CASE: COST-AWARE STOPPING FOR EFFICIENT LONG-VIDEO AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-video agents can actively gather question-relevant evidence, but they typically leave a central decision implicit: when has the agent seen enough to answer? We propose CASE, a plug-in termination framework that frames this decision as policy-conditioned sequential stopping. At each causal checkpoint, CASE combines an auxiliary multiple-choice assessment of accumulated evidence with the host agent's execution state. From complete native trajectories, we construct a cost-aware target that compares answering now with stopping later along the same search path, accounting jointly for answer correctness and the full cost of continued reasoning. A lightweight Ridge regressor learns this decision gap and produces STOP/CONTINUE decisions. We evaluate three vision-language models with VideoSeek and AVP. On Video-MME, end-to-end accuracy changes by $+0.67$ percentage points on average while CASE reduces model-token use by 53.63%. The same frozen policies then transfer zero-shot to LongVideoBench and MLVU, with end-to-end accuracy changes of $+3.38$ and $+4.58$ points while saving 58.78% and 51.28% of model tokens, respectively. Across all agent–model–benchmark combinations, CASE attains the highest accuracy–efficiency Pareto-frontier coverage among the compared stopping methods (83.3%) at the selected operating points. Online execution preserves this favorable accuracy–efficiency trade-off and additionally reduces measured runtime by 54.1% on average. CASE provides a plug-in termination framework for long-video reasoning agents, enabling them to decide when further evidence acquisition is no longer worthwhile.

1 INTRODUCTION

Long-video question answering often depends on locating sparse question-relevant evidence across a long temporal span. As videos grow longer, dense analysis increases the costs of visual processing, context construction, and model inference.

Existing work has substantially improved how long-video agents find evidence. VideoAgent methods use iterative visual queries and structured memory to locate and organize question-relevant information, while VideoSeek combines video logic flows with multi-granularity tools to narrow search from a global overview to relevant temporal regions in a think–act–observe loop (Wang et al., 2024; Fan et al., 2024; Lin et al., 2026). These advances make active evidence acquisition effective, yet leave a key decision unresolved: after receiving new evidence, has the agent seen enough to answer, or is another search step still worthwhile?

Existing agents typically leave termination implicit in the generative model or impose a fixed step or budget limit. Some systems add explicit sufficiency checks; AVP, for example, uses a reflector to assess whether accumulated evidence is sufficient to answer Wang et al. (2026). Fixed limits apply the same stopping pressure across questions, while confidence, entropy, or agreement signals summarize current certainty rather than the value of another search step (Zhou et al., 2020; Schuster et al., 2022; Kieback et al., 2026). Useful evidence can still follow a high-confidence state, and an unstable answer can coincide with an expensive low-value tool call. The relevant comparison is therefore between the risk of answering now and the expected benefit and cost of continuing.

Sequential hypothesis testing provides a natural framework for this decision. Unlike fixed-sample inference, it allows a decision maker to stop as evidence arrives: each observation updates support for competing hypotheses, after which the process either terminates or continues sampling, trading lower

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

decision risk against additional observation cost (Wald, 1945). Active multiple-hypothesis testing and controlled sensing extend this setting to chosen observations and nonuniform costs (Chernoff, 1959; Naghshvar & Javidi, 2013; Nitinawarat & Veeravalli, 2015; Vershinin et al., 2025). Once the host agent's evidence-acquisition policy is fixed, the remaining problem is optimal stopping—comparing the risk of answering now with the value of acquiring more information along that policy (Shiryaev, 2008; Novikov, 2010). This perspective maps directly to long-video agents: answer options are competing hypotheses, video-tool results arrive as sequential evidence, and each causal checkpoint poses a STOP/CONTINUE decision. Because future tool use and model computation depend on the evolving trajectory, continuation cost is state-dependent and heterogeneous.

Motivated by this perspective, we propose CASE (**C**ost-**A**ware **S**topping for **E**fficient Long-Video Agents), a plug-in termination framework for long-video agents. The host agent retains control over where to seek evidence, while its learned stopping controller decides whether to execute the next proposed visual action at each causal checkpoint. At each checkpoint, it combines answer support with the agent's execution state. Complete native trajectories expose the future evidence and realized costs reachable under the host search policy, yielding the stopping target

$$g_t = L_{\text{stop}}(t) - L^*_{\text{continue}}(t),$$

where $L^*_{\text{continue}}(t)$ is the lowest answer-risk-plus-cost loss among later checkpoints and the native terminal point on the same trajectory. We factorize this gap into a future answer-quality advantage and a heterogeneous-cost correction and fit both with lightweight Ridge models. Figure 2 summarizes the execution and learning pipeline. This checkpoint-level stopping decision differs from Stop-RAG's iterative retrieval stopping and CAM-DF's pre-execution tool configuration (Park et al., 2025; Feng et al., 2026).

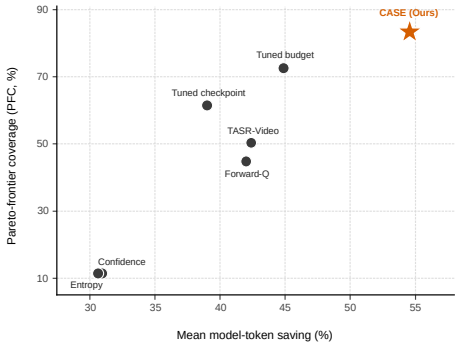

Figure 1: **Cross-configuration accuracy–efficiency frontier.** PFC is the share of agent–model–benchmark configurations on the Pareto frontier.

We evaluate CASE on VideoSeek and Active Video Perception (AVP) (Lin et al., 2026; Wang et al., 2026) with three open-source vision-language models over Video-MME, LongVideoBench, and MLVU. Figure 1 summarizes the selected accuracy–efficiency operating points across the 18 agent–model–benchmark configurations. Policies selected on Video-MME transfer zero-shot to the other benchmarks and are also evaluated online, where the favorable accuracy–efficiency trade-off persists and measured runtime falls by 54.1% on average.

To summarize, our contributions are three-fold:

- We formulate tool-guided long-video question answering as a multi-hypothesis sequential stopping problem under an existing evidence-acquisition policy and model the state-dependent heterogeneous costs induced by generative video search.
- We propose CASE, a plug-in termination framework for long-video reasoning agents. Its learned stopping controller factorizes trajectory-level stopping supervision into future answer-quality advantage and a heterogeneous continuation-cost correction, yielding a lightweight causal policy while keeping the host vision-language model fixed.
- Across the evaluated hosts, termination exposes substantial removable model-token use: without changing the host search policy or VLM, CASE reduces model-token use by 54.6% on average, transfers zero-shot across benchmarks, and achieves the highest Pareto-frontier coverage (83.3%) among the compared stopping methods. Online execution preserves this trade-off while reducing measured runtime by 54.1%.

2 Related Work

Long-video evidence acquisition and efficient processing. Long-video agents localize question-relevant evidence through visual queries, temporal search, and memory (Wang et al., 2024; Fan et al.,

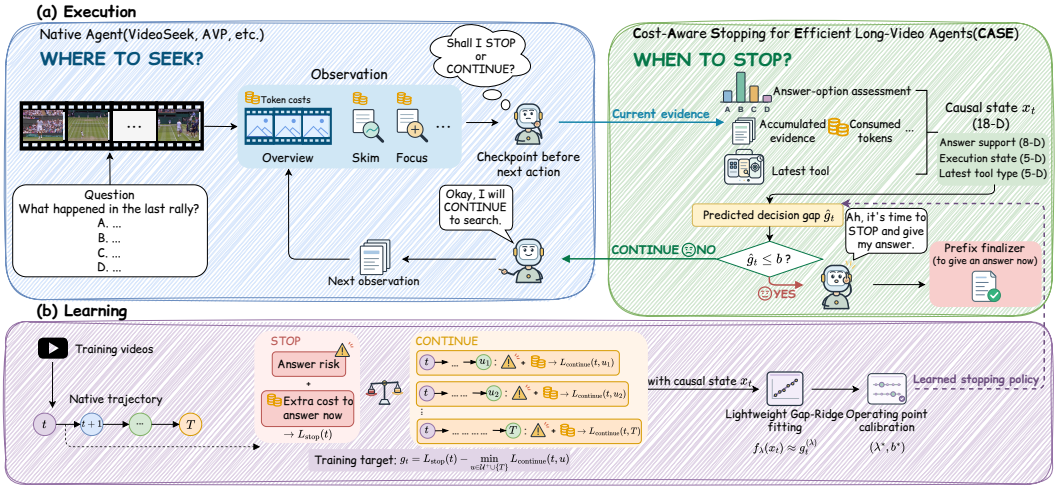

Figure 2: **Overview of CASE.** The host chooses where to acquire evidence; CASE wraps the native search loop with STOP/CONTINUE control and constrained prefix finalization. Complete native trajectories supervise answer quality and heterogeneous continuation cost.

2024; Zhi et al., 2025), with newer systems adding specialized search and reasoning tools (Zhang et al., 2025; Pan et al., 2026; Lin et al., 2026; Li et al., 2026; Liu et al., 2026; Wang et al., 2026). Other work improves reliability or efficiency through discriminative acquisition, adaptive frame/token allocation, and representation compression (Qiu et al., 2026a; Xu et al., 2026; Wang et al., 2025b; Tang et al., 2025; Zhu et al., 2026; Qi et al., 2026; Jiang et al., 2025; Wang et al., 2025a). These methods primarily decide where or how much to inspect or compress; CASE leaves acquisition unchanged and asks whether the next proposed search action is worth its state-dependent continuation cost.

Sequential decisions with observation costs. Sequential hypothesis testing and controlled sensing trade decision error against observation cost, including active and nonuniform observations (Wald, 1945; Chernoff, 1959; Naghshvar & Javidi, 2013; Nitinawarat & Veeravalli, 2015; Vershinin et al., 2025). Optimal stopping characterizes when further information acquisition is no longer worthwhile under a given observation process (Shiryaev, 2008; Snell, 1952; Novikov, 2010). CASE instantiates this perspective for generative video search with state-dependent continuation cost.

Adaptive stopping and decision-focused learning. Adaptive Computation Time and PonderNet learn input-dependent halting (Graves, 2016; Banino et al., 2021); BERT Loses Patience, CALM, Fast yet Safe, and Learning To Stop While Learning To Predict use agreement, confidence, calibrated risk, or jointly learned stopping (Zhou et al., 2020; Schuster et al., 2022; Jazbec et al., 2024; Chen et al., 2020). For iterative acquisition, Probing-RAG predicts retrieval need, TASR uses answer agreement and a calibrated margin, and Stop-RAG learns continuation value (Baek et al., 2025; Kieback et al., 2026; Park et al., 2025). CAM-DF selects tool configurations before execution, while LongVideo-R1 jointly trains navigation and halting (Feng et al., 2026; Qiu et al., 2026b). In contrast, CASE controls termination of an existing host at a causal checkpoint, using current evidence/execution state and realized downstream quality and cost without modifying the host VLM or search policy.

