NEURIPS 2026

ParaVT

Taming the Tool Prior Paradox for Parallel Tool Use in Agentic Video Reinforcement Learning

> Learn to gather parallel video evidence
> with parseable and reward-relevant tool use.

Zuhao Yang[1,2,7] Kaichen Zhang[1,3,7] Sudong Wang[1,4] Keming Wu[1,5,7] Zhongyu Yang[6]
Bo Li[7] Xiaojuan Qi[3] Shijian Lu[2,*] Xingxuan Li[1,*] Lidong Bing[1]

[1] Apodex [2] NTU [3] HKU [4] HKUST(GZ) [5] THU [6] LZU [7] LMMs-Lab

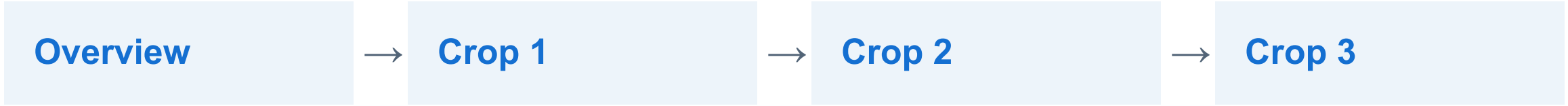
* Corresponding authors

https://evolvinglmms-lab.github.io/ParaVT/

MOTIVATION

Why gather temporal evidence in parallel?

A long video can hide the decisive evidence in a short window.



Sequential crop turns

01

Localization errors

One wrong crop can steer later reasoning off course.

02

Growing context

Each crop adds evidence to the running context.

