect or inappropriate mechanism, producing a hallucinated architecture that does not satisfy the QoS requirements of the flows. This restriction forces the model to use a single solution space. It further ensures that the WCDs provided by different models are not caused by architectural faults or mechanism selection ambiguity, but rather by calculation and implementation errors within the specified mechanism.

Structured Output: We instruct the model through the prompt to provide the output strictly in JSON format [Yang et al., 2026].

11.4 TSNBench Open-Ended Question Details

For the open-ended questions, there are three variable entries: network topology, flow information, and flow routing. We use the K-shortest path algorithm to determine the routes of the flows. The routes are then directly provided to the models as input for further evaluation.

Network Topologies Used: For the open-ended questions, we selected three different topologies to evaluate the models: a one-switch topology, a medium-mesh topology, and an industrial ring topology. Figures 15, 16, and 17 represent the one-switch, medium-mesh, and ring topologies used in TSNBench, respectively.

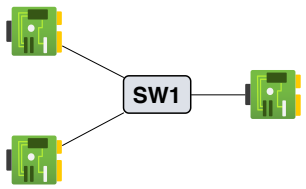

Figure 15: One-switch topology used to evaluate open-ended questions in TSNBench.

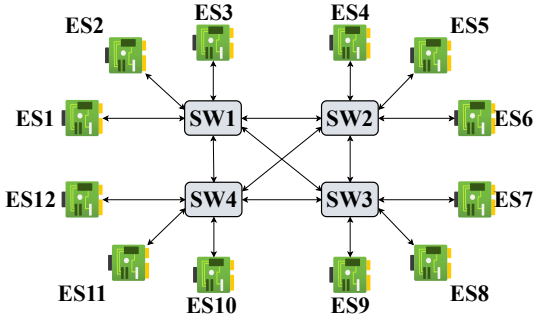

Figure 16: Medium-mesh topology used to evaluate open-ended questions in TSNBench.

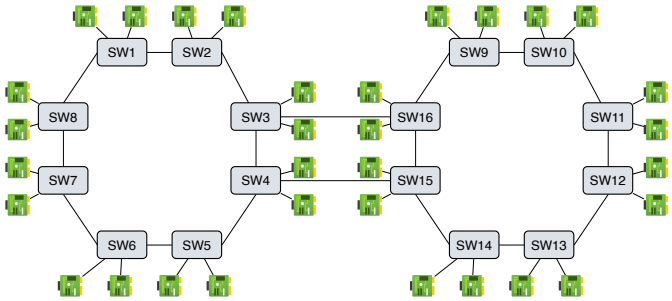

Figure 17: Ring topology representing the industrial ring network used to evaluate open-ended questions in TSNBench.

Flow parameters: We show the flow information used in TSNBench as follows.

Flow Information *TC1_flows.txt*

```
0,node2_1,node5_2,2500,709,965
1,node5_4,node3_2,2500,610,825
2,node0_4,node0_1,1000,786,887
3,node2_3,node4_3,2500,1088,1233
4,node0_4,node3_3,1000,1015,488
5,node0_4,node0_1,2500,926,501
...
```

Ground Truth WCD Values The ground-truth WCD values of the flows for all open-ended test cases for the CBS mechanism are calculated using a verified NC tool [Zhao et al., 2018, Debnath et al., 2025c, Gavriluţ and Pop, 2020]. For the WCD of the CQF mechanism, we use the mathematical equation given in Eq. 22.

12 More on TSNBench MCQA Evaluation

We evaluate both open-source and closed-source state-of-the-art LLMs on TSNBench. A detailed list of the models, along with their model numbers and snapshots, is given in Table 6. This ensures that the results are reproducible by the community.

Table 6: Details of the models used for the benchmarking on TSN. Both MCQA and open-end questions are evaluated on these models. We provide the specific model number and snapshot for reproducibility.

Chat Models				
Model	Family	Model ID	Organization	Country
Grok 4.1 Fast	Grok	grok-4-1-fast-reasoning	xAI	USA
Grok 4.1 Fast (Non-Reasoning)	Grok	grok-4-1-fast-non-reasoning	xAI	USA
DeepSeek-V3.2 (Non-thinking Mode)	DeepSeek	deepseek-chat	DeepSeek AI	China
GPT-4o	GPT	gpt-4o-2024-08-06	OpenAI	USA
GPT-4o mini	GPT	gpt-4o-mini-2024-07-18	OpenAI	USA
Llama 3.3	Llama	Llama-3.3-70B-Instruct	Meta (via HF)	USA
Mistral Medium 3.1	Mistral	mistral-medium-2508	Mistral AI	France
Mistral Large 3	Mistral	mistral-large-2512	Mistral AI	France
Reasoning/Thinking Models				
Claude Sonnet 4.5	Claude	claude-sonnet-4-5-20250929	Anthropic	USA
o3	GPT	o3-2025-04-16	OpenAI	USA
GPT-5	GPT	gpt-5-2025-08-07	OpenAI	USA
DeepSeek-V3.2 (Thinking Mode)	DeepSeek	deepseek-reasoner	DeepSeek AI	China
Gemini 2.5 Flash	Gemini	gemini-2.5-flash	Google	USA
Small Models				
Llama 3.2 1B	Llama	llama-3.2-1B	Meta (via HF)	USA
Qwen3 8B	QwenLM	Qwen3-8B	Alibaba Cloud	China
Ministral 3 8B	Ministral	ministral-8b-2512	Mistral AI	France

12.1 Extended Experimental Evaluation

We evaluate the models under two different configurations: (i) default temperature settings (0.7) and (ii) temperature set to 0.0, for both MCQA and open-ended questions. As in safety-critical networks, we want to ensure deterministic results. Therefore, we evaluate whether LLMs can provide consistent results when the temperature is set to 0.0. For models that do not support the temperature parameter, we use the default temperature for evaluation.

Table 7 provides the accuracy and average consistency of the models for the MCQA dataset under the default temperature and temperature set to 0.0. Average consistency represents the ability of the model to provide the same results across three runs.

Table 7: Extended evaluation results of TSNBench MCQA dataset across different state-of-the-art models across different families. We provide the accuracy in percentage under two different temperature setting (default and set to 0.0). The consistency shows the performance of the model in providing the same response across three runs. For those models which do not support temperature = 0.0, we use their default temperature and this is marked next to the model in the table.