3 METHOD

3.1 SEQUENTIAL STOPPING ALONG THE NATIVE SEARCH

CASE augments an existing tool-guided video agent without changing its evidence-acquisition policy. For question i, let $Y_i \in \{1, \dots, K_i\}$ be the correct option and $\mathcal{F}_{i,t}$ the accumulated evidence, history, and computation available at checkpoint t. After the native policy π proposes the next visual action but before its execution, CASE chooses STOP, invoking a constrained prefix finalizer, or

CONTINUE, executing that proposal. Complete native trajectories provide training supervision; deployment uses only the current causal state.

This setting follows the fixed-policy stopping principle (Shiryaev, 2008; Novikov, 2010). Let $R_{\text{stop}}(\mathcal{F}_{i,t})$ be the expected loss of answering now and $Q^{\pi}_{\text{cont}}(\mathcal{F}_{i,t})$ the minimum expected loss achievable by continuing along π and stopping later. The ideal value is

$$V^{\pi}(\mathcal{F}_{i,t}) = \min\{R_{\text{stop}}(\mathcal{F}_{i,t}), Q^{\pi}_{\text{cont}}(\mathcal{F}_{i,t})\}\,. \tag{1}$$

The optimal comparison stops when the first term is no larger than the second. Here the continuation cost is state-dependent: the next tool, temporal window, decoded frames, context, and model output are generated as the trajectory unfolds. The stopping decision must therefore value the expected evidence gain together with the full computation needed to obtain it.

3.2 Targets from complete native trajectories

The conditional continuation value in Equation 1 averages over future trajectories; we use the best realized loss on each recorded suffix as a hindsight surrogate (Appendix A.3). Let $a_{i,t}$ and $C_f(i,t)$ be the prefix-finalizer answer and cost at checkpoint t, S the mean native training-trajectory cost, and λ the computation weight. Stopping now has loss

$$L_{\text{stop}}(i,t) = \mathbb{1}[a_{i,t} \neq Y_i] + \lambda \frac{C_f(i,t)}{S}. \tag{2}$$

For a later checkpoint u, let $C(i, t \to u)$ include native model-token cost and intervening option assessments. Stopping at u has loss

$$L_{\text{future}}(i,t,u) = \mathbb{1}[a_{i,u} \neq Y_i] + \lambda \frac{C(i, t \to u) + C_f(i,u)}{S}. \tag{3}$$

The native terminal point is also a candidate, with loss

$$L_{\text{native}}(i,t) = \mathbb{1}[a_i^{\text{native}} \neq Y_i] + \lambda \frac{C(i, t \to T_i)}{S}. \tag{4}$$

The assessment at checkpoint t is sunk under either decision; subsequent assessments and answer-generation costs remain in the objective.

Let $\mathcal{U}_{i,t}$ contain the later intermediate checkpoints on the same trajectory. Making the cost weight explicit, we define the trajectory-level decision gap as

$$g^{(\lambda)}_{i,t} = L_{\text{stop}}(i,t) - \min(\{L_{\text{future}}(i,t,u) : u \in \mathcal{U}_{i,t}\} \cup \{L_{\text{native}}(i,t)\})\,, \tag{5}$$

A positive gap indicates a better realized later stopping point; a nonpositive gap favors answering now. Future labels and states construct this supervision only, while deployment uses the causal state defined next.

3.3 Causal stopping state

At each checkpoint, an external multiple-choice assessment reads the accumulated evidence and returns scores restricted to the question's original options. The assessment is generated at temperature zero, then temperature-scaled by checkpoint group using the training trajectories Guo et al. (2017). The assessment remains external to the native agent context, preserving the native search decisions.

We represent each checkpoint with an 18-dimensional causal vector $x_{i,t}$: eight answer-support statistics (including top probabilities, entropy, margin, and changes in option support), five progress/cost features (checkpoint, consumed model tokens, evidence length, video duration, and subtitles), and five tool-type indicators. Sorting the top probabilities removes option-label dependence, while temporal changes remain aligned by option identity. VideoSeek supplies tool types directly; AVP uses the shared none state for host-specific actions. Every feature is available at or before the current checkpoint.

3.4 LEARNING AND EXECUTION

The target admits an exact decomposition that makes the role of continuation cost explicit. Let

$$q_{i,t} = g_{i,t}^{(0)}, \qquad c_{i,t}^{(\lambda)} = \frac{g_{i,t}^{(\lambda)} - g_{i,t}^{(0)}}{\lambda}, \quad \lambda > 0, \tag{6}$$

so that $g_{i,t}^{(\lambda)} = q_{i,t} + \lambda c_{i,t}^{(\lambda)}$. The quality term $q_{i,t}$ compares current answer error with the best later answer error without charging computation. The second term is a cost-induced correction: because the best later point is chosen after answer risk and cost are combined, it captures both heterogeneous continuation cost and any cost-induced change in which later stopping point is preferred.

We fit both terms with the same weighted Ridge operator. For a supervision target $y_{i,t}$, where y is either q or $c^{(\lambda)}$, and trajectory i contains m_i learnable checkpoints, the model solves

$$(\hat{w}_y, \hat{\beta}_y) = \arg\min_{w,\beta} \sum_i \frac{1}{m_i} \sum_{t=0}^{m_i-1} \left(y_{i,t} - w^\top \widetilde{x}_{i,t} - \beta\right)^2 + \alpha \|w\|_2^2, \tag{7}$$

where $\widetilde{x}_{i,t}$ uses weighted training-set standardization and $\alpha = 10$. The $1/m_i$ weights equalize total influence across trajectories of different lengths. Regressing the decision gap aligns the learned quantity with the downstream stopping comparison Elmachtoub & Grigas (2021); the vision-language model, host agent, and tools remain fixed.

Writing the two fitted components as $\hat{q}(x)$ and $\hat{c}_\lambda(x)$, the deployed score is

$$\hat{g}_{i,t}^{(\lambda)} = \hat{q}(x_{i,t}) + \lambda \hat{c}_\lambda(x_{i,t}). \tag{8}$$

By linearity of weighted Ridge in its target, this factorized fit is exactly equivalent to fitting $g^{(\lambda)}$ directly, so the two components collapse to one affine deployment score. Appendix E.2 verifies the equivalence and isolates the cost correction.

At an eligible checkpoint, the decision is

$$d_{i,t} = \begin{cases} \text{STOP}, & \hat{g}_{i,t}^{(\lambda)} \le b, \\ \text{CONTINUE}, & \hat{g}_{i,t}^{(\lambda)} > b. \end{cases} \tag{9}$$

The cost weight λ and boundary b are selected on calibration videos. A STOP decision invokes a constrained prefix finalizer using the same underlying VLM and the evidence accumulated up to the current checkpoint, with decoding restricted to the original answer options. A CONTINUE decision executes the proposed visual action unchanged.

4 EXPERIMENTS

4.1 SETUP

Data and hosts. We evaluate Video-MME, LongVideoBench, and multiple-choice MLVU (Fu et al., 2025; Wu et al., 2024; Zhou et al., 2025). Video-MME is split by video into 50/100/750 training/calibration/evaluation videos (150/300/2,250 questions); each agent–model pair fits on the training trajectories, selects its operating point on calibration, and transfers the frozen policy zero-shot to LongVideoBench and MLVU. We pair VideoSeek and AVP with Qwen3.5-35B-A3B-FP8, GLM-4.6V-Flash, and InternVL3.5-30B-A3B-HF. For each host, Native, CASE, and all stopping baselines share the same host configuration; CASE intervenes only in termination, leaving native evidence acquisition unchanged. CASE acts on Medium/Long videos, with Short videos left native, and each agent–model pair has its own Ridge regressor and calibrated operating point.

Selection and accounting. For CASE, we search $\lambda \in \{0.05, 0.1, 0.2, 0.4, 0.8\}$ and 81 boundaries. The shared selector first requires at least 55% model-token saving and an accuracy lower confidence bound no worse than -1 pp, then maximizes saving among candidates within 2 pp of the best feasible accuracy; if none is feasible, it selects closest to 40% saving. The same rule applies to all tunable methods (Appendix D). We report end-to-end accuracy and accuracy change, model-token saving, and visual-token saving relative to Native. Model tokens include text input/output and visual tokens; option assessments, prefix finalizers, and native computation for continuing episodes are fully charged.

Baselines. We compare with six stopping strategies. *Tuned checkpoint* uses one global checkpoint, while *tuned budget* stops at a cumulative model-token budget. *Top-option confidence* and *option entropy* threshold the external multiple-choice assessment (Geifman & El-Yaniv, 2019; Schuster et al., 2022). *TASR-Video* adapts TASR to the option-level probe, combining consecutive-answer agreement with a calibrated top-two margin Kieback et al. (2026). *Forward-Q* implements a Monte Carlo continuation baseline following Stop-RAG: separate Ridge predictors estimate current-prefix correctness and the best later correctness, and their difference is thresholded Park et al. (2025). Each checkpoint, budget, margin, or threshold is selected on Video-MME with the same calibration rule as CASE. All methods use the same activation schedule, prefix finalizer, and cost ledger.

4.2 Main results

Table 1 reports end-to-end results relative to native search. Under a common-terminal-finalizer control, CASE saves 55.24% of model tokens with a 1.92-point accuracy decrease relative to Native with the same constrained terminal finalizer applied at the terminal state (Native+F); Appendix C.4 compares the seven stopping methods under this protocol.

Table 1: **Main results.** Model/visual tokens per question (k) and end-to-end accuracy are Native → CASE; savings are relative to Native. Bold marks lower token use or higher accuracy.