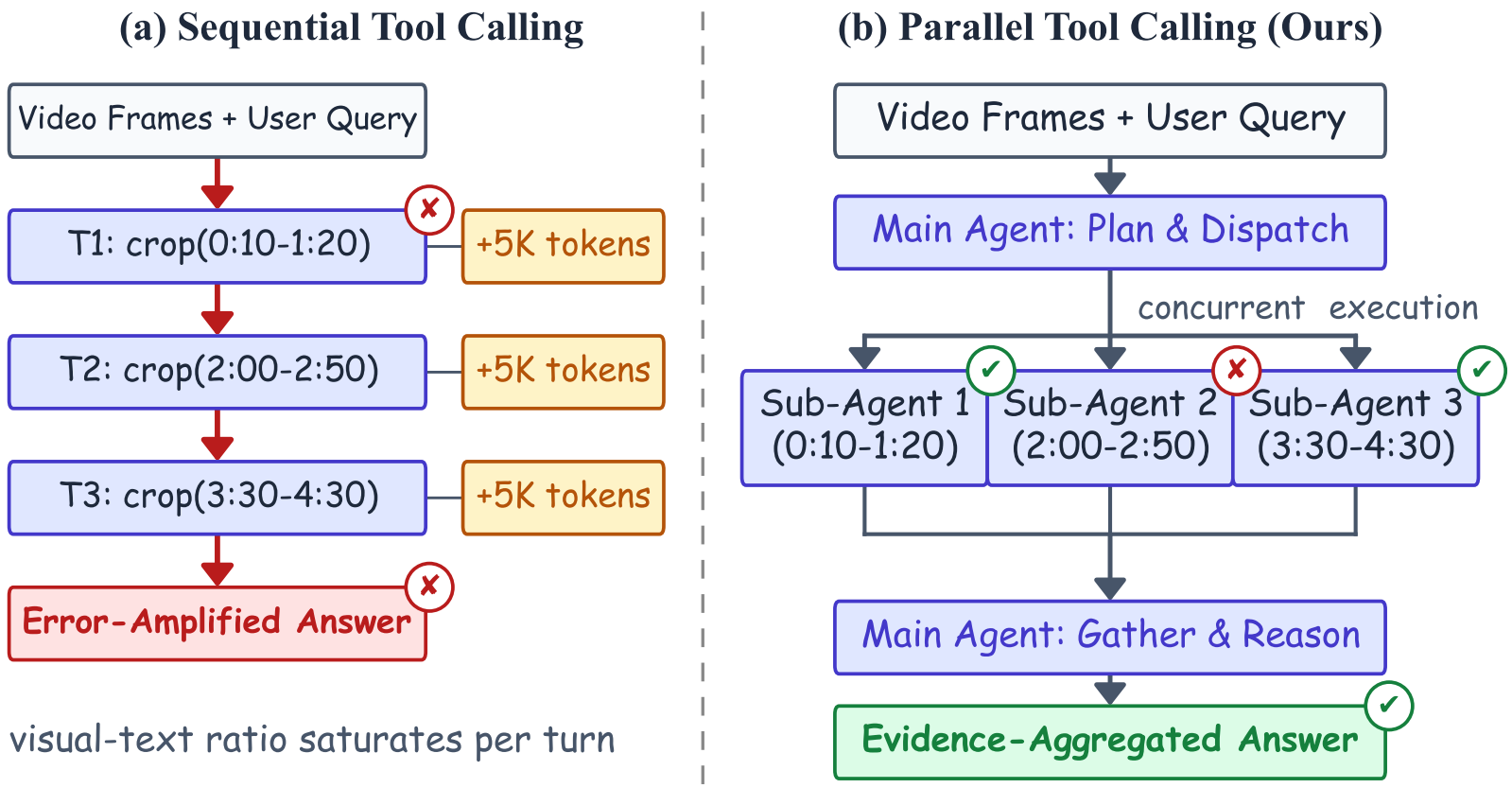
03

Repeated encoding

The main model revisits that context every turn.

Parallel dispatch lets the main agent compare independent windows before answering.

One shared policy: parallel evidence, then aggregation



Shared weights

Main agent: dispatch and aggregate.
Sub-agents: summarize assigned windows.

97K SFT samples

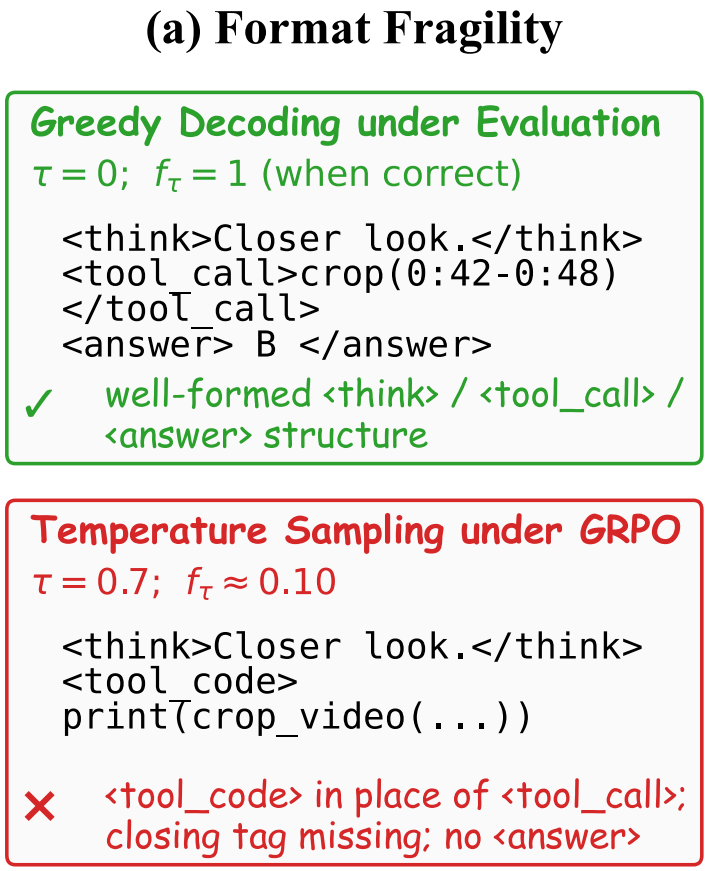
Learn the parallel tool-call schema.

4,406 RL samples

Optimize main-agent rollouts with PARA-GRPO.

Sub-agent generations are excluded from the RL loss; they change through shared weights.