Model	MCQA Accuracy (%)			Average Consistency (%)		
	Default Temp.	Temp=0.0	Temp=0.7	Default Temp.	Temp=0.0	Temp=0.7
Grok 4.1 Fast†	**93.2**	–	–	0.99	–	–
Grok 4.1 Fast (Non-Reasoning)	–	91.7	91.6	–	1.00	1.00
DeepSeek-V3.2 (Non-thinking)	–	94.0	93.4	–	1.00	0.98
GPT-4o	–	91.8	92.1	–	1.00	0.98
GPT-4o mini	–	88.3	88.2	–	0.99	0.98
Llama 3.3	–	88.9	89.1	–	0.99	0.99
Mistral Medium 3.1	–	92.1	92.3	–	1.00	0.99
Mistral Large 3	–	92.8	92.9	–	1.00	1.00
Claude Sonnet 4.5	–	**95.3**	**95.3**	–	1.00	1.00
o3†	94.7	–	–	**0.98**	–	–
GPT-5†	**95.0**	–	–	0.99	–	–
DeepSeek-V3.2 (Thinking)†	94.7	–	–	**0.98**	–	–
Gemini 2.5 Flash	–	90.1	90.8	–	**0.98**	0.97
Llama 3.2 1B	–	**67.4**	**67.0**	–	1.00	**0.93**
Qwen3 8B	–	83.7	82.8	–	0.99	0.97
Ministral 3 8B	–	86.9	86.5	–	1.00	0.97

† Temperature parameter not supported. Evaluated with default settings.

12.2 Cost and Latency

The cost and latency of a model are important evaluation parameters for the research community. Spending a large amount of money on benchmark evaluation is a real bottleneck for research groups. Moreover, not all models can be evaluated locally. Table 8 presents the cost and latency of the TSNBench MCQA and open-ended questions. Evaluating MCQA is relatively much cheaper than evaluating open-ended questions.

Table 8: Extended results of cost and latency comparison for MCQA and open-ended evaluation in TSNBench. "–" indicates that cost and latency are not reported for this model, as it successfully evaluated fewer than 50 out of 100 TCs, where a TC is considered successfully evaluated only if the model provided WCD estimates for at least 80% of the flows within that TC.

Model	MCQA		CBS Open-ended questions		CQF Open-ended questions	
	Cost (USD)	Latency (ms)	Cost (USD)	Latency (ms)	Cost (USD)	Latency (ms)
Grok 4.1 Fast†	0.2490	18,769,322	**0.2241**	43,788,625	0.3047	49,367,058
Grok 4.1 Fast (Non-Reasoning)	0.2612	1,450,175	0.3256	3,200,483	0.3251	3,383,209
DeepSeek-V3.2 (Non-thinking)	**0.0420**	2,264,129	–	–	0.4816	13,211,941
GPT-4o	2.3661	2,053,601	–	–	4.3866	**2,122,167**
GPT-4o mini	0.1417	2,251,786	0.3438	7,083,613	0.3642	7,092,033
Llama 3.3	0.5224	1,028,334	0.7399	**1,264,847**	0.7137	1,136,920
Mistral Medium 3.1	0.4432	1,839,587	1.9918	6,335,803	2.2442	6,658,056
Mistral Large 3	0.4868	15,487,495	1.5989	11,149,853	1.2298	8,226,564
Claude Sonnet 4.5	3.5967	5,190,222	21.1719	14,342,470	18.3093	12,583,398
o3†	7.6293	10,831,712	10.9954	10,127,407	10.0134	8,591,382
GPT-5†	**12.8766**	15,860,682	**57.5232**	**74,473,434**	**42.4470**	**58,404,966**
DeepSeek-V3.2 (Thinking)†	0.4069	12,385,365	–	–	–	–
Gemini 2.5 Flash	0.4164	18,732,689	2.6471	24,160,941	2.7690	17,688,716
Llama 3.2 1B	0.0864	1,883,306	–	–	–	–
Qwen3 8B	0.3736	**42,272,121**	–	–	–	–
Ministral 3 8B	0.1237	**972,292**	0.2425	10,122,749	**0.2434**	9,603,062

† Temperature parameter not supported. Evaluated with default settings.

13 More on TSNBench Open-Ended Questions Evaluation

We provide MAE and MAPE evaluations for the open-ended questions. A sample calculation is given as follows:

MAE and MAPE calculation example: Consider a model evaluated on three test cases (TCs). These three TCs may have different topologies, different flows and flow parameters, and different routes. For each TC, we have the ground-truth and predicted WCD values shown in Table 9. The ground truth is calculated using an NC solver for CBS and a mathematical equation for CQF.

Table 9: Sample example of test cases (TC) with ground truth, predicted and absolute error values.

TC	Flow	Ground Truth (μs)	Predicted (μs)	Abs. Error (μs)
TC1	F0	200	212	12
TC1	F1	150	180	30
TC1	F2	500	490	10
TC2	F0	100	108	8
TC2	F1	300	255	45
TC3	F0	400	420	20
TC3	F1	250	265	15
TC3	F2	600	600	0

Per-TC MAE: Suppose TC1, TC2, and TC3 contain three, two, and three flows, respectively.

$$\{f_1, f_2, f_3\} \in TC1;$$
$$\{f_1, f_2\} \in TC2;$$
$$\{f_1, f_2, f_3\} \in TC3;$$

Let $\Gamma(f_0)$ denote the absolute error of flow f_0 in TC1, $\beta(f_0)$ denote the predicted WCD of flow f_0 given by the LLM model, and $\Omega(f_0)$ denote the ground truth of flow f_0. We calculate $\Gamma(f_0)$ as follows:

$$\Gamma(f_0) = |\beta(f_0) - \Omega(f_0)|$$

In the given example, let $\Gamma(f_0)$ = 12, $\Gamma(f_1)$ = 30, and $\Gamma(f_2)$ = 10 for TC1. Similarly, for TC2, $\Gamma(f_0)$ = 8 and $\Gamma(f_1)$ = 45 and for TC3, $\Gamma(f_0)$ = 20, $\Gamma(f_1)$ = 15, and $\Gamma(f_2)$ = 0. We calculate the MAE for TC1, TC2, and TC3 represented as MAE_{TC1}, MAE_{TC2}, and MAE_{TC3} as follows:

$$\text{MAE}_{\text{TC1}} = (12 + 30 + 10)/3 = 17.3\ \mu\text{s}$$
$$\text{MAE}_{\text{TC2}} = (8 + 45)/2 = 26.5\ \mu\text{s}$$
$$\text{MAE}_{\text{TC3}} = (20 + 15 + 0)/3 = 11.7\ \mu\text{s}$$