Benchmark	Agent	Model	M tok. (k)	M save	V tok. (k)	V save	Acc. (%)	Δ (pp)
Video-MME	VideoSeek	Qwen3.5	57.2 → **27.2**	52.43	12.0 → **5.1**	57.64	51.82 → **53.73**	+1.91
Video-MME	VideoSeek	GLM-4.6V	64.3 → **32.3**	49.77	22.4 → **9.4**	58.07	**56.40** → 55.69	-0.71
Video-MME	VideoSeek	InternVL3.5	49.5 → **14.9**	69.85	19.0 → **4.4**	77.06	**52.84** → 52.67	-0.18
Video-MME	AVP	Qwen3.5	51.9 → **22.5**	56.67	14.9 → **4.5**	69.67	65.02 → **65.29**	+0.27
Video-MME	AVP	GLM-4.6V	55.5 → **38.5**	30.72	14.6 → **9.9**	32.36	**57.91** → 57.51	-0.40
Video-MME	AVP	InternVL3.5	81.5 → **30.7**	62.35	49.7 → **17.3**	65.26	51.11 → **54.27**	+3.16
LongVideoBench	VideoSeek	Qwen3.5	79.5 → **28.4**	64.24	16.9 → **6.1**	64.03	40.91 → **44.13**	+3.22
LongVideoBench	VideoSeek	GLM-4.6V	98.3 → **44.4**	54.87	35.8 → **13.8**	61.39	**47.49** → 47.34	-0.15
LongVideoBench	VideoSeek	InternVL3.5	48.1 → **12.5**	74.08	21.2 → **3.1**	85.39	42.11 → **46.00**	+3.89
LongVideoBench	AVP	Qwen3.5	55.3 → **24.4**	55.83	11.7 → **3.4**	71.40	59.01 → **61.71**	+2.69
LongVideoBench	AVP	GLM-4.6V	62.2 → **40.2**	35.43	10.8 → **6.2**	42.33	51.31 → **54.38**	+3.07
LongVideoBench	AVP	InternVL3.5	79.4 → **25.2**	68.23	39.7 → **9.7**	75.43	38.22 → **45.77**	+7.55
MLVU	VideoSeek	Qwen3.5	47.7 → **19.3**	59.46	15.0 → **4.6**	69.53	45.58 → **47.88**	+2.30
MLVU	VideoSeek	GLM-4.6V	65.9 → **35.9**	45.52	28.4 → **13.2**	53.51	**50.23** → 49.95	-0.28
MLVU	VideoSeek	InternVL3.5	68.5 → **31.4**	54.20	48.2 → **19.3**	59.96	44.99 → **50.87**	+5.89
MLVU	AVP	Qwen3.5	36.4 → **12.5**	65.70	12.6 → **2.3**	81.88	55.43 → **59.89**	+4.46
MLVU	AVP	GLM-4.6V	43.5 → **29.9**	31.19	18.7 → **11.5**	38.73	53.13 → **53.54**	+0.41
MLVU	AVP	InternVL3.5	49.0 → **23.7**	51.58	33.8 → **15.8**	53.26	34.08 → **48.76**	+14.67

Across Video-MME, LongVideoBench, and MLVU, CASE changes end-to-end accuracy by $+0.67$, $+3.38$, and $+4.58$ points while saving 53.63%, 58.78%, and 51.28% of model tokens, respectively; LongVideoBench and MLVU use the same frozen Video-MME policies. Visual-token savings are 60.01%, 66.66%, and 59.48%.

4.3 Comparison with stopping baselines

Table 2 compares the six stopping baselines with CASE under the same Video-MME calibration rule. The two blocks report end-to-end accuracy change and model-token saving for VideoSeek and AVP.

The baselines occupy different parts of the accuracy–saving plane: some favor accuracy, while others stop more aggressively. The next analysis summarizes these two objectives jointly using Pareto dominance.

4.4 Accuracy–Efficiency Frontier Analysis

Inference-efficient systems form quality–compute Pareto frontiers (Bian et al., 2025; Saad-Falcon et al., 2025). For evaluation configuration j, method m gives $z_{jm} = (\Delta A_{jm}, S_{jm})$, where ΔA is

Table 2: **Stopping baselines.** All methods use the same Video-MME calibration rule. Entries are end-to-end accuracy change (Δ, pp), model-token saving (M, %), and visual-token saving (V, %). LVB and MLVU use the frozen Video-MME operating point; bold marks the best value, underlining the second-best, and blue shading marks CASE.

(a) VideoSeek

Model	Method	Video-MME			LongVideoBench			MLVU		
		M	V	Δ	M	V	Δ	M	V	Δ
Qwen3.5	Tuned checkpoint	42.57	42.17	+2.00	57.15	52.11	+3.89	59.14	62.99	+2.39
Qwen3.5	Tuned budget	36.73	34.58	**+2.36**	51.99	45.10	**+4.19**	42.34	48.70	+2.16
Qwen3.5	Confidence	36.13	42.66	+1.69	37.57	39.57	+2.69	48.31	57.58	+2.21
Qwen3.5	Entropy	33.72	40.13	+1.51	42.52	44.58	+2.09	43.84	53.71	+2.58
Qwen3.5	TASR-Video	37.65	46.10	+1.56	52.77	54.10	+3.29	52.79	65.26	**+3.36**
Qwen3.5	Forward-Q	38.64	43.25	+0.93	50.47	49.59	+3.74	32.91	43.63	+2.44
Qwen3.5	CASE	**52.43**	**57.64**	+1.91	**64.24**	**64.03**	+3.22	**59.46**	**69.53**	+2.30
GLM-4.6V	Tuned checkpoint	30.51	29.92	+0.22	32.82	31.95	+0.15	31.77	30.95	**+0.55**
GLM-4.6V	Tuned budget	32.36	33.06	**+0.36**	35.14	34.54	**+0.45**	31.48	36.97	+0.41
GLM-4.6V	Confidence	34.64	43.10	−2.09	32.51	36.21	−2.09	**56.83**	**62.83**	−5.66
GLM-4.6V	Entropy	32.53	42.61	−1.82	36.92	40.14	−2.99	51.06	56.90	−5.24
GLM-4.6V	TASR-Video	26.86	42.77	−0.36	37.75	50.59	−0.75	44.08	58.41	−0.18
GLM-4.6V	Forward-Q	43.49	50.08	−0.13	50.19	56.58	+0.22	32.00	39.67	+0.46
GLM-4.6V	CASE	**49.77**	**58.07**	−0.71	**54.87**	**61.39**	−0.15	45.52	53.51	−0.28
InternVL3.5	Tuned checkpoint	44.72	41.88	+0.80	36.23	34.31	+3.59	38.84	40.72	+3.08
InternVL3.5	Tuned budget	28.01	20.86	+0.13	18.70	14.37	+0.22	21.74	21.94	+1.10
InternVL3.5	Confidence	33.90	38.85	+0.18	10.71	21.03	+2.47	35.87	40.06	+2.71
InternVL3.5	Entropy	30.91	35.50	+0.22	12.13	21.64	+2.69	30.29	34.14	+2.02
InternVL3.5	TASR-Video	48.30	55.65	**+1.60**	38.36	50.93	**+4.71**	52.17	**60.69**	+5.47
InternVL3.5	Forward-Q	54.53	59.96	+1.51	46.82	54.88	+4.04	35.38	40.29	+3.40
InternVL3.5	CASE	**69.85**	**77.06**	−0.18	**74.08**	**85.39**	+3.89	**54.20**	59.96	**+5.89**

(b) AVP

Model	Method	Video-MME			LongVideoBench			MLVU		
		M	V	Δ	M	V	Δ	M	V	Δ
Qwen3.5	Tuned checkpoint	25.27	32.53	**+1.96**	23.75	32.47	**+4.41**	31.23	40.49	**+5.75**
Qwen3.5	Tuned budget	**64.11**	**73.24**	−0.49	**72.06**	**80.59**	−2.54	**71.89**	**87.01**	+4.78
Qwen3.5	Confidence	28.04	38.97	−0.76	9.49	24.66	+0.07	28.99	39.89	−0.64
Qwen3.5	Entropy	29.59	40.23	−1.11	13.30	28.26	+0.00	29.57	40.45	−0.51
Qwen3.5	TASR-Video	45.72	61.45	+0.76	39.85	60.66	+3.66	57.53	74.50	+5.47
Qwen3.5	Forward-Q	54.11	67.36	+0.49	53.15	68.72	+2.99	61.00	77.04	+4.65
Qwen3.5	CASE	56.67	69.67	+0.27	55.83	71.40	+2.69	65.70	81.88	+4.46
GLM-4.6V	Tuned checkpoint	23.01	26.06	−0.13	24.13	27.77	**+3.96**	33.01	38.07	+1.70
GLM-4.6V	Tuned budget	30.00	29.02	**+0.09**	**39.05**	40.06	+3.66	26.57	32.41	**+1.89**
GLM-4.6V	Confidence	27.24	32.71	−0.71	21.04	29.52	+1.65	44.83	52.11	+0.28
GLM-4.6V	Entropy	27.79	33.22	−0.80	25.15	32.98	+1.65	**45.72**	53.13	−0.23
GLM-4.6V	TASR-Video	24.41	**34.00**	−0.40	20.43	34.52	+1.80	43.48	**53.89**	+1.15
GLM-4.6V	Forward-Q	**30.76**	32.60	−0.44	35.36	42.33	+3.07	32.92	40.56	+0.55
GLM-4.6V	CASE	30.72	32.36	−0.40	35.43	**42.33**	+3.07	31.19	38.73	+0.41
InternVL3.5	Tuned checkpoint	54.30	56.13	**+4.04**	58.66	62.90	+8.90	56.56	56.75	+14.03
InternVL3.5	Tuned budget	**67.70**	**69.63**	+1.69	**78.13**	**84.14**	+6.58	**61.49**	**62.76**	+14.12
InternVL3.5	Confidence	28.35	32.40	+0.58	14.36	22.33	+3.89	29.88	33.64	+7.27
InternVL3.5	Entropy	25.68	29.70	+0.98	16.34	24.45	+3.89	26.08	29.73	+5.75
InternVL3.5	TASR-Video	46.07	51.06	+3.87	48.28	58.47	**+9.65**	48.30	50.36	+14.54
InternVL3.5	Forward-Q	59.85	62.91	+3.73	64.44	71.82	+7.93	51.90	53.81	+14.49
InternVL3.5	CASE	62.35	65.26	+3.16	68.23	75.43	+7.55	51.58	53.26	**+14.67**

end-to-end accuracy change and S is Model Token saving. A point is Pareto-optimal if no compared method is at least as good in both objectives and strictly better in one.

We summarize how consistently a method reaches this empirical nondominated frontier with *Pareto-frontier coverage* (PFC):

$$\mathrm{PFC}(m) = \frac{1}{J}\sum_{j=1}^{J} \mathbb{1}[m \in \mathcal{P}_j], \qquad J = 18, \tag{10}$$

Table 3: **Accuracy–efficiency frontier summary.** Each benchmark contains six agent–model configurations. A method is Pareto-optimal when no compared method jointly improves accuracy change and Model Token saving. PFC is the percentage of all 18 configurations in which the method is Pareto-optimal; Pareto rank is obtained by successive nondominated sorting (lower is better). Bold marks the best value and underlining the second-best.

Method	Mean M (%)	Mean Δ (pp)	Pareto-optimal agent–model pairs VMM	LVB	MLVU	Overall /18	PFC (%)	Mean rank ↓
Tuned checkpoint	39.09	**+3.41**	3/6	4/6	**4/6**	11/18	61.1	1.44
Tuned budget	44.97	+2.29	5/6	**5/6**	3/6	13/18	72.2	1.72
Confidence	31.04	+0.76	0/6	0/6	2/6	2/18	11.1	2.72
Entropy	30.73	+0.59	0/6	0/6	2/6	2/18	11.1	2.67
TASR-Video	42.49	+3.29	2/6	3/6	**4/6**	9/18	50.0	1.67
Forward-Q	46.00	+3.00	5/6	4/6	2/6	11/18	61.1	1.56
CASE	**54.56**	+2.88	**6/6**	**5/6**	**4/6**	**15/18**	**83.3**	**1.22**

where $\mathcal{P}_j$ is the nondominated set. We also report mean rank from successive nondominated sorting; rank 1 is best. Table 3 gives benchmark-wise coverage and ranks.
CASE has the highest PFC at 83.3% (15/18; next-best Tuned budget: 72.2%), the largest mean model-token saving (54.56%), and the lowest mean Pareto rank (1.22). It is Pareto-optimal for 6/6 Video-MME, 5/6 LongVideoBench, and 4/6 MLVU agent–model pairs; only five of 108 pairwise CASE–baseline comparisons dominate it on both objectives. Across 10,000 paired video-cluster bootstrap draws with frozen operating points, CASE has the highest or tied-highest PFC in 97.5% of draws and is strictly highest in 90.0% (Appendix D.4).

