Dynamic Delayed Tree Expansion
For Improved Multi-Path Speculative Decoding

Rahul Thomas[1,2,†,*] Teo Kitanovski[1,3,†] Micah Goldblum[1,2] Arka Pal[1]

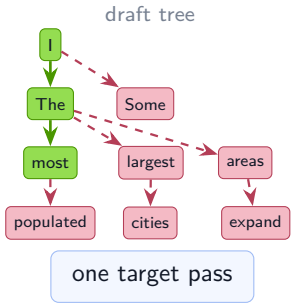
[1]Ritual [2]Columbia University [3]Vanderbilt University

† Equal contribution * Corresponding author

NeurIPS 2026

What is multi-path speculative decoding?

- Autoregressive decoding is typically **memory-bound**, not compute-bound.
- A cheap draft model proposes a tree.
- The target model scores the entire tree.
- Verification accepts a path while preserving the target distribution exactly **(lossless)**.
- Throughput depends on **accepted tokens** and the **draft and target latency**.



Where should we spend branching capacity in the draft tree?

Traversal is the best existing verifier

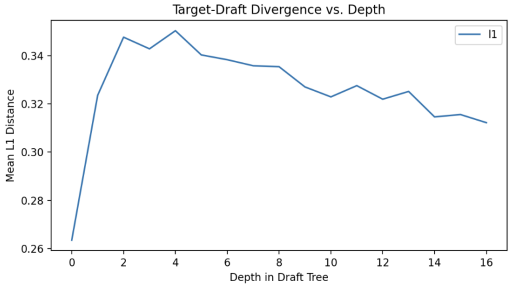
Method	Qwen	Gemma	Llama	Average
NaiveTree	19.16	6.00	14.71	13.29
NSS	16.88	5.94	12.91	11.91
SpecTr	19.01	6.09	14.86	13.32
SpecInfer	19.58	6.21	14.86	13.55
Khisti	18.67	5.93	13.66	12.75
Traversal	**21.35**	**11.56**	**14.77**	**15.89**

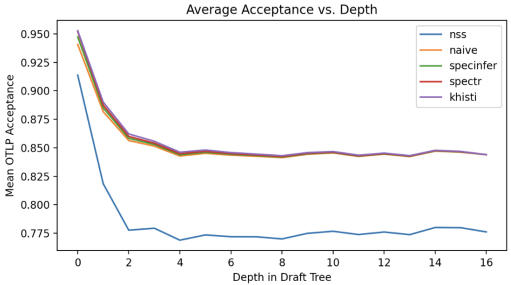
- Throughput in tokens/sec, averaged across **5 datasets** and **8 sampling settings**.
- Traversal is best overall. No OT-based verifier dominates across model families.

Why does bottom-up Traversal outperform top-down OT verification?

Root branching wastes diversity

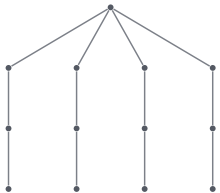
- On 200,000+ draft tree roots, target–draft divergence rises sharply after the root.
- At the same time, node-wise OT acceptance falls rapidly with depth.
- Branching near the root spends diversity where draft and target already agree.



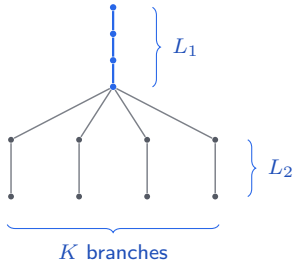


Delayed tree expansion

Standard i.i.d. expansion

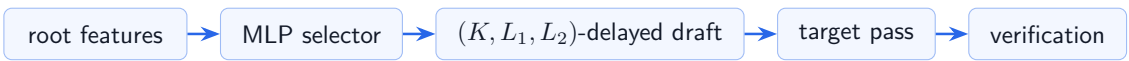


Delayed expansion



- Draft a length-L_1 trunk, then expand into K i.i.d. length-L_2 continuations.
- Delayed trees are **lossless** for both OT-based and Traversal verification.
- Branch later, where draft diversity is more valuable for block efficiency gains.

Neural delayed expansion (NDE)



- Static (K, L_1, L_2) is not optimal across contexts, sampling settings, and hardware.
- Train the selector offline with verifier block efficiencies and draft/target latencies.
- Choose action before drafting, with a training objective that optimizes throughput:

$$TPS_\pi(c) = \frac{\mathbb{E}_{a\sim\pi(\cdot|c)}[\tau + 1 \mid c, a]}{\mathbb{E}_{a\sim\pi(\cdot|c)}[W(c, a)]}$$

NDE improves OT methods and Traversal

Method	Qwen	Gemma	Llama	Average
Traversal	21.35	11.56	**14.77**	15.89
SpecInfer-NDE	**22.54**	**13.26**	14.27	**16.69**

- SpecInfer-NDE is **5%** faster than Traversal, closing the OT-Traversal gap.
- Traversal-NDE improves average Traversal throughput by **26%** across families.

Where and when we expand the tree matters as much as how we verify it.