For every model, we have 100 test cases, and the final MAE is averaged across all test cases (in this example 3 test cases) and is represented as:

$$\text{MAE} = (17.3 + 26.5 + 11.7)/3 = 18.5\ \mu\text{s}$$

The per-flow MAPE denoted as $\alpha(f_0)$ is calculated as follows:

$$\alpha(f_0) = \frac{|\beta(f_0) - \Omega(f_0)|}{\Omega(f_0)} \times 100$$

For TC1, we calculate the MAPE as follows:

$$\text{MAPE}_{\text{TC1}} = \frac{\alpha(f_0) + \alpha(f_1) + \alpha(f_2)}{3} = 8.7\%$$

Similarly, the MAPE for TC2 and TC3 is given as follows:

$$\text{MAPE}_{\text{TC2}} = 11.5\%$$
$$\text{MAPE}_{\text{TC3}} = 3.7\%$$

The final MAPE for each model is averaged across the 3 test cases:

$$\text{MAPE} = (8.7 + 11.5 + 3.7)/3 = 8.0\%$$

Table 10: MAE (μs) for **CBS** open-ended evaluation across **One-Switch topology** test cases. "–" denotes invalid, missing, or partial response (model predicted fewer than 80% of flows in the one TC). "**0**" denotes trivial failure (model returned all-zero WCD values). Best result per TC shown in **bold**. The MAE (μs) reported in this table is based on average across three runs per TC.

Model	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	TC9	TC10	TC11
Grok 4.1 Fast	**51.93**	91.48	175.11	31.03	16.04	209.24	81.05	–	136.0	172.41	167.5
Grok 4.1 Fast (Non-Reasoning)	–	–	–	–	–	–	–	–	–	–	–
DeepSeek-V3.2 (Non-Thinking)	147.46	698.76	291.06	–	130.71	–	–	–	237.33	432.63	–
GPT-4o	505.94	486.53	294.39	85.41	124.37	265.02	204.64	63.72	237.22	510.61	415.26
GPT-4o mini	–	–	293.06	133.41	–	301.19	210.14	–	242.67	–	–
Llama 3.3 70B	516.89	322.23	301.96	140.34	43.2	300.79	213.14	58.87	244.67	391.42	427.25
Mistral Medium 3.1	509.17	488.48	294.34	130.85	35.19	295.19	205.13	70.37	240.64	510.08	418.62
Mistral Large 3	397.0	381.23	172.06	20.79	16.35	183.52	76.58	181.61	127.67	403.18	302.1
Claude Sonnet 4.5	499.76	484.85	286.34	109.67	122.86	267.11	202.77	61.65	216.11	499.51	413.98
o3	**61.05**	121.41	**10.57**	17.5	105.48	84.57	143.37	**19.64**	**125.73**	171.22	
GPT-5	178.03	**86.44**	30.66	**15.86**	**50.59**	**40.84**	14.55	23.66	136.91	225.17	
DeepSeek-V3.2 (Thinking)	–	–	–	–	–	–	–	–	–	–	–
Gemini 2.5 Flash	–	225.61	220.86	61.78	57.02	135.3	84.65	39.35	164.93	334.01	**158.82**
Small Models											
Llama 3.2 1B	–	–	–	–	–	–	–	–	–	–	–
Qwen3 8B	–	–	–	–	–	–	–	–	–	–	–
Ministral 3 8B	778156.77	1467.11	652.98	858.59	863.29	1360.14	1183.86	**10.08**	882.67	979.38	905.23

Table 11: MAE (μs) for **CQF** open-ended evaluation across **One-Switch topology** test cases. "–" denotes invalid, missing, or partial response (model predicted fewer than 80% of flows in the one TC). "**0**" denotes trivial failure (model returned all-zero WCD values). Best result per TC shown in **bold**. The MAE (μs) reported in this table is based on average across three runs per TC.

Model	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	TC9	TC10	TC11
Grok 4.1 Fast	177.2	112.99	54.6	46.64	28.29	66.95	–	52.89	150.67	288.03	128.18
Grok 4.1 Fast (Non-Reasoning)	95.0	93.67	71.09	91.0	36.11	94.5	91.0	95.0	93.67	–	91.0
DeepSeek-V3.2 (Non-Thinking)	38.38	95.0	198.24	45.78	41.0	95.0	95.0	57.0	89.0	–	149.89
GPT-4o	316.33	283.33	633.0	**0.0**	**1.0**	32.33	313.67	15.67	949.0	317.0	283.67
GPT-4o mini	88.33	85.67	93.0	91.0	97.0	98.33	97.0	29.67	93.0	91.0	95.44
Llama 3.3 70B	67.72	97.72	29.26	66.82	99.33	37.74	57.94	90.25	717.31	171.29	109.71
Mistral Medium 3.1	801.11	93.0	91.0	90.33	91.0	93.0	91.17	58.94	93.0	92.67	502.0
Mistral Large 3	100.0	84.0	101.0	82.67	101.0	100.0	101.0	101.67	84.0	50.0	78.11
Claude Sonnet 4.5	87.43	**44.48**	**11.68**	46.29	47.44	43.38	**30.28**	48.89	46.54	**13.69**	**31.28**
o3	**34.33**	159.11	146.89	68.99	32.78	69.33	32.4	70.37	47.0	126.11	81.15
GPT-5	169.3	162.19	85.82	74.85	27.11	76.64	54.51	25.58	108.33	160.43	115.66
DeepSeek-V3.2 (Thinking)	–	–	–	–	–	–	–	–	–	–	–
Gemini 2.5 Flash	80.68	91.67	79.79	5.34	15.68	**21.28**	49.73	**8.42**	**44.36**	157.85	80.49
Small Models											
Llama 3.2 1B	–	–	–	–	–	–	–	–	–	–	–
Qwen3 8B	–	–	–	–	–	–	–	–	–	–	–
Ministral 3 8B	1216.33	2279.78	362.33	1053.89	1177.67	765.22	80.56	593.44	722.67	–	2119.22

Table 12: MAE (μs) for CBS open-ended evaluation across Ring topology test cases (TC1-TC20), taken from 100 total test cases spanning three topologies. "–" denotes invalid, missing, or partial response (model predicted fewer than 80% of flows in the one TC). "**0**" denotes trivial failure (model returned all-zero WCD values). Best result per TC shown in **bold**. The MAE (μs) reported in this table is based on average across three runs per TC.