4.5 Long-Trajectory Analysis

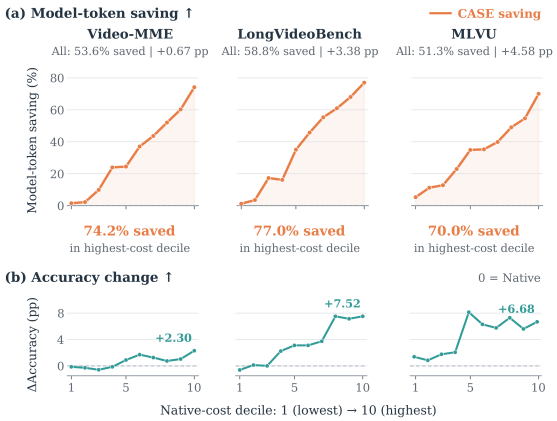

Figure 3: **Savings concentrate on expensive Native trajectories.** Native-cost deciles run from 1 (lowest) to 10 (highest); curves macro-average the six agent–model cells per benchmark.

The native computation tail is concentrated in a small set of expensive trajectories. Figure 3 ranks questions by Native model-token cost within each agent–model–benchmark cell and divides them into ten approximately equal deciles. The saving rises sharply toward the expensive end: in the highest-cost decile, CASE saves 74.2%, 77.0%, and 70.0% of model tokens on Video-MME, LongVideoBench, and MLVU, with end-to-end accuracy changes of +2.30, +7.52, and +6.68 points. The complementary post-hoc Top20% and Top10% slices, including all agent–model cells, are reported in Appendix C.5.

4.6 Ablations

Table 4 treats each ablation as a complete stopping method, selected on the same Video-MME training/calibration roles and transferred frozen to LongVideoBench and MLVU. Variants remove answer-

support features, execution-state features, continuation-aware supervision, or the heterogeneous-cost correction.

Table 4: **End-to-end component and target ablations.** Variants are independently selected on Video-MME calibration and transferred zero-shot to LongVideoBench and MLVU. Entries are mean accuracy change (Δ, pp) and model-token saving (M, %) over six agent–model cells; Overall averages all 18 configurations and Strict counts primary-branch selections (/6). Bold/underline mark best/second-best.

Variant	Video-MME		LongVideoBench		MLVU		Overall		Strict
	M	Δ	M	Δ	M	Δ	M	Δ	/6
Full	<u>53.63</u>	<u>+0.67</u>	<u>58.78</u>	<u>+3.38</u>	**51.28**	**+4.58**	<u>54.56</u>	<u>+2.88</u>	**5/6**
Belief-only	36.90	+0.38	33.46	+2.49	<u>49.28</u>	+2.70	39.88	+1.86	2/6
Exec-state-only	**55.21**	+0.09	**64.48**	+2.26	46.20	+4.27	**55.30**	+2.21	<u>4/6</u>
Current-corr.	35.99	−0.19	31.55	+2.02	40.57	+1.34	36.04	+1.06	1/6
w/o cost correction	46.90	**+1.01**	50.07	**+3.66**	41.02	<u>+4.33</u>	46.00	**+3.00**	<u>4/6</u>

Full gives +2.88 points in end-to-end accuracy / 54.56% model-token saving; without the cost correction it gives +3.00/46.00%, showing that the cost term mainly shifts selection toward greater saving. Appendix E gives detailed controls.

4.7 Online Execution

In online execution on 900 questions per benchmark and agent–model pair, every STOP changes the subsequent trajectory. The favorable accuracy–efficiency trade-off persists: across all 12 cells, CASE averages +1.56 points with 54.50% model-token saving, while measured runtime additionally falls by 54.10% (Figure 4). The close agreement between token and runtime reductions provides an execution-time check on the token-based efficiency metric: Video-MME averages 52.9% model-token versus 52.5% runtime saving, and LongVideoBench 56.1% versus 55.7%.

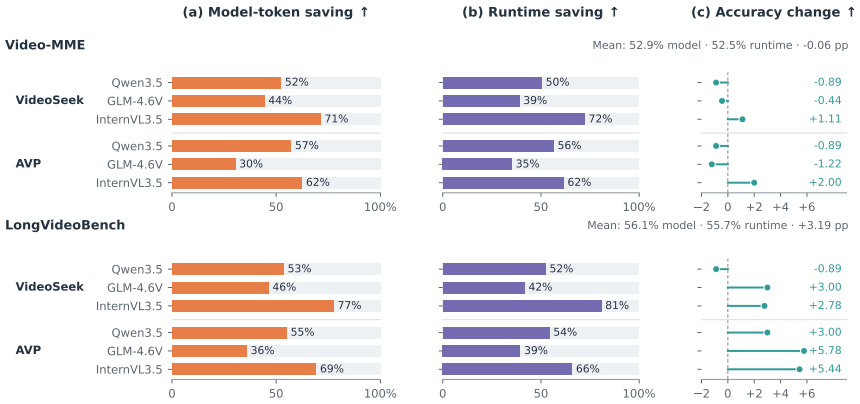

Figure 4: **Online execution.** End-to-end accuracy change and savings in model tokens and measured runtime; exact values are in Appendix Table 7.

5 Conclusion

Long-video agents must decide both where to seek evidence and when further search is no longer worth its cost. CASE learns this termination decision from complete native trajectories while leaving the host search policy and VLM unchanged. Across two agents, three VLMs, and three benchmarks, it reduces model-token use by 54.6% on average and attains the highest PFC (83.3%) among the compared stopping methods; frozen policies transfer zero-shot across benchmarks. Online execution preserves the same favorable accuracy–efficiency trade-off and additionally reduces measured runtime by 54.1% on average. Together, these results show that explicit stopping is a practical complement to better evidence acquisition: long-video agents can retain their native search policy while learning when further search is no longer worth its cost.

AI USE STATEMENT

We used generative AI tools to assist with literature discovery, experimental design, implementation and debugging, and manuscript polishing.

We did not use generative AI to generate synthetic training or evaluation datasets. The experimental results reported in this paper were obtained from executed experiments and analyses of their outputs.

AI-assisted implementations were tested, numerical claims were checked against experimental artifacts, and references were verified against primary sources.

REFERENCES

Ingeol Baek, Hwan Chang, ByeongJeong Kim, Jimin Lee, and Hwanhee Lee. Probing-RAG: Self-probing to guide language models in selective document retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3287–3304, 2025. doi: 10.18653/v1/2025.findings-naacl.181.

Andrea Banino, Jan Balaguer, and Charles Blundell. Pondernet: Learning to ponder. In *8th ICML Workshop on Automated Machine Learning*, 2021. URL https://arxiv.org/abs/2107.05407.

Song Bian, Minghao Yan, and Shivaram Venkataraman. Scaling inference-efficient language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 4303–4323. PMLR, 2025. URL https://proceedings.mlr.press/v267/bian25b.html.

Xinshi Chen, Hanjun Dai, Yu Li, Xin Gao, and Le Song. Learning to stop while learning to predict. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1520–1530. PMLR, 2020. URL https://proceedings.mlr.press/v119/chen20c.html.

Herman Chernoff. Sequential design of experiments. *The Annals of Mathematical Statistics*, 30(3): 755–770, 1959. doi: 10.1214/aoms/1177706205.

Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize". *Management Science*, 68(1): 9–26, 2021. doi: 10.1287/mnsc.2020.3922.

Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. Videoagent: A memory-augmented multimodal agent for video understanding. In *Computer Vision – ECCV 2024*, volume 15080 of *Lecture Notes in Computer Science*, pp. 75–92. Springer, 2024. doi: 10.1007/978-3-031-72670-5_5. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/3241_ECCV_2024_paper.php.

Yicheng Feng, Yan Zhang, Yan Cheng, and Wei Qi. Scores are not decisions: Cost-aware stopping for tool acquisition in llm agents. *arXiv preprint arXiv:2607.27083*, 2026.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24108–24118, 2025.

Yonatan Geifman and Ran El-Yaniv. Selectivenet: A deep neural network with an integrated reject option. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2151–2159. PMLR, 2019. URL https://proceedings.mlr.press/v97/geifman19a.html.

Alex Graves. Adaptive computation time for recurrent neural networks. *arXiv preprint arXiv:1603.08983*, 2016.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330. PMLR, 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

Metod Jazbec, Alexander Timans, Tin Hadži Veljkovič, Kaspar Sakmann, Dan Zhang, Christian A. Naesseth, and Eric Nalisnick. Fast yet safe: Early-exiting with risk control. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-4124. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/ea5a63f7ddb82e58623693fd1f4933f7-Abstract-Conference.html.

Jindong Jiang, Xiuyu Li, Zhijian Liu, Muyang Li, Guo Chen, Zhiqi Li, De-An Huang, Guilin Liu, Zhiding Yu, Kurt Keutzer, Sungjin Ahn, Jan Kautz, Hongxu Yin, Yao Lu, Song Han, and Wonmin Byeon. Storm: Token-efficient long video understanding for multimodal llms, 2025. URL https://arxiv.org/abs/2503.04130.

Adrian Kieback, Uyiosa Philip Amadasun, Aman Chadha, and Aaron Elkins. Tasr: Training-free adaptive stopping for iterative retrieval. In *KDD 2026 Workshop on Agentic AI for Information Retrieval (Agent4IR)*, 2026. URL https://arxiv.org/abs/2606.13814.

Chenglin Li, Qianglong Chen, Feng Han, Yikun Wang, Xingxi Yin, Yan Gong, Ruilin Li, Yin Zhang, and Jiaqi Wang. Videothinker: Building agentic videollms with llm-guided tool reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 8226–8236, 2026.

Jingyang Lin, Jialian Wu, Jiang Liu, Ximeng Sun, Ze Wang, Xiaodong Yu, Jiebo Luo, Zicheng Liu, and Emad Barsoum. Videoseek: Long-horizon video agent with tool-guided seeking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2603.20185.

Runtao Liu, Ziyi Liu, Jiaqi Tang, Yue Ma, Renjie Pi, Jipeng Zhang, and Qifeng Chen. Longvideoagent: Multi-agent reasoning with long videos. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 40404–40416, 2026. doi: 10.18653/v1/2026.acl-long.1876. URL https://aclanthology.org/2026.acl-long.1876/.