Failure 1: greedy format does not survive sampled RL



Competing tool schemas reappear

`<tool_call>` → `<tool_code>`

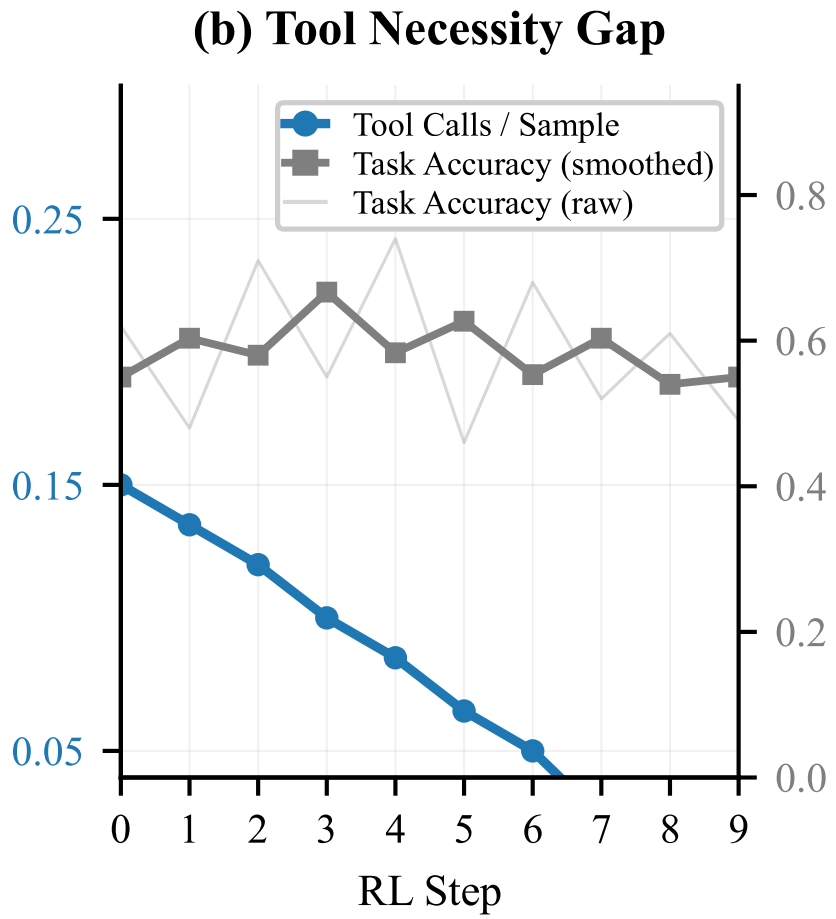
Closing tags disappear

Unclosed reasoning and missing answer blocks make trajectories difficult to parse.

Malformed trajectories disrupt tool execution before useful tool-use credit can be assigned.

Paper figure: greedy decoding vs. vanilla-GRPO sampling (temperature 0.7).

Failure 2: correct answers can reward skipping tools



The overview can already be enough

Tool calls approach zero within 7 steps,
while task accuracy continues to vary.

Vanilla GRPO: cross-model observations

Qwen3-VL-8B · stronger tool prior

Explores tools; format collapses.

Qwen2.5-VL-7B · weaker tool prior

Retains format; emits no tool calls.

The model comparison does not isolate prior strength from other model differences.

METHOD OVERVIEW

PARA-GRPO addresses structure and incentive together

Exploration Anchoring

Can the rollout be parsed?

Fix the opening reasoning tag.
Reward structural closure.
Penalize an unclosed reasoning block.

nFrames Gating

Can using tools improve reward?

Vary the overview-frame budget.
Make missing evidence recoverable
through temporal crop tools.

Completed rollout → sequence-level reward → group-normalized advantage

The advantage updates all trained main-agent tokens.

Parseability-Anchored and Ratio-Gated GRPO

COMPONENT 1

Exploration Anchoring preserves structural boundaries

1 Fix the response entry

```
<think>\n
```

The model remains free to choose its reasoning content and tool-call sequence.

2 Credit a final answer block

Reward recovery into <answer> … </answer> even when intermediate structure is imperfect.

3 Check completed-rollout structure

+ Reasoning block is closed

+ Full reasoning-to-answer flow

− Opened <think> never closes

String checks produce one scalar.
GRPO broadcasts its group advantage.

Reward closure at the sequence level; the signal does not provide per-token credit.

COMPONENT 2

nFrames Gating creates room for tools to help

For each prompt, sample one overview budget:

4 frames	8 frames	16 frames	32 frames	64 frames

n ~ Uniform {4, 8, 16, 32, 64}

All G = 8 rollouts for that prompt share the same n.