Model	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	TC9	TC10	TC11	TC12	TC13	TC14	TC15	TC16	TC17	TC18	TC19	TC20
Grok 4.1 Fast	–	–	–	–	–	–	–	0	–	–	–	–	–	–	0	–	–	–	–	–
Grok 4.1 Fast (Non-Reasoning)	353.37	2107.44	1346.37	947.7	1088.22	8750.11	3031.05	4794.74	212.79	2154.81	449.66	10680.41	317.1	1815.54	215.52	264.96	1491.62	1992.75	686.01	559.0
DeepSeek-V3.2 (Non-Thinking)	0	0	0	0	0	0	0	0	–	–	0	0	0	0	0	0	0	0	–	0
GPT-4o	0	0	0	0	0	0	0	0	–	0	0	0	0	0	0	0	0	0	–	0
GPT-4o mini	575.21	1021.42	1010.98	974.13	435.05	602.95	680.03	763.06	409.39	330.35	538.01	0	339.8	366.84	343.9	216.14	513.14	509.49	713.7	828.38
Llama 3.3 70B	537.49	0	834.68	940.46	347.3	568.77	531.01	688.05	409.44	292.29	282.04	271.79	0	–	–	–	399.24	503.22	–	448.82
Mistral Medium 3.1	222.92	894.74	**236.19**	477.27	394.88	449.45	231.57	283.69	183.42	295.81	**248.21**	808.44	910.92	653.12	889.7	709.57	498.84	781.82	690.99	**229.13**
Mistral Large 3	511.74	–	919.4	–	375.99	–	–	693.06	312.21	–	456.66	271.54	–	–	–	–	404.25	–	–	–
Claude Sonnet 4.5	530.12	941.39	938.51	880.56	279.59	400.01	595.62	393.4	315.46	251.47	400.11	317.76	266.55	295.4	289.41	171.74	115.43	399.78	616.89	733.79
o3	250.72	474.97	859.62	**380.65**	**225.1**	172.92	**225.86**	434.22	73.84	**124.12**	260.18	92.34	**91.49**	194.26	152.06	172.15	**91.48**	111.08	429.72	784.25
GPT-5	**110.06**	**132.15**	754.56	541.75	260.18	**113.28**	254.26	**245.89**	**60.51**	303.68	–	164.84	156.55	**91.19**	**107.45**	**167.67**	257.56	172.03	464.36	305.86
DeepSeek-V3.2 (Thinking)	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Gemini 2.5 Flash	–	–	750.1	448.42	444.25	536.47	898.0	608.64	–	237.37	362.35	**84.77**	350.92	276.76	347.04	255.09	234.62	**91.7**	**238.5**	679.01
Small Models																				
Llama 3.2 1B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	–	0	0	0	–
Qwen3 8B	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Ministral 3 8B	3571.31	413.05	318.9	–	826.43	486.78	–	460.25	581.21	323.18	274.1	976.95	–	898.99	–	–	787.18	740.15	293.11	–

Table 13: MAE (μs) for CQF open-ended evaluation across Ring topology test cases (TC1-TC20), selected from 100 total test cases spanning three topologies. "–" denotes invalid, missing, or partial response (model predicted fewer than 80% of flows in the one TC). "**0**" denotes trivial failure (model returned all-zero WCD values). Best result per TC shown in **bold**. The MAE (μs) reported in this table is based on average across three runs per TC.

Model	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	TC9	TC10	TC11	TC12	TC13	TC14	TC15	TC16	TC17	TC18	TC19	TC20
Grok 4.1 Fast	137.1	179.01	55.82	27.44	20.53	–	–	47.88	57.49	13.89	169.51	71.4	169.18	141.5	151.1	–	–	175.44	201.77	301.2
Grok 4.1 Fast (Non-Reasoning)	141.15	166.57	172.65	152.08	173.0	177.0	166.62	178.33	209.56	204.81	202.16	185.33	204.27	198.7	220.07	185.43	216.04	184.53	230.49	220.4
DeepSeek-V3.2 (Non-Thinking)	237.15	173.33	183.33	164.15	7.0	199.16	183.67	0	218.6	212.31	213.13	197.46	219.53	205.5	237.87	197.75	233.42	193.73	237.82	228.17
GPT-4o	8.08	**0.57**	**2.57**	**0.45**	**0.78**	29.18	**1.51**	**0.41**	223.13	**0.52**	144.23	**5.14**	**1.0**	**1.75**	**1.4**	**5.4**	**5.07**	**1.0**	66.8	14.04
GPT-4o mini	147.23	175.3	173.17	160.0	195.33	190.38	175.24	185.53	222.29	200.1	211.15	186.95	208.8	208.87	226.62	198.73	223.73	190.13	239.33	228.17
Llama 3.3 70B	148.92	–	173.41	–	196.0	–	–	203.84	211.58	–	208.08	187.3	–	–	–	–	227.18	–	–	–
Mistral Medium 3.1	175.03	173.92	158.68	157.65	149.24	143.71	132.14	124.24	203.6	195.5	202.84	138.42	116.09	113.37	161.92	156.62	139.56	117.54	223.69	191.39
Mistral Large 3	26.15	–	43.14	–	45.0	–	–	44.84	**16.27**	–	**1.0**	68.67	–	–	–	–	44.27	–	–	–
Claude Sonnet 4.5	**5.71**	135.05	72.06	122.75	55.69	**3.12**	123.14	166.63	96.75	181.58	102.01	44.4	164.4	157.96	209.06	16.1	17.23	3.69	127.51	**7.45**
o3	115.13	148.83	107.3	58.18	52.29	207.42	105.08	101.71	99.13	80.73	81.12	77.46	48.74	98.05	192.89	74.34	82.77	166.86	**50.62**	198.67
GPT-5	162.28	110.45	198.84	98.48	126.95	114.18	93.79	92.27	210.55	8.97	154.81	153.82	27.24	121.55	40.93	27.79	117.35	107.33	166.49	129.34
DeepSeek-V3.2 (Thinking)	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Gemini 2.5 Flash	122.97	51.27	73.69	28.2	32.12	15.17	93.29	78.73	165.45	16.18	15.06	26.55	69.58	76.32	123.55	38.22	126.44	136.07	231.22	215.35
Small Models																				
Llama 3.2 1B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	–
Qwen3 8B	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Ministral 3 8B	4348.12	1196.35	6907.14	714.58	5215.69	905.0	768.1	9065.2	121.13	206.04	785.87	811.05	–	1641.37	2440.85	852.55	6831.6	1151.91	237.98	4420.84

