Test-Time Compute Scaling and Language Model Reasoning Performance on Benchmark Suites

Assignee Research

June 6, 2026

Abstract

This report synthesises findings from 14 peer-reviewed papers addressing the following research question: How does test-time compute scaling improve language model performance on reasoning benchmarks v15. 13 claims were extracted from source literature; 2 were independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 5.6/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: Understanding Dynamic Compute Allocation in Recurrent Transformers. Research question: How does test-time compute scaling improve language model performance on reasoning benchmarks v15.

2 Methodology

Systematic literature search across multiple databases yielded 14 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 5.6/10.

3 Results

14 papers retrieved. 13 claims extracted; 2 independently verified. Quality review score: 5.6/10.

4 Limitations

This report is a machine-generated literature synthesis and does not constitute original research. Automated retrieval and verification may introduce

errors or omissions. Review scores reflect automated assessment, not human peer review. Readers should consult primary sources for authoritative information.

5 Extracted Claims

Claim	Verified	Confidence
ANIRA (Adaptive Neural Iterative Reasoning Architectures) enables token-level adaptivity by varying the amount of comput	×	0.10
ANIRA follows a Prelude–Recurrent–Coda architecture where the Prelude and Coda are small stacks of causal Transformer la	×	0.04
ANIRA-E is a variant where depth allocation is decided from a shallow (pre-recurrence) representation.	×	0.03
ANIRA-O is a variant where online halting decisions are made for each token between each recurrent layer.	×	0.09
Once a token completes its allocated number of recurrent steps in ANIRA, its representation is frozen and subsequent ite	×	0.03
ANIRA models were trained on MANO instances with difficulty levels L ranging from 3 to 16.	×	0.03
ANIRA models were trained on BREVO instances with sizes N ranging from 3 to 30.	×	0.04
The maximum number of recurrent iterations (D) was set to 6 for all tasks except MANO, where D was set to 14.	×	0.03
ANIRA models learn to allocate compute consistent with task complexity on MANO and BREVO tasks without explicit complexi	×	0.08
Compute allocation aligned with task complexity emerges without explicit difficulty supervision but does not imply algor	✓	0.32
Early compute decisions in ANIRA rely on static structural cues.	✓	0.19
Online halting decisions in ANIRA more closely track algorithmic execution state compared to early decisions.	×	0.14
The training dynamics of adaptive computation in ANIRA follow a consistent two-phase regime: learning followed by comput	×	0.06

References

- http://arxiv.org/abs/2504.00869v2
- http://arxiv.org/abs/2510.00071v2
- http://arxiv.org/abs/2602.08864v1