Mohammad Naghshvar and Tara Javidi. Active sequential hypothesis testing. *The Annals of Statistics*, 41(6):2703–2738, 2013. doi: 10.1214/13-AOS1144.

Sirin Nitinawarat and Venugopal V. Veeravalli. Controlled sensing for sequential multihypothesis testing with controlled markovian observations and non-uniform control cost. *Sequential Analysis*, 34(1):1–24, 2015. doi: 10.1080/07474946.2014.961864.

Andrey Novikov. Optimal sequential procedures with bayes decision rules. *Kybernetika*, 46(4): 754–770, 2010. URL https://eudml.org/doc/196417.

Junwen Pan, Qizhe Zhang, Rui Zhang, Ming Lu, Xin Wan, Yuan Zhang, Chang Liu, and Qi She. Timesearch-r: Adaptive temporal search for long-form video understanding via self-verification reinforcement learning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=gqb1hvuGcj.

Jaewan Park, Solbee Cho, and Jay-Yoon Lee. Stop-rag: Value-based retrieval control for iterative rag. In *NeurIPS 2025 Workshop on Multi-Turn Interactions in Large Language Models (MTI-LLM)*, 2025. URL https://openreview.net/forum?id=fN4PJNYgs3.

Haozhe Qi, Kevin Qu, Mahdi Rad, Rui Wang, Alexander Mathis, and Marc Pollefeys. Adapttoken: Entropy-based adaptive token selection for mllm long video understanding, 2026. URL https://arxiv.org/abs/2603.28696.

Chenhao Qiu, Yechao Zhang, Xin Luo, Shien Song, and Xusheng Liu. Videoseal: Mitigating evidence misalignment in agentic long video understanding by decoupling answer authority, 2026a. URL https://arxiv.org/abs/2605.12571. Accepted to the 43rd International Conference on Machine Learning (ICML 2026).

Jihao Qiu, Lingxi Xie, Xinyue Huo, Qi Tian, and Qixiang Ye. Longvideo-r1: Smart navigation for low-cost long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40505–40515, 2026b.

Jon Saad-Falcon, Adrian Gamarra Lafuente, Shlok Natarajan, Nahum Maru, Hristo Todorov, Etash Kumar Guha, E. Kelly Buchanan, Mayee F. Chen, Neel Guha, Christopher Re, and Azalia Mirhoseini. An architecture search framework for inference-time techniques. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 52475–52507. PMLR, 2025. URL https://proceedings.mlr.press/v267/saad-falcon25a.html.

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Q. Tran, Yi Tay, and Donald Metzler. Confident adaptive language modeling. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17456–17472, 2022. doi: 10.52202/068431-1269. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/6fac9e316a4ae75ea244ddcef1982c71-Abstract-Conference.html.

Albert N. Shiryaev. *Optimal Stopping Rules*, volume 8 of *Stochastic Modelling and Applied Probability*. Springer, 2008. doi: 10.1007/978-3-540-74011-7.

J. Laurie Snell. Applications of martingale system theorems. *Transactions of the American Mathematical Society*, 73(2):293–312, 1952. doi: 10.2307/1990670.

Xi Tang, Jihao Qiu, Lingxi Xie, Yunjie Tian, Jianbin Jiao, and Qixiang Ye. Adaptive keyframe sampling for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 29118–29128, 2025.

George Vershinin, Asaf Cohen, and Omer Gurewitz. Active sequential hypothesis testing with non-homogeneous costs. *arXiv preprint arXiv:2509.11632*, 2025.

Abraham Wald. Sequential tests of statistical hypotheses. *The Annals of Mathematical Statistics*, 16 (2):117–186, 1945. doi: 10.1214/aoms/1177731118.

Xiao Wang, Qingyi Si, Shiyu Zhu, Jianlong Wu, Li Cao, and Liqiang Nie. Adaretake: Adaptive redundancy reduction to perceive longer for video-language understanding. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 5417–5432, 2025a. doi: 10.18653/v1/2025.findings-acl.283. URL https://aclanthology.org/2025.findings-acl.283/.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In *Computer Vision – ECCV 2024*, volume 15138 of *Lecture Notes in Computer Science*, pp. 58–76. Springer, 2024. doi: 10.1007/978-3-031-72989-8_4. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/10325_ECCV_2024_paper.php.

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. Videotree: Adaptive tree-based video representation for llm reasoning on long videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3272–3283, 2025b.

Ziyang Wang, Honglu Zhou, Shijie Wang, Junnan Li, Caiming Xiong, Silvio Savarese, Mohit Bansal, Michael S. Ryoo, and Juan Carlos Niebles. Active video perception: Iterative evidence seeking for agentic long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 9088–9099, 2026. URL https://openaccess.thecvf.com/content/CVPR2026F/html/Wang_Active_Video_Perception_Iterative_Evidence_Seeking_for_Agentic_Long_Video_CVPRF_2026_paper.html.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. In *Advances in Neural Information Processing Systems*, volume 37, pp. 28828–28857, 2024. doi: 10.52202/079017-0907. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/329ad516cf7a6ac306f29882e9c77558-Abstract-Datasets_and_Benchmarks_Track.html.

Baixuan Xu, Yinyui Xu, Tianshi Zheng, Zhaowei Wang, Weiqi Wang, Haochen Shi, Jiayu Liu, Qing Zong, Xiyu Ren, Xinyu Geng, Zhitao He, and Yangqiu Song. Finding the right evidence: Factor-guided coarse-to-fine reasoning for long videos, 2026. URL https://arxiv.org/abs/2608.26355.

Xiaoyi Zhang, Zhaoyang Jia, Zongyu Guo, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Deep video discovery: Agentic search with tool use for long-form video understanding. In *Advances in Neural Information Processing Systems*, volume 38, pp. 99802–99834, 2025. doi: 10.52202/085713-3005. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/8190b210e9808e54ee16263b673a847d-Abstract-Conference.html.

Zhuo Zhi, Qiangqiang Wu, Minghe Shen, Wenbo Li, Yinchuan Li, Kun Shao, and Kaiwen Zhou. Videoagent2: Enhancing the llm-based agent system for long-form video understanding by uncertainty-aware cot, 2025. URL https://arxiv.org/abs/2504.04471.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: Benchmarking multi-task long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13691–13701, 2025. doi: 10.1109/CVPR52734.2025.01278. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhou_MLVU_Benchmarking_Multi-task_Long_Video_Understanding_CVPR_2025_paper.html.

Wangchunshu Zhou, Canwen Xu, Tao Ge, Julian McAuley, Ke Xu, and Furu Wei. Bert loses patience: Fast and robust inference with early exit. In *Advances in Neural Information Processing Systems*, volume 33, pp. 18330–18341, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/d4dd111a4fd973394238aca5c05bebe3-Abstract.html.

Zirui Zhu, Hailun Xu, Yang Luo, Yong Liu, Kanchan Sarkar, Zhenheng Yang, and Yang You. Focus: Efficient keyframe selection for long video understanding. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/a010577f93a8301dada61f0b97cda923-Abstract-Conference.html.

APPENDIX

A FORMAL FORMULATION, PROPERTIES, AND PROOFS

We expand the stopping formulation in Section 3 and connect the trajectory-level target used by CASE to the corresponding policy-conditioned stopping problem. Throughout this section, the native evidence-acquisition policy is fixed and denoted by π; the stopping policy only chooses when to terminate the resulting evidence-acquisition process.

A.1 POLICY-CONDITIONED STOPPING PROBLEM

For question i, let $Y_i \in \{1, \dots, K_i\}$ be the correct option and let $\{\mathcal{F}_{i,t}\}_{t=0}^{T_i}$ be the information available along the native trajectory. The filtration grows as video-tool observations, interaction history, and consumed computation accumulate. A causal stopping time τ satisfies $\{\tau = t\} \in \mathcal{F}_{i,t}$ and may terminate at an intermediate checkpoint or at the native terminal state.

For an intermediate stop, $A_{i,t}$ denotes the prefix-finalizer answer. At the native terminal state it denotes the native answer. Let $C_i(t, \tau)$ be the incremental model-token cost from checkpoint t until answering at τ: it contains all native computation after t, every future option assessment, and the answer-generation cost at the stopping point. The conditional loss of a stopping time is

$$J_{i,t}^{\pi}(\tau) = \mathbb{E}\left[\mathbb{1}[A_{i,\tau} \neq Y_i] + \lambda \frac{C_i(t,\tau)}{S} \,\middle|\, \mathcal{F}_{i,t}\right], \tag{11}$$

where the expectation follows future trajectories generated by π. If $\mathcal{T}_{i,t}^{\pi}$ is the set of causal stopping times no earlier than t, the policy-conditioned value is

$$V_{i,t}^{\pi} = \inf_{\tau \in \mathcal{T}_{i,t}^{\pi}} J_{i,t}^{\pi}(\tau). \tag{12}$$

This is the fixed-policy optimal-stopping problem underlying Equation 1 (Shiryaev, 2008; Novikov, 2010). The multiple-choice structure determines the answer error, while the native agent induces the sequence of future evidence and computation on which the stopping decision operates.

A.2 Finite-horizon Bellman relation

Let $K_{i,t+1}$ denote the complete incremental cost incurred by choosing CONTINUE at checkpoint t and reaching the next decision state. It includes the native model/tool computation and the next option assessment. Because the next tool, temporal window, decoded frames, and generated context depend on the evolving agent state, $K_{i,t+1}$ is generally state dependent and random conditional on $\mathcal{F}_{i,t}$.

Proposition 1 (policy-conditioned Bellman recursion). For a finite native trajectory, the value in Equation 12 satisfies

$$V^{\pi}_{i,T_i} = R_{\mathrm{stop}}(\mathcal{F}_{i,T_i}), \qquad V^{\pi}_{i,t} = \min\left\{ R_{\mathrm{stop}}(\mathcal{F}_{i,t}), \mathbb{E}\left[\lambda \frac{K_{i,t+1}}{S} + V^{\pi}_{i,t+1} \,\middle|\, \mathcal{F}_{i,t} \right] \right\}. \tag{13}$$

Consequently, if

$$G^{\pi}_{i,t} = R_{\mathrm{stop}}(\mathcal{F}_{i,t}) - \mathbb{E}\left[\lambda \frac{K_{i,t+1}}{S} + V^{\pi}_{i,t+1} \,\middle|\, \mathcal{F}_{i,t} \right], \tag{14}$$

then the ideal decision stops when $G^{\pi}_{i,t} \le 0$ and continues when $G^{\pi}_{i,t} > 0$.

Proof. At the terminal state there is no continuation action, giving the first equality. At any earlier checkpoint, every admissible stopping time either stops immediately or chooses CONTINUE and subsequently follows an admissible stopping rule from the next state. The immediate branch has conditional risk $R_{\mathrm{stop}}(\mathcal{F}_{i,t})$; the continuation branch first pays $K_{i,t+1}$ and then incurs the optimal value from $t+1$. Taking the smaller conditional risk gives Equation 13. Equation 14 is the difference between these two branches, so its sign determines the minimizing action. □

The relation shows that stopping depends jointly on answer support, continuation cost, and the future evidence reachable from the current state. Two states with similar answer support can therefore have different continuation values when the next evidence-acquisition step has different expected cost or reaches different future evidence.