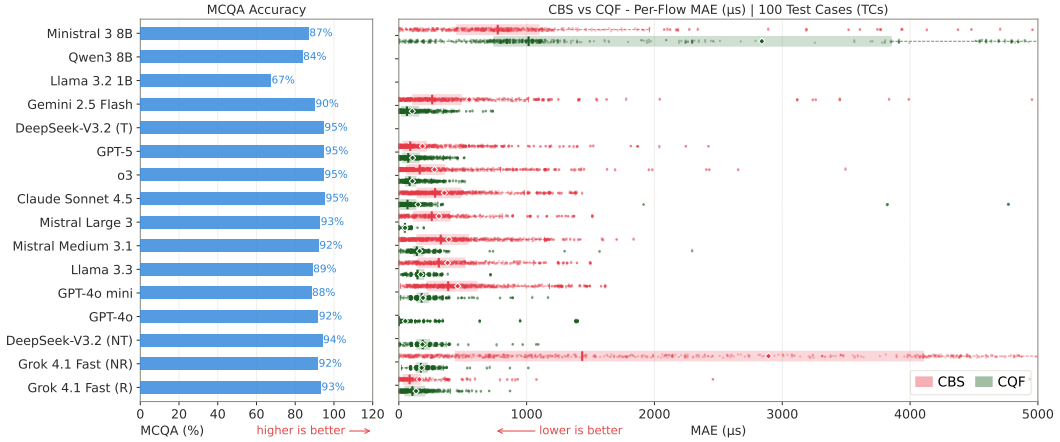

Figure 18: Performance comparison across MCQA and open-ended WCD computation for all 16 evaluated models in TSNBench, illustrating the dissociation between declarative knowledge and computational reasoning. (*Left*) MCQA accuracy (%) per model. (*Right*) Per-TC MAE distribution (in μs) for CBS and CQF open-ended questions, shown as box plots over 100 total evaluated test cases, aggregated across three independent runs. Models achieving above 90% MCQA accuracy exhibit substantially high MAE on open-ended WCD computation.

In TSNBench, all test cases contributes equally towards the model performance irrespective of the number of flows in the network. As per the network architecture, all flows are equally critical and needs the same preference. This ensures that for each network scenario all the flows are weighted equally.

Table 14: CBS Error Analysis Case 1: Lack of Specific Knowledge.

Test Case: TC1
TSN mechanism: CBS

You are an expert Time-Sensitive Networking (TSN) orchestrator. Your task is to calculate the worst case delay (WCD) for each TSN flow.

Input:
Network Topology ($TC1\_topo.txt$)
Flow Information ($TC1\_flows.txt$)
Routing of the Flow ($TC1\_route.txt$)

Constant:
Bandwidth link = 100 Mbps; Propagation delay = 1 μs; Switching delay = 1 μs;
Time synchronization error = 1 μs; The switches of the network are cut-through switches.

TSN Mechanism:
Only Credit-Based Shaper (CBS, IEEE 802.1Qav) is allowed;
All flows are AVB Class A, PCP = 6, using queue 6 only.

Task:

1. Map each egress port's queues and collect the set of flows traversing from that port, using the given topology, flows, and route of the flow.
2. For each egress port, use the given IdleSlope and then compute the SendSlope.
3. For each flow, construct an arrival curve from its frame size and periodicity.
4. For each port, derive a lower-bounded CBS service curve.
5. Calculate the worst case delay (WCD) in microseconds (μs) for each flow using Network Calculus method.
6. Provide the confidence score between 0.0 and 1.0 from your answers. 1.0 means mathematically or procedurally provable from given info with zero ambiguity. 0.0 means zero confidence.

Table 14: CBS Error Analysis Case 1: Lack of Specific Knowledge. (continued)

Grok 4.1 Fast (Non-Reasoning) output:

F0: 1452.0, F1: 1124.0, F2: 678.0, F3: 1234.0, F4: 1567.0, F5: 689.0, F6: 456.0, F7: 789.0, F8: 567.0, F9: 589.0, F10: 678.0, F11: 645.0, F12: 1123.0, F13: 789.0, F14: 456.0, F15: 678.0, F16: 789.0, F17: 1567.0, F18: 456.0, F19: 1234.0

Model Reasoning and Explanation Output:

Parameters used:
send_slope: $75\% \times 100$ Mbps $= 75$ Mbps $= 75000$ bits/μs
frame_overhead: 24 bytes (Ethernet header + FCS),
mtu: 1522 bytes (max frame size),
queue: 6 (PCP=6, AVB Class A),
cbs_credit_max: MTU $\times$ 8 $= 12176$ bits

