TSNBench: Benchmarking LLM Proficiency in Time-Sensitive Networking

Rubi Debnath[1], Daniel Bujosa[2], Luxi Zhao[3], Silviu S. Craciunas[2,4], Paul Pop[2], Sebastian Steinhorst[1]

[1]Technical University of Munich, [2]Technical University of Denmark, [3]Beihang University, [4]NXP Semiconductors

NEURAL INFORMATION PROCESSING SYSTEMS

LLMs score up to 95% on TSN multiple-choice questions, yet even the best model misses CBS worst-case delays by 150 µs on average.

Why it matters

- Time-Sensitive Networking (TSN, IEEE 802.1) gives Ethernet bounded, deterministic latency for safety-critical applications like automotive, aircraft and factories.
- Engineers are starting to use LLMs to configure such networks.
- A worst-case delay (WCD) error of tens of microseconds can break a safety-critical requirement.

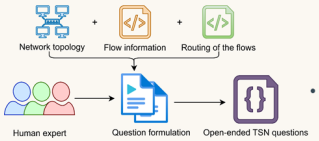
We ask:

- Do LLMs know TSN? (MCQs)
- Can they compute worst-case delays? (CBS, CQF)
- Does giving them the formula help?
- Are results robust to prompts and a real network?

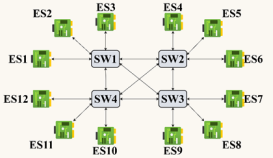
TSNBench – Dataset Generation

939 expert-validated MCQs | **200** WCD tasks: CBS + CQF | **20** WCD tasks: Real-world network (Orion)

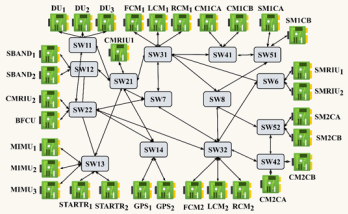




- **Open-ended WCD:** given topology, flows, and routes, compute each flow's WCD (in µs).
- **Verified ground truth:** Network Calculus (NC) solver (CBS) and closed-form upper bound (CQF).



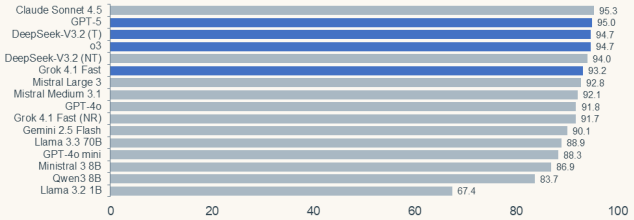
Medium-mesh topology (also one-switch and ring topologies).



Real-world network: NASA Orion Crew Exploration Vehicle (CEV).

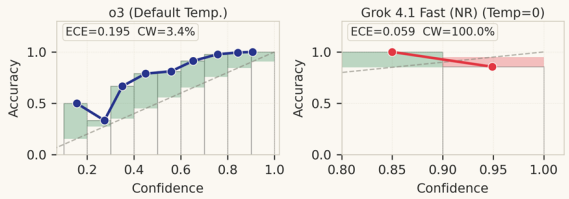
Multiple choice: Knowledge looks strong

Accuracy (%) on 939 questions; blue = reasoning models



- **67 to 95%** accuracy. Reasoning models do not beat the best general model (Claude Sonnet 4.5, 95.3%)

But models are often confidently wrong

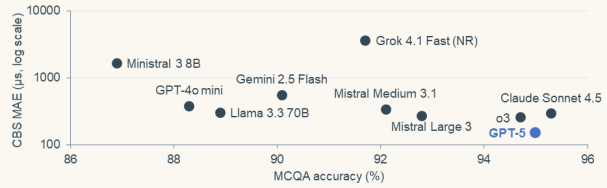


Accuracy vs. confidence: o3 (left), Grok 4.1 Fast NR (right)

11 of 16 models are confidently wrong on 95% or more of their errors.
o3 is the exception: 3.4%

Open-ended WCD: computation fails

Best model GPT-5: CBS MAE 150.2 µs, and only 39.4% of its flows within 50 µs (all others ≤ 22%). Every other model exceeds 250 µs.



High MCQA accuracy does not mean correct WCD computation.

Does giving the formula help?

WCD MAE (µs), closed book (CB) vs. formula-in-context (FC)

	CBS (network calculus)			CQF (closed form)		
Model	CB	FC	Δ %	CB	FC	Δ %
GPT-5	150.2	161.4	−7.5	107.0	0.0	+100.0
o3	255.8	501.9	−96.2	101.6	0.0	+100.0
Mistral Large 3	270.6	273.5	−1.1	54.2	19.3	+64.3
Claude Sonnet 4.5	293.0	307.7	−5.0	213.2	30.3	+85.8
Llama 3.3 70B	301.2	333.5	−10.7	173.3	5.1	+97.1
Mistral Medium 3.1	339.6	374.2	−10.2	150.3	33.0	+78.0
GPT-4o mini	378.5	461.6	−21.9	180.5	1377.1	−662.9
Gemini 2.5 Flash	554.6	258.0	+53.5	112.0	0.1	+99.9
Ministral 3 8B	1643.3	1551.7	+5.6	3057.4	1354.6	+55.7
GPT-4o	–	–	–	82.2	11.3	+86.3

Mean absolute WCD error (µs) averaged over valid test cases (of 100). Δ > 0 (blue): the formula lowers the error.

- **CQF** (one closed-form bound): the formula lowers the error for **9 of 10** models. o3 and GPT-5 hit the ground truth exactly.
- **CBS** (multi-step NC derivation): helps only **2 of 9** models. o3 gets 96% worse.

Robust to prompts and real networks?

Real network (NASA Orion):

- **CBS**: formula helps only **4 of 10** models. Best performing model is o3 with MAE 256.1 µs (closed-book setting).
- **CQF**: formula helps **7 of 11** models.

Prompt format (synthetic CBS):

- Plain-text output makes GPT-4o mini's CBS error **316× higher**.
- Most stable: o3 (CV 1.2%) and Claude Sonnet 4.5 (CV 2.0%).

Takeaways

- High MCQ accuracy can overestimate LLM proficiency in safety-critical networking.
- Difficulty follows the math: NC-based CBS fails, closed-form CQF is handled better.
- **Next step**: evaluate LLMs that produce checkable artifacts verified by analysis tools.

References

L. Zhao et al. (2018). "Timing Analysis of AVB Traffic in TSN Networks Using Network Calculus." In: IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS). IEEE, pp. 25-36.

X. Guo et al. (2025). "Toward Engineering AGI: Benchmarking the Engineering Design Capabilities of LLMs." In: NeurIPS Datasets and Benchmarks Track 2025.

T. Hua et al. (2025). "ResearchCodeBench: Benchmarking LLMs on Implementing Novel Machine Learning Research Code." In: NeurIPS Datasets and Benchmarks Track 2025.

Code (GitHub)

Dataset (Kaggle)