A.3 Reachable-suffix hindsight target

The conditional continuation value in Equation 13 averages over future trajectories, whereas training provides complete realized trajectories. For a recorded trajectory, define the best later realized loss

$$H_{i,t} = \min(\{ L_{\mathrm{future}}(i,t,u) : u \in \mathcal{U}_{i,t} \} \cup \{ L_{\mathrm{native}}(i,t) \}). \tag{15}$$

The training target in Equation 5 is therefore $g^{(\lambda)}_{i,t} = L_{\mathrm{stop}}(i,t) - H_{i,t}$. A positive target means that the observed suffix contains a later answer point with lower realized error-plus-cost loss.

Proposition 2 (hindsight lower bound). Let $\tau > t$ be any causal stopping rule restricted to the later checkpoints and native terminal state reachable on the recorded suffix. Then

$$\mathbb{E}[H_{i,t} \mid \mathcal{F}_{i,t}] \le \mathbb{E}[L_{i,t}(\tau) \mid \mathcal{F}_{i,t}], \tag{16}$$

and hence

$$\mathbb{E}[H_{i,t} \mid \mathcal{F}_{i,t}] \le \inf_{\tau \in \mathcal{T}^{\pi}_{i,t+1}} \mathbb{E}[L_{i,t}(\tau) \mid \mathcal{F}_{i,t}]. \tag{17}$$

Proof. For every realized suffix, $H_{i,t}$ is the minimum loss among all later candidate stopping locations on that suffix. The location selected by any causal τ is one member of the same candidate set, so $H_{i,t} \le L_{i,t}(\tau)$ pointwise. Conditional expectation preserves the inequality, and taking the infimum over causal τ gives Equation 17. □

This property explains the role of the complete trajectory in supervision. A recorded suffix reveals the best answer point reachable on that realization, while Equation 13 describes the conditional

expectation over future trajectories. The resulting $g_{i,t}^{(\lambda)}$ is a cost-complete reachable-suffix surrogate for the policy-conditioned decision gap. Because its minimizing checkpoint is chosen after the suffix is observed, its conditional continuation loss can be lower than that of a causal stopping rule, which is the standard hindsight advantage captured by Equation 17.

A.4 Cost cancellation at a checkpoint

The stopping decision is made after the current option assessment has been computed. Let $C_{i,t}^{\mathrm{sunk}}$ collect all native computation and assessments already paid before the STOP/CONTINUE comparison. If the agent stops at t, the total episode cost can be written as

$$C_{i,t}^{\mathrm{stop}} = C_{i,t}^{\mathrm{sunk}} + C_f(i,t). \tag{18}$$

If it continues to a later checkpoint u, the corresponding cost is

$$C_{i,t\to u}^{\mathrm{cont}} = C_{i,t}^{\mathrm{sunk}} + C(i,t\to u) + C_f(i,u), \tag{19}$$

where $C(i,t\to u)$ contains future native computation and future option assessments. Subtracting the two losses removes $C_{i,t}^{\mathrm{sunk}}$ exactly. The current assessment therefore cancels from the decision gap, while every future assessment, native step, and answer-generation cost remains part of the continuation comparison in Equations 2–3.

A.5 Invariance to option relabeling

The probability-derived part of the 18-dimensional state is invariant to the literal option labels. Let σ be any permutation of the K_i answer options and let $p_t^{\sigma}(k) = p_t(\sigma^{-1}(k))$ be the consistently relabeled option distribution. The option count, sorted top-three probabilities, normalized entropy, and top-two margin are unchanged by σ. The aligned total variation also satisfies

$$\frac{1}{2}\sum_{k=1}^{K_i}\left|p_t^{\sigma}(k) - p_{t-1}^{\sigma}(k)\right| = \frac{1}{2}\sum_{k=1}^{K_i}\left|p_t(k) - p_{t-1}(k)\right|, \tag{20}$$

and the indicator of whether the leading-option set changes between checkpoints is preserved under the same relabeling. The remaining progress, cost, video, subtitle, and tool-state features are label-independent. Hence the full feature vector $x_{i,t}$ is unchanged by a consistent permutation of the answer labels and supports three-, four-, and five-option questions with the same representation.

B CASE Algorithms and Causal Implementation

We now describe the complete data flow from native trajectories to a deployed stopping policy. The implementation follows the quantities defined in Section 3 and Appendix A: complete native trajectories provide training supervision, the learned score is calibrated on a disjoint development role, and deployment evaluates checkpoints in causal order using only information available at that point.

B.1 Target construction from a native trajectory

A native trajectory contains a sequence of nonterminal decision checkpoints followed by its native terminal state. For every nonterminal checkpoint, the training cache contains the constrained prefix-finalizer answer and its cost, while the native trajectory records the terminal answer and the cumulative native cost. Let m_i be the number of nonterminal decision checkpoints for trajectory i. The cost scale S is the mean full native model-token cost over training trajectories with $m_i > 0$.

Table 5: **Causal stopping state.** All quantities are available at or before the current checkpoint. Probability ranks are computed after temperature scaling; total variation is aligned by option identity.

Block	Dimensions	Definition
Answer support	1	Number of answer options K_i
	2–4	Three largest calibrated option probabilities, sorted descending
	5	Entropy of the calibrated distribution, normalized by $\log K_i$
	6	Difference between the largest and second-largest probabilities
	7	Total variation from the previous calibrated distribution, aligned by option identity
	8	Indicator that the set of probability-maximizing options changed from the previous checkpoint
Progress and cost	9	Checkpoint index
	10	Log of one plus model tokens consumed before the current planner call
	11	Log of one plus characters in accumulated visible evidence
	12	Log of one plus video duration in seconds
	13	Subtitle-availability indicator
Latest tool	14–18	One-hot state: overview, skim, focus, answer, or none

Algorithm 1 Decision-gap target construction for one cost weight λ

Input: Complete native trajectories; gold answers; prefix-finalizer answers and costs; cost scale S; cost weight λ

Output: Decision-gap targets $\{g_{i,t}^{(\lambda)}\}$

foreach *training trajectory* i **do**
  $m_i \leftarrow$ number of nonterminal decision checkpoints
  for $t \leftarrow 0$ **to** $m_i - 1$ **do**
    $L_{\text{stop}} \leftarrow L_{\text{stop}}(i,t)$ using Eq. 2
    $L_{\text{best}} \leftarrow L_{\text{native}}(i,t)$ using Eq. 4
    for $u \leftarrow t+1$ **to** $m_i - 1$ **do**
      $L_{\text{best}} \leftarrow \min\{L_{\text{best}}, L_{\text{future}}(i,t,u)\}$ using Eq. 3
    end for
    $g_{i,t}^{(\lambda)} \leftarrow L_{\text{stop}} - L_{\text{best}}$
  end for
end foreach

The option-probe cost at checkpoint t is shared by both branches and cancels inside $g_{i,t}^{(\lambda)}$; Appendix A.4 gives the corresponding algebra explicitly. Repeating the procedure for each $\lambda \in \{0.05, 0.1, 0.2, 0.4, 0.8\}$ produces one target set for each cost weight.

B.2 Causal state and policy fitting

The external option probe is evaluated at temperature zero and returns a log-probability for every valid answer option. On the 50 training videos, we fit one scalar temperature for each checkpoint-depth group 0, 1, 2, and 3+ by minimizing mean negative log-likelihood. The calibrated option distribution is then combined with execution information into the 18-dimensional state used by every agent–model stopping policy; Table 5 lists the complete state.

Algorithm 2 Training the family of Gap-Ridge stopping scores

Input: Training trajectories and prefix finalizers; raw option-probe log-probabilities; $\Lambda = \{0.05, 0.1, 0.2, 0.4, 0.8\}$; $\alpha = 10$
Output: Probe temperatures and fitted scores $\{f_\lambda\}_{\lambda \in \Lambda}$

Fit one temperature for checkpoint-depth groups $0, 1, 2, 3+$ by training-set negative log-likelihood
Apply the fitted temperatures and construct every causal feature vector $x_{i,t}$
Compute the model-token scale S from learnable training trajectories
foreach $\lambda \in \Lambda$ **do**
    Construct $g_{i,t}^{(\lambda)}$ for all training checkpoints using Algorithm 1
    Assign checkpoint weight $1/m_i$ within trajectory i
    Compute weighted feature means and scales and standardize $x_{i,t}$
    Fit weighted Ridge f_λ using Eq. 7
end foreach

The intercept is represented by the weighted target mean, while the coefficient penalty applies to the standardized feature weights. Probe calibration, feature construction, and Ridge fitting use only the training role.

The feature definition is shared across VideoSeek and AVP. The first eight dimensions summarize answer support with label-invariant statistics; the next five encode trajectory progress and accumulated computation; the final five expose the latest VideoSeek tool state when that taxonomy is defined. AVP uses the shared *none* state for this last block, as described in Section B.5.

B.3 Operating-point calibration

Calibration converts the five fitted score functions into one frozen operating point. For each λ, we evaluate 81 boundaries

$$b \in \{-0.200, -0.195, \ldots, 0.195, 0.200\},$$

with Medium+Long activation, giving $5 \times 81 = 405$ candidates per agent–model pair. Each candidate is replayed on the 100 Video-MME calibration videos using the same prefix finalizer and full cost accounting as deployment.

Algorithm 3 Unified operating-point calibration

Input: Calibration trajectories; fitted scores $\{f_\lambda\}$; boundary grid $\mathcal{B} = \{-0.200, -0.195, \ldots, 0.200\}$
Output: Frozen cost weight λ^*, boundary b^*, and score f_{λ^*}

$\mathcal{C} \leftarrow \emptyset$
foreach $\lambda \in \Lambda$ **do**
    foreach $b \in \mathcal{B}$ **do**
        Replay first-threshold stopping with (f_λ, b) on calibration videos
        Record accuracy change $\Delta_{\lambda,b}$, model-token saving $s_{\lambda,b}$, and video-cluster lower-bound diagnostic $\ell_{\lambda,b}$
        Add $(\lambda, b, \Delta_{\lambda,b}, s_{\lambda,b}, \ell_{\lambda,b})$ to $\mathcal{C}$
    end foreach
end foreach
$\mathcal{C}_{\text{strict}} \leftarrow \{c \in \mathcal{C} : s_c \geq 0.55,\ \ell_c \geq -0.01\}$
if $\mathcal{C}_{\text{strict}} \neq \emptyset$ **then**
    $\Delta^* \leftarrow \max_{c \in \mathcal{C}_{\text{strict}}} \Delta_c$
    $\mathcal{N} \leftarrow \{c \in \mathcal{C}_{\text{strict}} : \Delta_c \geq \Delta^* - 0.02\}$
    $(\lambda^*, b^*) \leftarrow \arg\max_{c \in \mathcal{N}} (s_c, \Delta_c, \ell_c, \text{index}_c)$ lexicographically
else
    $(\lambda^*, b^*) \leftarrow \arg\min_{c \in \mathcal{C}} (|s_c - 0.40|, -\ell_c, -\Delta_c, \text{index}_c)$ lexicographically
end if

The selected (λ, b) and the corresponding Ridge model are then frozen. The same frozen policy is applied to the 750 Video-MME evaluation videos and transferred zero-shot to LongVideoBench and MLVU. Section D reports calibration sensitivity to the 55% floor, 2-pp near-optimality band, and 40% fallback target.