WCD equation given by the model:

$$\alpha(t) = \frac{L_{\max}}{\text{Period}} + \frac{t}{\text{Period}} \quad \text{(arrival curve)}$$

$$\beta(t) = R_{\text{send}} \times t - C \quad \text{(CBS service curve)}$$

$$WCD = h \times (\text{prop\_delay} + \text{switch\_delay} + \text{sync\_error}) + \max(\alpha \cap \beta)$$

$$\text{SendSlope} = \text{IdleSlope} \times \text{Bandwidth} = 0.75 \times 100 \text{ Mbps}$$

$$R_{\text{send}} = \text{SendSlope} = 75 \text{ Mbps}$$

$$C = \text{credit}_{\max} = \text{MTU} \times 8 \text{ bits}$$

$$h = \text{hop count from path}$$

Per-flow WCD:

$$WCD_{F_i} = \sum_{h \in \text{path}(F_i)} \left(d^h_{\text{prop}} + d^h_{\text{switch}} + d^h_{\text{sync}}\right) + \max_{t \geq 0} \left(\alpha_{F_i}(t) \cap \beta_{\text{port}}(t)\right)$$

Table 14: CBS Error Analysis Case 1: Lack of Specific Knowledge. (continued)

Run 2 of the model (Grok 4.1 Fast (Non-Reasoning)):

WCD equation given by the model during run 2:

$$\begin{aligned}
\text{Frame size:} \quad & L_i = \text{payload_}i + 24 \text{ bytes}, \\
\text{Transmission time:} \quad & T_{x_i} = \frac{8L_i}{R}, \\
\text{Arrival curve:} \quad & \alpha_i(t) = \frac{L_i}{p_i} \cdot t, \\
\text{CBS service curve:} \quad & \beta(t) = \min\left\{\alpha_{h_i}(t),\ \text{sendSlope} \cdot t + \text{credit}_{\max}\right\}, \\
\text{Worst-case delay:} \quad & WCD_i = \max_{h \in \text{path}} \sum \left(T_{\text{prop},h} + T_{\text{switch},h} + T_{\text{queue},h}\right) + T_{\text{sync}}, \\
\text{Queueing delay:} \quad & T_{\text{queue},h} = \frac{L_i}{\alpha C} + \text{burst_interference}, \\
\text{Send slope:} \quad & \text{sendSlope} = \text{idleSlope} \cdot R - \text{idleSlope} \cdot R = \alpha R(1 - 2\alpha);
\end{aligned}$$

Per-flow WCD:

$$WCD_i = \text{hop\_count}_i \cdot (1 + 1) + \text{max\_queue\_delay} + 1;$$

Expert Explanation:

The model uses different equations across different runs even with temperature set to 0.0. When analyzing the first run, the model makes several fundamental errors in evaluating the interference/blocking experienced by the queue under analysis, i.e., both the arrival curve and CBS service curve.
First, it assumes the wrong maximum burst size and does not consider the variant output arrival curve at subsequent hops along the path.
Second, it assumes a non-zero maximum credit, which is possible in this scenario due to the non-preemption frame of lower priority of non-CBS traffic. However, the model applies the wrong maximum credit to the CBS service curve, incorrectly understanding the relation of the corresponding rate (75%).
Furthermore, in the final WCD calculation, the model considers only the arrival curve of the frame under analysis. This is incorrect: the proper approach requires using the aggregate arrival curve of all frames transmitted through the queue. It is also not clear whether the $max(\alpha \cap \beta)$ operation represents the maximum horizontal deviation between α and β. The output arrival curve is not considered in the following hops' calculation.
On the other hand, the second run exhibits even more severe hallucinations, producing calculations of the wrong arrival curve without considering any burst size, and the wrong CBS service curve considering the wrong service rate (sendSlope) and wrong relation between the corresponding rate and the maximum credit, as well as without considering the output arrival curve either etc.

GPT-5 output:

F0: 714.65, F1: 821.79, F2: 171.48, F3: 687.17, F4: 907.26, F5: 171.48, F6: 183.9, F7: 453.91, F8: 611.08, F9: 181.3, F10: 611.08, F11: 321.35, F12: 1124.12, F13: 483.67, F14: 172.43, F15: 489.26, F16: 521.33, F17: 918.73, F18: 181.3, F19: 624.12

Table 14: CBS Error Analysis Case 1: Lack of Specific Knowledge. (continued)

Model Reasoning and Explanation Output:

$$\text{sendSlope} = -25 \text{ Mbps},$$

$$\text{Arrival curve:} \quad b = \sum_{\forall \text{F}_\text{i} \mid \text{ share output port}} L_i \text{ bits},$$

$$r = \sum_{\forall \text{F}_\text{i} \mid \text{ share output port}} \frac{L_i}{p_i} \text{ bits}/\mu\text{s},$$

$$\alpha(t) = b + rt,$$

$$\begin{aligned} T_{\text{tech,total}} &= T_{\text{prop}} + T_{\text{switching}} + T_{\text{sync}} \\ &= hops \times 1\ \mu\text{s} + (hops - 1) \times 1\ \mu\text{s} + (hops - 1) \times 1\ \mu\text{s} = 13\ \mu\text{s}, \end{aligned}$$

$$\begin{aligned} WCD_{\text{total}} &= WCD_{\text{sum}} + T_{\text{tech,total}} \\ &= 701.65\ \mu\text{s} + 13\ \mu\text{s} = 714.65\ \mu\text{s}. \end{aligned}$$

Expert Explanation:

GPT uses the correct arrival and service curves. Its main error is that it fails to include the output port (output arrival curves) in the analysis.

Ministral 3 8B output:

F0: 1250.0, F1: 1100.0, F2: 500.0, F3: 1500.0, F4: 600.0, F5: 1100.0, F6: 2500.0, F7: 1200.0, F8: 500.0, F9: 500.0, F10: 2000.0, F11: 500.0, F12: 1000.0, F13: 1000.0, F14: 2000.0, F15: 500.0, F16: 1500.0, F17: 1000.0, F18: 500.0, F19: 2000.0

