LATERAL TREE-OF-THOUGHTS SURPASSES TOT BY INCORPORATING LOGICALLY-CONSISTENT, LOW-UTILITY CANDIDATES

Anonymous authors
Paper under double-blind review

ABSTRACT

1 INTRODUCTION

2 MOTIVATION

3 RELATED WORK

4 ARCHITECTURE DESIGN

Goal. LToT is a search-time controller for reasoning with language models (LMs) that (i) keeps *mainlines* narrow to avoid exponential blow-up in depth and (ii) makes *lateral* exploration very wide but cheap via a budgeted racing procedure with short-circuit promotion. The controller decides when to exploit mainlines vs. explore laterals, and—during exploration—how to allocate compute across many lateral branches while maintaining guarantees on cost and false promotions.

4.1 PROBLEM SETTING AND NOTATION

We reason over a rooted tree (or DAG) of partial traces. Each node x is a partial solution; its children are produced by prompting the LM with x. Two evaluators score nodes:

$$v(x) \in \mathbb{R} \quad \text{(utility; e.g., answer- or verifier-aligned)}, \qquad c(x) \in [0,1] \quad \text{(logical consistency / soundness)}.$$

We measure compute in either node expansions or tokens and denote cumulative compute by C.

Frontier, origins, and exploitation set. At time t the search maintains a frontier $\mathcal{F}_t$ and an *exploitation set* $M_t \subseteq \mathcal{F}_t$ of nodes eligible for *mainline* exploitation. Nodes carry an immutable `origin` tag in $\{\text{MAINLINE\_ORIGIN}, \text{LATERAL\_ORIGIN}\}$ indicating how they first entered the frontier. We also maintain a *mainline acceptance bar* B_t (e.g., the best-so-far v or a top-k mean with a small margin $\delta > 0$).

Mainlines vs. laterals. Children with high v are admitted to M_t (mainlines). Children with low v but high c enter the *lateral pool* L_t for potential exploration. Intuitively, laterals represent hypotheses that appear unpromising under a myopic utility but are logically coherent and may become valuable after a short lookahead.

Branch envelope and gain. For a lateral branch i (rooted at node x_i), let $V_i(h)$ denote a branch *envelope*—e.g., a Top-k mean of v among leaves within horizon h steps from x_i (or within a per-branch micro-beam). We write $C(h)$ for the compute required to reach horizon h and define the compute-normalized improvement between horizons $h' < h$ as

$$g_i(h,h') \;=\; \frac{V_i(h) - V_i(h')}{C(h) - C(h')}.$$

These quantities let us reason about *marginal value of compute*, not just absolute levels.

Algorithm 1 LToT controller (high level)

```
Inputs: initial frontier F_0, evaluator v, consistency c, plateau thresholds; LR-SC params (η, b_0, ρ, κ, δ).
Initialize M_0 with high-v children; L_0 with low-v, high-c children; set bar B_0.
while budget remains do
    Exploit M_t while EWMA of ΔB_t per compute ≥ τ (with a small patience & hysteresis).
    Explore laterals with LR-SC over the current lateral pool (Alg. 2).
    if some lateral branch reaches v ≥ B_t + δ (promotion) then
        add promoted node(s) to M_t; update B_t; return to exploitation
    else
        freeze survivors for future phases; return to exploitation
    end if
end while
```

4.2 Controller overview

Exploit–explore loop. LToT alternates between:

1. **Mainline exploitation.** Expand nodes from M_t while a compute-normalized progress statistic (e.g., an EWMA of ΔB_t per unit compute) exceeds a plateau threshold. This keeps mainlines narrow (beam- or quota-capped).
2. **Lateral exploration via LR-SC.** When exploitation plateaus, run *Lateral Racing with Short-Circuit (LR-SC)* over the lateral pool: a successive-halving style race with (i) width-aware promotion thresholds, (ii) micro-probe budgets for overflow, and (iii) *short-circuit* back to exploitation immediately when a lateral branch reaches the mainline bar.

Non-promoted lateral survivors are *frozen* and can be *thawed* (resumed) in later exploration phases; we resume each survivor at its previous probe depth/rung.

4.3 LR-SC: overflow-capped racing with short-circuit

Let N be the active lateral width. LR-SC proceeds in rungs $r = 0, 1, \dots$ with *culling factor* $\eta > 1$. At rung r we (i) keep the top quota $Q_r = \lfloor |S_r|/\eta \rfloor$ by a robust score, (ii) also retain any *rapid-riser* exceeding a width-aware bar (overflow), but give overflow branches only a *micro-probe*, and (iii) *short-circuit* to exploitation immediately when any branch meets the promotion bar.