B.4 ONLINE STOP/CONTINUE WRAPPER

A deployment checkpoint is attached to a native visual proposal immediately before the associated visual observation is executed. The planning computation that produced the proposal is already part of the sunk native cost, while the stopping state contains only evidence available before that future visual observation. The wrapper processes eligible checkpoints in temporal order.

Algorithm 4 Causal CASE execution

Input: Native long-video agent; frozen f_{λ^*} and boundary b^*; training-fitted probe temperatures
Output: Final multiple-choice answer

if *video is Short* **then**
  run native execution and **return** native terminal answer
end if
while *the native agent exposes a nonterminal decision checkpoint* t **do**
  Run the constrained option probe on currently visible evidence
  Apply the fitted depth-group temperature and construct $x_{i,t}$
  $\hat{g}_{i,t} \leftarrow f_{\lambda^*}(x_{i,t})$
  if $\hat{g}_{i,t} \leq b^*$ **then**
    Invoke the constrained prefix finalizer on the exact current prefix
    if *a valid option is returned* **then**
      return prefix-finalizer answer
    end if
  end if
  Execute the native visual action and resume the native agent
end while
return native terminal answer

Thus the deployed policy terminates at the first available threshold crossing. When it stops at checkpoint t, its episode cost is the native cost already sunk at t, plus every option probe visited through t, plus the stopping finalizer. When it reaches the native terminal, the cost is the full native episode plus all option probes visited by the stopping wrapper. These are the same costs used during calibration and offline replay.

B.5 AGENT-SPECIFIC CHECKPOINT ADAPTERS

The two host agents expose different internal trajectories, but both are converted to the same checkpoint interface: accumulated evidence, a native cost view before the next visual observation, a sequence position, and a terminal state.

VideoSeek. VideoSeek records causal planner turns and their completed tool observations. A nonterminal checkpoint corresponds to a planner turn that proposes a visual action. The visible state contains the question and options, subtitles available to the native agent, and the observations returned by all completed earlier tool calls. The feature cost uses the native computation consumed before the current planner call, while the decision ledger separately records the planner cost that has already been incurred before the proposed visual action is executed. The latest-tool block records the most recent VideoSeek visual tool among overview, skim, and focus (with the answer/none states retained in the common encoding). For a stopped prefix, the finalizer reconstructs the native conversation up to the accumulated evidence at that checkpoint and requests a constrained multiple-choice answer.

AVP. AVP alternates planning with query-conditioned visual observations. A nonterminal checkpoint is created after a valid AVP planning proposal and immediately before its associated visual inference. The visible state contains the question and options, video duration, available subtitles, and all completed AVP observation rounds, including their returned evidence summaries. Backend recovery calls belonging to one logical observation remain grouped under the same planner proposal and form one stopping state. The common tool block reserves overview/skim/focus for VideoSeek, while AVP uses the shared *none* state. The terminal checkpoint records the complete accrued native cost and terminal answer status.

These adapters leave the learned interface unchanged: both produce the same 18-dimensional state, use the same target construction and Ridge learner, and apply the same first-threshold-crossing rule.

B.6 Option probe, prefix finalizer, and parsing

Both auxiliary calls use the same VLM backend as the corresponding host execution and are generated at temperature zero. They operate only on the question's original three to five answer choices.

Option probe. For VideoSeek, the probe receives the question, the subtitles visible to the native policy, and the accumulated completed tool evidence. For AVP, it receives the rendered visible state described above. Its system instruction is:

Option-probe system prompt

```
Answer the multiple-choice question using only the supplied visible
evidence.  Return exactly one allowed uppercase option letter.
```

The call is restricted to the valid option letters, generates one token, and requests token log-probabilities. A probe record is accepted only when a log-probability is available for every valid option; these values are subsequently temperature-scaled into the calibrated distribution used by the stopping state.

Prefix finalizer. The multiple-choice query lists the original options as `(A)`, `(B)`, and so on. Its final instruction is instantiated with the valid letter set for that question:

Prefix-finalizer query suffix

```
Please directly answer with the best option letter from the valid
choices:  [valid letters].  Output exactly one uppercase option
letter and no other text.
```

VideoSeek keeps its native system prompt and reconstructed prefix conversation, then appends a final user request to provide the answer in this required format. AVP uses the same visible state supplied to its adapter together with the system instruction:

AVP prefix-finalizer system prompt

```
You are a prefix answerer.  Use only the supplied visible state.
Return exactly one allowed uppercase option letter and no other text.
Do not infer unseen video content.
```

The finalizer is restricted by structured decoding to the valid option set. The accepted prefix-parser forms are a bare allowed letter (including the accepted `Reply-`n suffix form) and the exact boxed wrapper used by the backend. The prediction and the complete finalizer token cost are stored together and enter the target and evaluation ledger at the checkpoint where the finalizer is used.

C Complete Main Results and Cost Accounting

We expand the main results with paired offline uncertainty intervals, online execution values, component-wise token accounting, prefix-finalization/stopping attribution, post-hoc native-cost slices, stopping locations, and the one-time cost of collecting the supervision used to fit the stopping policy.

C.1 Complete offline results

Table 6 reports all eighteen agent–model–benchmark cells together with paired video-cluster bootstrap intervals and the fraction of questions stopped before the native terminal state.

Table 6: **Complete offline CASE results.** End-to-end accuracy is Native → CASE. The 95% interval is a paired video-cluster bootstrap interval for the accuracy change (10,000 draws, seed 42).

(a) Video-MME

Agent	Model	Model save (%)	Visual save (%)	Accuracy (%)	Δ (pp)	95% CI (pp)	Stop (%)
VideoSeek	Qwen3.5	52.43	57.64	51.82 → 53.73	+1.91	[+0.22, +3.56]	42.18
VideoSeek	GLM-4.6V	49.77	58.07	56.40 → 55.69	−0.71	[-1.78, +0.31]	23.42
VideoSeek	InternVL3.5	69.85	77.06	52.84 → 52.67	−0.18	[-1.78, +1.42]	57.47
AVP	Qwen3.5	56.67	69.67	65.02 → 65.29	+0.27	[-0.98, +1.56]	56.80
AVP	GLM-4.6V	30.72	32.36	57.91 → 57.51	−0.40	[-1.51, +0.67]	55.69
AVP	InternVL3.5	62.35	65.26	51.11 → 54.27	+3.16	[+1.64, +4.71]	58.98

(b) LongVideoBench

Agent	Model	Model save (%)	Visual save (%)	Accuracy (%)	Δ (pp)	95% CI (pp)	Stop (%)
VideoSeek	Qwen3.5	64.24	64.03	40.91 → 44.13	+3.22	[+0.83, +5.56]	48.02
VideoSeek	GLM-4.6V	54.87	61.39	47.49 → 47.34	−0.15	[-1.59, +1.26]	29.54
VideoSeek	InternVL3.5	74.08	85.39	42.11 → 46.00	+3.89	[+1.44, +6.36]	66.57
AVP	Qwen3.5	55.83	71.40	59.01 → 61.71	+2.69	[+0.68, +4.70]	61.11
AVP	GLM-4.6V	35.43	42.33	51.31 → 54.38	+3.07	[+1.43, +4.68]	65.52
AVP	InternVL3.5	68.23	75.43	38.22 → 45.77	+7.55	[+4.79, +10.33]	68.44

(c) MLVU

Agent	Model	Model save (%)	Visual save (%)	Accuracy (%)	Δ (pp)	95% CI (pp)	Stop (%)
VideoSeek	Qwen3.5	59.46	69.53	45.58 → 47.88	+2.30	[+0.50, +4.07]	47.84
VideoSeek	GLM-4.6V	45.52	53.51	50.23 → 49.95	−0.28	[-1.26, +0.73]	26.03
VideoSeek	InternVL3.5	54.20	59.96	44.99 → 50.87	+5.89	[+4.27, +7.52]	63.57
AVP	Qwen3.5	65.70	81.88	55.43 → 59.89	+4.46	[+2.85, +6.05]	75.76
AVP	GLM-4.6V	31.19	38.73	53.13 → 53.54	+0.41	[-0.47, +1.30]	66.24
AVP	InternVL3.5	51.58	53.26	34.08 → 48.76	+14.67	[+12.73, +16.67]	82.98

Mean Model Token saving is 53.63% on Video-MME, 58.78% on LongVideoBench, and 51.28% on MLVU. The corresponding mean accuracy changes are $+0.67$, $+3.38$, and $+4.58$ points. The complete intervals show the variation across host agents and models around these aggregate results.

C.2 Online execution results

Figure 4 summarizes the online accuracy–efficiency pattern visually. Table 7 gives the exact Native → CASE accuracy, model-token saving, runtime saving, and speedup for every agent–model pair on the two online benchmarks. Runtime saving is $1 - T_{\mathrm{CASE}}/T_{\mathrm{Native}}$ and speedup is $T_{\mathrm{Native}}/T_{\mathrm{CASE}}$, where T is measured per-question agent-execution time within the host online loop; one-time service/process initialization is excluded, and CASE-specific probe/finalizer calls are included.

Table 7: **Online execution.** Results are reported on 900-question subsets of Video-MME and LongVideoBench. End-to-end accuracy is Native → CASE; Δ is in percentage points. Model-token/runtime savings are relative to Native, and speedup is $T_{\text{Native}}/T_{\text{CASE}}$. Bold indicates the higher accuracy in each pair.