Model Reasoning and Explanation Output:

$$\text{Bandwidth:} \quad R = 100 \text{ Mbps} = 12.5 \text{ MB/s},$$

$$\text{Propagation delay:} \quad T_{\text{prop}} = 1\ \mu\text{s} \quad \text{per hop},$$

$$\text{Switching delay:} \quad T_{\text{switch}} = 1\ \mu\text{s} \quad \text{per switch},$$

$$\text{Time synchronization error:} \quad T_{\text{sync}} = 1\ \mu\text{s},$$

$$\text{IdleSlope:} \quad \text{IdleSlope} = 0.75R = 75 \text{ Mbps} = 9.375 \text{ MB/s},$$

$$\text{Cut-through switch:} \quad T_{\text{buffer}} = 0,$$

$$\text{Payload overhead:} \quad L_{\text{overhead}} = 18 \text{ B} + 4 \text{ B} = 22 \text{ B},$$

$$\text{Maximum frame size:} \quad L_{\text{max}} = 1433 \text{ B} + 22 \text{ B} = 1455 \text{ B},$$

$$\text{Maximum transmission time:} \quad T_{\text{tx,max}} = \frac{1455 \times 8}{100 \text{ Mbps}} = 9.24\ \mu\text{s}.$$

Table 14: CBS Error Analysis Case 1: Lack of Specific Knowledge. (continued)

$$\begin{aligned}\text{SendSlope} &= \text{IdleSlope}\,(1 - \text{IdleSlope}) \\ &= 9.375\ \text{MB/s} \times 0.25 \\ &= 2.34375\ \text{MB/s},\end{aligned}$$

Burst size: $B_{\text{burst},i} = \text{SendSlope} \cdot p_i,$

CBS service curve: $S(t) = \text{SendSlope}\left(t - \dfrac{B_{\text{burst}}}{\text{SendSlope}}\right), \quad t \geq \dfrac{B_{\text{burst}}}{\text{SendSlope}}.$

WCD Calculation:

Arrival curve for flow F_x : $R(t) = \min\{B_x,\ S_x t\},$

$$B_x = L_{\text{payload},x} + L_{\text{overhead}},$$

$$S_x = \frac{8L_{\text{payload},x}}{p_x},$$

Service curve for each port: $S(t) = \text{SendSlope}\left(t - \dfrac{B_{\text{burst}}}{\text{SendSlope}}\right), \quad t \geq \dfrac{B_{\text{burst}}}{\text{SendSlope}},$

$$\begin{aligned}WCD = \max\left(0,\ \frac{B_{\text{burst}} - \text{SendSlope} \cdot Period}{\text{SendSlope}}\right) \\ + T_{\text{prop}} + T_{\text{switch}} + T_{\text{sync}}.\end{aligned}$$

Expert Explanation:

Even though it is a small model, this agent delivers a poor result. Similar to what was observed with Grok 4.1 Fast (Non-Reasoning), it fails to account for the arrival curves of flows and the wrong CBS service curve.
In addition, it analyzes only a single queue, instead of computing the delay at the output port of each device along the end-to-end path.
Moreover, it introduces unit inconsistencies such as dividing the frame size expressed in bytes by a link speed given in bits/μs, which leads to incorrect results.

Table 15: CQF Error Analysis Case 1: Lack of Specific Knowledge.

Test Case: TC1
TSN mechanism: CQF

You are an expert Time-Sensitive Networking (TSN) orchestrator. Your task is to calculate the worst case delay (WCD) for each TSN flow.

Input:
Network Topology ($TC1\_topo.txt$)
Flow Information ($TC1\_flows.txt$)
Routing of the Flow ($TC1\_route.txt$)

Constant:
Bandwidth link = 100 Mbps; Propagation delay = 1 μs; Switching delay = 1 μs;
Time synchronization error = 1 μs; The switches of the network are cut-through switches.

TSN Mechanism:
Only Cyclic Queuing and Forwarding (CQF, IEEE 802.1Qch) is allowed;
All flows are TT, PCP = 7, using queue 7 (odd) and 6 (even) only.

Task:

1. Map each egress port's queues and collect the set of flows traversing that port, using the given topology, flows, and route of the flow.
2. For the entire network, use the given cycle duration and compute the Hypercycle.
3. For each flow, set the offset or the start time of the flow from the sending node as 0.
4. Calculate the worst case delay (WCD) in microseconds (μs) for each flow.
5. Provide the confidence score between 0.0 and 1.0 from your answers. 1.0 means mathematically or procedurally provable from given info with zero ambiguity. 0.0 means zero confidence.

Table 15: CQF Error Analysis Case 1: Lack of Specific Knowledge. (continued)

Claude Sonnet's output:

F0: 257.72, F1: 206.8, F2: 105.096, F3: 218.704, F4: 253.904, F5: 104.008, F6: 103.808, F7: 153.52, F8: 119.408, F9: 119.752, F10: 122.896, F11: 118.464, F12: 222.04, F13: 156.552, F14: 103.808, F15: 156.576, F16: 155.68, F17: 256.616, F18: 101.696, F19: 208.328

Model Reasoning and Explanation Output:

$$\begin{aligned} &\text{Cycle duration: } = 50\ \mu\text{s}, \\ \text{LCM of all periods and cycle:}\quad &\text{LCM}(1000, 2500, 5000, 50) = 5000\ \mu\text{s}, \\ &\text{Hypercycle: } = 5000\ \mu\text{s}. \end{aligned}$$

$$\text{WCD equation:}\quad WCD = \sum_{\text{hops}} (T_{\text{trans}} + T_{\text{prop}} + T_{\text{sw}} + T_{\text{sync}}) + T_{\text{buffering}}.$$

$$\begin{aligned} T_{\text{trans}} &= \frac{(\text{Payload} + \text{Ethernet\_overhead}) \cdot 8}{R}, \\ T_{\text{prop}} &= 1\ \mu\text{s} \quad \text{per link}, \\ T_{\text{sw}} &= 1\ \mu\text{s} \quad \text{per switch}, \\ T_{\text{sync}} &= 1\ \mu\text{s} \quad \text{per hop}, \\ T_{\text{buffering}} &= (\text{hop\_count} - 1) \cdot T_{\text{cycle}} \end{aligned}$$

In CQF, frames may wait up to one cycle duration at each intermediate switch if they arrive just after the transmission window closes. For cut-through with CQF, worst case adds $(N-1) * \text{cycle\_duration}$ where N is hop count.