Scores and width-aware bar. For branch i at rung r we compute a compute-normalized improvement g_i (using V_i) and a robust standardization z_i (e.g., subtract rung median and divide by a MAD-like scale). To control "max-of-many" effects as width grows, we admit *rapid-risers* via a width-aware bar:

$$z_i \;\geq\; \underbrace{\kappa\sqrt{2\log|S_r|}}_{\text{width penalty}} + \delta,$$

with $\kappa \approx 1$ and a margin $\delta > 0$. We optionally standardize scores within parent-depth bands to compare fairly across heterogeneous depths.

Overflow cap. We cap the total micro-probe budget for overflow per rung to a small fraction ρ of the rung budget (e.g., $\rho \in [0.1, 0.2]$), ensuring per-rung cost stays near constant.

Derivative-based continuation (principle and instantiation). We view V_i as a function of horizon/compute and continue branch i if a discrete derivative of order $m \in \{1, \dots, M\}$ is reliably positive:

$$\widehat{\Delta^{(m)} V_i} \;\geq\; \mathrm{bar}(|S_r|, M) \quad \text{with} \quad \mathrm{bar}(|S_r|, M) \propto \sqrt{2\log(|S_r| \cdot M)}.$$

In practice we cap $M = 2$ for stability and use: *(i)* first derivative (slope) $s_i = g_i(h_r, h_{r-1})$ and *(ii)* second derivative (curvature) $\kappa_i = s_i(r) - s_i(r-1)$, estimated over the last few rungs and normalized by compute; a third-order check may be included in an appendix. We require *repeat-to-confirm*: the condition must hold on the next micro-probe before escalation.

Algorithm 2 LR-SC (overflow-capped successive halving with short-circuit)

1: **Inputs:** active lateral set S_r (size N), culling factor $\eta > 1$, base budget b_0, overflow cap ρ, thresholds (κ, δ), horizon schedule $(h_0, h_1, \dots)$
2: Compute robust improvement scores $\{z_i\}_{i \in S_r}$ from compute-normalized gains (optionally depth-standardized).
3: $Q_r \leftarrow \lfloor |S_r|/\eta \rfloor$; $T \leftarrow$ top Q_r by z_i; $R \leftarrow \{\, i : z_i \ge \kappa\sqrt{2\log|S_r|} + \delta \,\}$.
4: Assign budget $b_{\text{full}} = b_0 \eta^r$ to $i \in T$; assign micro-probe b_{micro} to up to $\lfloor \rho |S_r| \rfloor$ branches in $R \setminus T$ (by z_i); freeze the rest.
5: Expand per budgets to horizon h_r (within-branch beam is tiny); update V_i, g_i, and bar B_t.
6: **if** some i satisfies $V_i \ge B_t + \delta$ and *repeat-to-confirm* **then**
7: promote i; **short-circuit** to exploitation
8: **end if**
9: $S_{r+1} \leftarrow T \cup$ (confirmed overflow); $r \leftarrow r + 1$; continue if budget remains.

4.4 Promotion and safety

A lateral promotes when its envelope meets the mainline bar: $V_i \ge B_t + \delta$. When v is verifier-aligned (e.g., unit tests for code, exact-match for math), this binds promotion to correctness. For plausibility-aligned v, LToT can add a lightweight dual gate at promotion time: $V_i \ge B_t + \delta$ *and* an aggregate path-consistency (e.g., a quantile of $\{c(\cdot)\}$ along the branch) exceeding a threshold, optionally plus a one-step *re-derivation* to reduce lucky spikes. These checks cost one micro-probe and do not change the asymptotics.

4.5 Theoretical properties

We summarize the main guarantees; proofs are short and rely on standard successive-halving arguments and sub-Gaussian tail bounds for rung-wise statistics.