Reduced overview (n < 64)

Tools can recover evidence that is missing from the overview and earn higher task rewards.

Full overview (n = 64)

Retain easy prompts where direct answering is sufficient and tool skipping is appropriate.

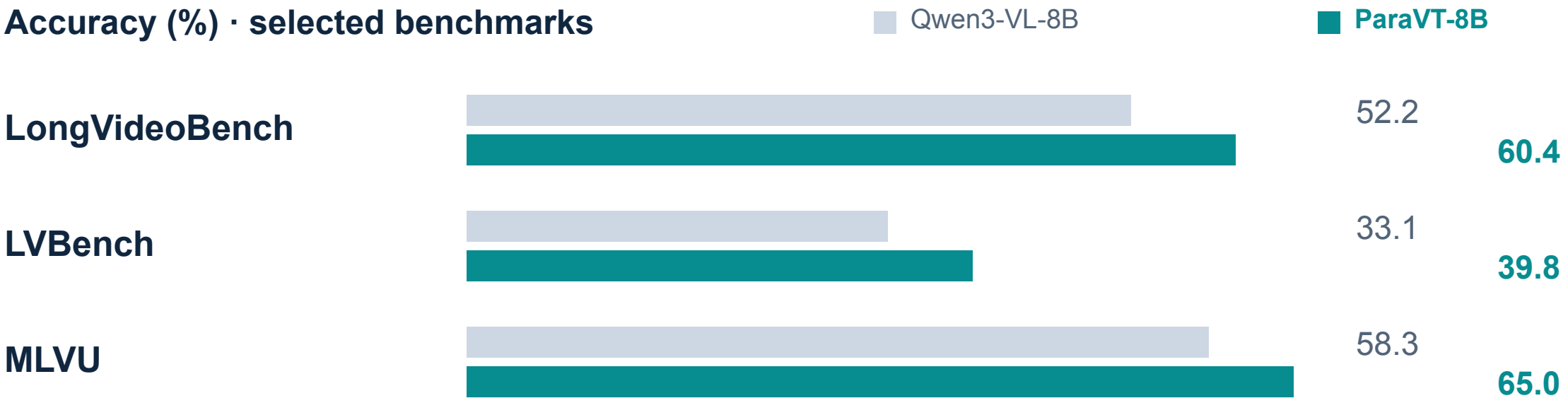
MAIN RESULTS

Higher accuracy across long-video evaluations

+7.9%

average relative improvement over Qwen3-VL-8B
across 7 evaluation splits / 6 benchmarks

Accuracy (%) · selected benchmarks



Unified 64-frame overview protocol. Bars share a zero baseline; labels retain manuscript values.

EFFICIENCY

Parallel dispatch lowers inference cost

Same trained checkpoint: sequential → parallel dispatch

Mean latency	**80.1 → 48.5**	seconds
Prompt tokens	**136,502 → 77,380**	per query
Main-model calls	**4.23 → 2.40**	per query

94.1% of 11,460 evaluated samples issue multiple crops within one turn.

Per-query cost measured at 64-way client concurrency; latency values are means.

TRAINING DIAGNOSTICS

The two components have complementary effects

Variant	Format reward	Tool calls / rollout
Vanilla GRPO	0.13	0.02
+ Exploration Anchoring	0.35	0.19
+ nFrames Gating	0.10	1.36
Full PARA-GRPO	**0.41**	**0.21**

More calls alone do not mean better tool use.

Full recipe: higher accuracy than both single-component variants on all six ablation splits.

Training-stream means at temperature 0.7; call counts are not inference frequencies.

TAKEAWAY AND RESOURCES

Parallel evidence needs structure and incentive

Dispatch in parallel.

Cross-check independent temporal evidence.

Anchor parseability. Gate the overview.

Keep exploration executable and reward-relevant.

Project, paper, and resources

Code	github.com/EvolvingLMMs-Lab/ParaVT
Data	huggingface.co/datasets/ParaVT/ParaVT-Parquet
Weights	huggingface.co/ParaVT/ParaVT-8B

Scope: one main backbone and a temporal crop tool; broader tool and backbone transfer remains open.