$$\begin{aligned} \text{Hop count} &= 5, \\ \text{Payload\_bytes} &= 965\ \text{bytes}, \\ \text{Frame\_size\_bytes} &= 1003\ \text{bytes}, \\ \text{Transmission\_delay} &= 80.24\ \mu\text{s}, \\ \text{Total propagation delay} &= 5\ \mu\text{s}, \\ \text{Total switching delay} &= 4\ \mu\text{s}, \\ \text{Total synchronization error} &= 5\ \mu\text{s}, \\ \text{CQF buffering} &= 200\ \mu\text{s}, \\ \text{WCD} &= 257.72\ \mu\text{s}. \end{aligned}$$

Ground truth WCD::

$$\begin{aligned} &\text{Route hop count of } f_1 : 4, T = 50\ \mu\text{s}, \\ WCD_{f_1} &= f_1 \cdot \phi + (SW_{\text{num}} + 1)T + \xi \\ &= (3+1) \times 50 + 5 \\ &= 205\ \mu\text{s}. \end{aligned}$$

Expert Explanation:

The model knows the working mechanism of CQF. However, the model calculated the wrong Hypercycle by including cycle duration along with the periods. Furthermore, the CQF WCD explanation given by the model is wrong.

Table 15: CQF Error Analysis Case 1: Lack of Specific Knowledge. (continued)

GPT-4o output:

F0: 260.0, F1: 210.0, F2: 60.0, F3: 210.0, F4: 260.0, F5: 60.0, F6: 60.0, F7: 110.0, F8: 60.0, F9: 60.0, F10: 60.0, F11: 60.0, F12: 210.0, F13: 110.0, F14: 60.0, F15: 110.0, F16: 110.0, F17: 260.0, F18: 60.0, F19: 210.0

Model Reasoning and Explanation Output:

$$\text{Hypercycle} = 100\ \mu\text{s}.$$

$$\text{WCD} = (\text{hop\_count} \times (\text{cycle\_duration} + T_{\text{switch}} + T_{\text{prop}} + T_{\text{sync}})) + \frac{\text{Payload}}{\text{Bandwidth}}$$

Each flow's path, cycle duration, hypercycle, and CQF parameters are detailed, along with the equation used for WCD calculation. No missing inputs are identified, but assumptions are made based on typical network configurations.

Expert Explanation:

The Hypercycle calculated by the model is wrong, moreover the WCD equation used by the model has an extra wrong component $\frac{\text{Payload}}{\text{Bandwidth}}$.

One of the most interesting finding is all the models provide very different WCD equations across different runs and different test cases.

Ministral 3 8B output:

F0: 1160.0, F1: 1020.0, F2: 100.0, F3: 1160.0, F4: 1260.0, F5: 100.0, F6: 100.0, F7: 100.0, F8: 100.0, F9: 100.0, F10: 100.0, F11: 100.0, F12: 1160.0, F13: 100.0, F14: 100.0, F15: 100.0, F16: 100.0, F17: 1260.0, F18: 100.0, F19: 1160.0

Model Reasoning and Explanation Output:

$$\begin{aligned}
\text{Hypercycle:} \quad H &= 10000\ \mu\text{s}, \\
\text{Max frame size} &= 1526\ \text{bytes} = 1500 + 26\ \text{bytes}, \\
\text{Max transmission time} &= 12.208\ \mu\text{s}, \\
\text{Max queuing delay} &= 48\ \mu\text{s}, \\
WCD &= h\left(T_{\text{prop}} + T_{\text{switch}} + T_{\text{sync}}\right) \\
&\quad + T_{\text{tx,max}} \left\lceil \frac{\text{payload}}{\text{max_frame_size}} \right\rceil \\
&\quad + T_{\text{queue,max}}(h-1),
\end{aligned}$$

Table 15: CQF Error Analysis Case 1: Lack of Specific Knowledge. (continued)

$$\text{Hop count:} \quad h = 5,$$

$$\text{Propagation + switching + sync} = 5 \times (1 + 1 + 1) = 15\ \mu\text{s},$$

$$\text{Transmission time} = 12.208 \times \left\lceil \frac{2500}{1526} \right\rceil = 12.208 \times 2 = 24.416\ \mu\text{s},$$

$$\text{Queueing delay} = 48 \times (5 - 1) = 192\ \mu\text{s},$$

$$\text{Total} = 15 + 24.416 + 192 = 231.416\ \mu\text{s},$$

$$\text{Adjusted worst-case} = 1160\ \mu\text{s}.$$

Expert Explanation:

The WCD equation provided by the model is wrong. Even though the model takes into consideration the number of hops present in the route, the delays accumulated across each hop and also calculates the hop count. However, the model misses the most crucial part of the WCD equation which is the cycle duration. Furthermore, the two components of the WCD equation ($T_{\text{tx,max}} \left\lceil \frac{\text{payload}}{\text{max\_frame\_size}} \right\rceil$) and ($T_{\text{queue,max}}(h-1)$) considered by the model is entirely hallucinated. These two components are mainly contributing to the large WCD values of this model.

14 Failure Mode Analysis

To understand the nature of WCD computation failures, we identify five distinct failure modes observed across models and mechanisms.

Trivial Zero Failure: The model returns WCD = 0 for all flows, producing a structurally valid JSON response but with no computational content. This failure mode affects GPT-4o and DeepSeek-V3.2 (Non-thinking) on CBS, and Llama 3.2 1B across all test cases for CBS and CQF. This suggests these models recognize the output format requirement but cannot engage with the underlying NC computation or any reasoning behind the WCD calculation.

Partial Prediction Failure: The model produces valid WCD values for fewer than 80% of flows in a given TC, resulting in incomplete coverage. This affects Mistral Large 3 on CBS and Llama 3.3 on CBS, suggesting these models lose track of flow indexing in large topologies.

Timeout and Context Failure. The model cannot process the full open-ended prompt due to context window limitations or API timeout. This affects Qwen3 8B (API timeout across all TCs) and Llama 3.2 1B (context limit exceeded), confirming that small models are structurally unsuited for TSN open-end evaluation.

Empty Response: The model returns an empty response for all open-ended test cases, regardless of network topology or flow count. This failure mode exclusively affects DeepSeek-V3.2 (Thinking), which produces no output, neither WCD values nor intermediate reasoning, across all evaluated topologies, including one-switch, medium-mesh, and ring configurations, and across all flows, for both CBS and CQF mechanisms.