Cost law (pseudolinear in lateral width). Let N_0 be the initial lateral width. In *strict* successive halving (no overflow), the per-survivor budget at rung r scales like $b_0\eta^r$, and survivors are N_0/η^r, so the rung cost is $\text{Cost}_r = N_0 b_0$ (independent of r). With $R = \lceil \log_\eta N_0 \rceil$ rungs, the total lateral cost is

$$\boxed{\text{Total} \;=\; \Theta\big(N_0\, b_0\, \log_\eta N_0\big).}$$

In LR-SC with overflow cap $\rho \in (0, 1)$ and micro-probe $b_{\text{micro}} \ll b_0$, the rung cost is at most $(1+\rho)N_0 b_0$, hence the same asymptotic order with a constant factor $(1+\rho)$. Short-circuit promotion can only reduce cost. Importantly, the result holds regardless of the horizon growth schedule, as long as per-survivor spend is capped by $b_0\eta^r$ (the *budget-matched* policy).

Rung count (short in depth). The number of rungs required to reduce N_0 laterals to $O(1)$ survivors is $R = \lceil \log_\eta N_0 \rceil$, i.e., logarithmic in lateral width. Thus LR-SC is *wide and short*: constant per-rung cost and $\Theta(\log N_0)$ rungs.

Mainline growth (why we keep mainlines narrow). If at each mainline layer we admit a fixed fraction a of k children (effective reproduction $r_{\text{main}} = ak > 1$), then expansions to depth D are $\Theta(r_{\text{main}}^D)$ (exponential). With a beam/width cap W, mainline cost becomes $\Theta(D\,W\,k)$ (linear in depth). LToT therefore keeps W small and invests surplus compute in laterals, where width is cheap.

Width-aware threshold controls family-wise errors. Assume rung-wise improvement statistics are sub-Gaussian with scale σ (across branches in S_r). Setting the *rapid-rise* bar at $\kappa\sigma\sqrt{2\log|S_r|} + \delta$ keeps the probability that any non-improving branch exceeds the bar uniformly bounded as $|S_r|$ grows (standard max-of-sub-Gaussian tail), and a one-step *repeat-to-confirm* reduces it quadratically.

Horizon-lifted detection of delayed payoffs. Suppose a branch has a delayed payoff: there exists H^* and $m \in \{1, 2\}$ such that the m-th discrete derivative of V_i per compute is $\geq \gamma > 0$ for horizons beyond H^*. Under a geometric horizon schedule (e.g., $h_r = 2^r$ within the budget cap) and the derivative-based continuation rule with width-aware thresholds and repeat-to-confirm, the branch is detected and survives to promotion within $O(\log H^*)$ rungs (intuitively, each rung doubles the tested horizon). Total exploration cost remains $\Theta(N_0 \log_\eta N_0)$ because the per-survivor spend never exceeds $b_0\eta^r$.

Independence from nominal horizon multiplier. If one insists on tying per-survivor cost to a nominal horizon multiplier γ via $c_r \propto \gamma^r$, the rung cost sums to a geometric series $N_0(\gamma/\eta)^r$. Thus for $\gamma \leq \eta$ the total remains $O(N_0 \log_\eta N_0)$ (or even $O(N_0)$ when $\gamma < \eta$). In practice we adopt the budget-matched policy: cap spend by $b_0\eta^r$ and allocate within-branch depth/width flexibly up to that cap.

4.6 DESIGN CHOICES AND DEFAULTS

Exploitation trigger. EWMA of compute-normalized mainline progress with small patience and hysteresis; depth-banded statistics if v drifts with depth.

LR-SC parameters. $\eta \in \{3, 4, 5\}$; $b_0 \in \{1, 2\}$ expansions; micro-probe $b_{\text{micro}} = 1$; overflow cap $\rho \in [0.1, 0.2]$; width-aware bar with $\kappa \approx 1.0$ and a small margin δ over the mainline bar. Derivative-based continuation with slope+curvature ($M=2$) using a short window and robust scales; optional third-order check in ablations.

Promotion predicate. Require $V_i \geq B_t + \delta$; if v is not verifier-aligned, add a minimal path-consistency aggregate and a one-step re-derivation check. Both add at most one micro-probe and preserve the asymptotics.

Freeze–thaw. Cache for each survivor: rung index, envelope V_i, recent improvement stats, parent depth, and a lightweight duplicate signature; resume from the same rung during the next exploration phase and evict stale or dominated branches by constant-time tests (e.g., $UCB < B_t - \delta$ for several revisits).

Summary. LToT turns surplus compute into breadth where it is cheapest (laterals) while keeping mainlines narrow. Its LR-SC core provides near-linear (pseudolinear) cost in lateral width, width-aware error control, and immediate promotion when a lateral demonstrably reaches mainline utility.

5 EXPERIMENTS

6 RESULTS AND DISCUSSION

7 FUTURE WORK

8 CONCLUSION

REFERENCES