Video-MME						
Agent	**Model**	**Model save**	**Runtime save**	**Speedup**	**Accuracy**	**Δ**
VideoSeek	Qwen3.5	52.18%	50.40%	2.02×	**60.89** → 60.00	-0.89
VideoSeek	GLM-4.6V	44.24%	39.04%	1.64×	**56.33** → 55.89	-0.44
VideoSeek	InternVL3.5	71.20%	72.27%	3.61×	52.11 → **53.22**	+1.11
AVP	Qwen3.5	57.02%	56.47%	2.30×	**64.22** → 63.33	-0.89
AVP	GLM-4.6V	30.27%	35.16%	1.54×	**57.67** → 56.44	-1.22
AVP	InternVL3.5	62.27%	61.51%	2.60×	51.11 → **53.11**	+2.00
LongVideoBench						
Agent	**Model**	**Model save**	**Runtime save**	**Speedup**	**Accuracy**	**Δ**
VideoSeek	Qwen3.5	53.49%	52.35%	2.10×	**53.78** → 52.89	-0.89
VideoSeek	GLM-4.6V	46.20%	41.68%	1.71×	49.22 → **52.22**	+3.00
VideoSeek	InternVL3.5	77.47%	80.73%	5.19×	41.78 → **44.56**	+2.78
AVP	Qwen3.5	54.93%	54.31%	2.19×	58.22 → **61.22**	+3.00
AVP	GLM-4.6V	35.75%	39.41%	1.65×	51.22 → **57.00**	+5.78
AVP	InternVL3.5	69.02%	65.83%	2.93×	41.78 → **47.22**	+5.44

C.3 Complete cost accounting

Our primary Model Token measure is the sum of text and visual tokens. For a stopped trajectory, the policy total includes the native prefix already executed, every option probe visited through the stopping checkpoint, and the finalizer actually used to answer at that prefix. Native-terminal episodes contain the full native trajectory together with the visited probes. Table 8 gives the corresponding absolute totals and component-wise savings.

Table 8: **Complete token-cost accounting.** Model Token is text plus visual tokens. Absolute Model Token totals are shown in millions as Native → CASE.

(a) Video-MME

Agent	**Model**	**Model Token (M)**	**Input save**	**Output save**	**Text save**	**Visual save**
VideoSeek	Qwen3.5	128.70 → 61.22	51.79	46.64	51.04	57.64
VideoSeek	GLM-4.6V	144.72 → 72.69	45.21	47.08	45.35	58.07
VideoSeek	InternVL3.5	111.35 → 33.58	65.19	67.24	65.34	77.06
AVP	Qwen3.5	116.67 → 50.55	50.94	55.54	51.44	69.67
AVP	GLM-4.6V	124.90 → 86.53	29.19	37.15	30.14	32.36
AVP	InternVL3.5	183.31 → 69.02	57.65	59.47	57.79	65.26

(b) LongVideoBench

Agent	**Model**	**Model Token (M)**	**Input save**	**Output save**	**Text save**	**Visual save**
VideoSeek	Qwen3.5	106.34 → 38.03	65.93	53.67	64.30	64.03
VideoSeek	GLM-4.6V	131.39 → 59.30	51.49	47.01	51.13	61.39
VideoSeek	InternVL3.5	64.32 → 16.67	64.64	72.10	65.14	85.39
AVP	Qwen3.5	73.92 → 32.65	51.22	55.42	51.63	71.40
AVP	GLM-4.6V	83.16 → 53.70	33.10	41.27	33.99	42.33
AVP	InternVL3.5	106.21 → 33.74	61.02	61.57	61.05	75.43

(c) MLVU

Agent	**Model**	**Model Token (M)**	**Input save**	**Output save**	**Text save**	**Visual save**
VideoSeek	Qwen3.5	103.73 → 42.05	55.67	51.24	54.86	69.53
VideoSeek	GLM-4.6V	143.18 → 78.01	39.84	35.11	39.44	53.51
VideoSeek	InternVL3.5	149.01 → 68.25	41.36	34.96	40.60	59.96
AVP	Qwen3.5	79.13 → 27.14	56.01	63.74	57.18	81.88
AVP	GLM-4.6V	94.57 → 65.07	24.21	34.42	25.50	38.73
AVP	InternVL3.5	106.46 → 51.55	46.63	57.66	47.85	53.26

Across the three benchmarks, reductions in visual processing accompany reductions in the growing text context and decoded model output. The absolute Model Token totals in Table 8 are the quantities from which the percentage savings in the main tables are computed.

C.4 Prefix finalization and stopping attribution

Early stopping changes two parts of native execution at once. It removes the remaining trajectory, reducing future computation but also discarding any evidence that would have been acquired later. A STOP also invokes the constrained prefix finalizer on the retained evidence; all compared stopping methods use this same finalization path. The end-to-end accuracy change therefore combines a stopping-location effect with a prefix-finalization effect.

For each method m, let H_m be a matched terminal-finalizer hybrid. On the questions that m stops, H_m keeps the complete Native trajectory but replaces the Native terminal answer with the same constrained finalizer used by m; on the remaining questions it keeps the Native terminal answer. The finalizer only maps the available evidence to one of the original answer options and does not acquire additional video evidence. Comparing H_m with Native isolates the prefix-finalization effect, while comparing m with H_m isolates the effect of stopping earlier under the same finalizer:

$$\Delta A_m = \underbrace{A(H_m) - A(\mathrm{Native})}_{\Delta A_m^{\mathrm{fin}}} + \underbrace{A(m) - A(H_m)}_{\Delta A_m^{\mathrm{stop}}}.$$

Table 9 reports this decomposition for all seven methods. For CASE, the observed $+2.88$-point end-to-end change decomposes into a $+4.80$-point prefix-finalization effect and a -1.92-point stopping-location effect, alongside 54.56% Model Token saving. The negative stopping term is consistent with the role of early stopping: removing future computation can also remove useful later evidence. The same average pattern appears across the six baselines.

Table 9: **Matched attribution of prefix finalization and stopping.** For each method m, the matched hybrid H_m runs the complete Native trajectory but applies the same constrained finalizer on exactly the questions that m would stop; other questions retain the Native terminal answer. Finalizer is $A(H_m) - A(\mathrm{Native})$ and Stopping is $A(m) - A(H_m)$, so the two terms sum to the end-to-end accuracy change. M is the formal Model Token saving. Stop-PFC and rank use Stopping and M as the two Pareto objectives. Bold marks the best value and underlining the second-best.

Method	M (%)	End-to-end Δ (pp)	Finalizer (pp)	Stopping (pp)	Stop-PFC /18 (%)	Rank ↓
Tuned checkpoint	39.09	**+3.41**	+4.13	**−0.72**	12/18 (66.7)	1.39
Tuned budget	44.97	+2.29	+3.84	−1.55	<u>13/18 (72.2)</u>	<u>1.33</u>
Confidence	31.04	+0.76	+2.30	−1.54	3/18 (16.7)	2.39
Entropy	30.73	+0.59	+2.23	−1.64	5/18 (27.8)	2.11
TASR-Video	42.49	<u>+3.29</u>	<u>+4.42</u>	<u>−1.13</u>	8/18 (44.4)	1.67
Forward-Q	<u>46.00</u>	+3.00	+4.25	−1.25	<u>13/18 (72.2)</u>	<u>1.33</u>
CASE	**54.56**	+2.88	**+4.80**	−1.92	**16/18 (88.9)**	**1.11**

For a method-independent reference, we additionally apply the same constrained finalizer to every complete Native trajectory. At each frozen paper operating point, early-stopped questions retain their prefix finalizer while non-stopped questions use this common terminal finalizer, whose cost is charged explicitly. Table 10 therefore compares all methods against one shared finalization rule. CASE remains nondominated in 16/18 configurations, with 55.24% Model Token saving and a -1.92-point accuracy difference relative to Native+F.

Table 10: **Common terminal-finalizer control.** Native+F applies the same constrained finalizer to every complete Native trajectory. Each stopping method keeps its frozen paper operating point; an early-stopped question uses its prefix finalizer, while a non-stopped question uses the same terminal finalizer as Native+F. Model-token saving is measured relative to Native+F and therefore charges the terminal-finalizer call on non-stopped questions. PFC and rank are computed only among the seven stopping methods.

Method	Acc. (%)	Δ vs. Native+F	M save (%)	PFC /18 (%)	Rank ↓
Native+F	56.20	0.00	0.00	–	–
Tuned checkpoint	**55.48**	**−0.72**	40.11	12/18 (66.7)	1.39
Tuned budget	54.65	−1.55	45.62	13/18 (72.2)	1.33
Confidence	54.67	−1.54	31.68	3/18 (16.7)	2.39
Entropy	54.56	−1.64	31.38	4/18 (22.2)	2.17
TASR-Video	55.07	−1.13	43.49	8/18 (44.4)	1.67
Forward-Q	54.96	−1.25	46.81	13/18 (72.2)	1.33
CASE	54.28	−1.92	**55.24**	**16/18 (88.9)**	**1.11**

We also form a stopping-only frontier by retaining each method's Model Token saving and replacing end-to-end accuracy change with ΔA_m^{stop}. Under this matched-finalizer diagnostic, CASE is nondominated in 16 of 18 agent–model–benchmark configurations (88.9%) with mean Pareto rank 1.11.

We apply the same paired video-cluster bootstrap used for the end-to-end frontier, keeping all operating points frozen and recomputing the matched stopping effect, saving, frontier membership, and rank in every draw. Table 11 reports the full seven-method result. Across 10,000 draws, CASE has mean stopping-only PFC 84.5% with a central 95% bootstrap interval of [72.2%, 100.0%] and mean rank 1.17 with interval [1.00, 1.33]. Its stopping-only PFC is strictly highest in 72.2% of draws and highest or tied-highest in 88.9%.

Table 11: **Stopping-only frontier stability under evaluation resampling.** We use 10,000 paired video-cluster bootstrap draws (seed 42) within each agent–model–benchmark configuration. Operating points remain frozen. Each draw recomputes the method-specific matched stopping effect, Model Token saving, frontier membership, and Pareto rank. Bold marks the best value and underlining the second-best.

Method	Point PFC	Bootstrap PFC	Central 95% interval	Point rank	Bootstrap rank (95% interval)
Tuned checkpoint	66.7	62.1	[44.4, 77.8]	1.39	1.47 [1.28, 1.72]
Tuned budget	72.2	76.1	[66.7, 83.3]	1.33	1.41 [1.22, 1.61]
Confidence	16.7	23.6	[11.1, 38.9]	2.39	2.26 [1.94, 2.56]
Entropy	27.8	25.8	[11.1, 38.9]	2.11	2.21 [1.89, 2.50]
TASR-Video	44.4	46.9	[33.3, 61.1]	1.67	1.73 [1.50, 2.00]
Forward-Q	72.2	63.4	[44.4, 77.8]	1.33	1.43 [1.22, 1.67]
CASE	**88.9**	**84.5**	[72.2, 100.0]	**1.11**	**1.17** [1.00, 1.33]

C.5 NATIVE-COST TAIL

The aggregate saving is concentrated in trajectories that are expensive under native execution. We rank questions independently within each agent–model–benchmark cell by native Model Token and report the post-hoc Top20% and Top10% slices in Table 12. These slices provide a descriptive view of where native computation and CASE savings are concentrated.

Table 12: **Post-hoc native-cost slices.** Top20 and Top10 contain the questions with the largest Native Model Token cost within each agent–model–benchmark cell.

Benchmark	Agent	Model	Top20			Top10		
			Cost share	M save	Δ	Cost share	M save	Δ
Video-MME	VideoSeek	Qwen3.5	58.49	69.92	+6.44	42.85	73.82	+11.11
Video-MME	VideoSeek	GLM-4.6V	73.81	65.55	−1.11	60.66	72.28	−2.22
Video-MME	VideoSeek	InternVL3.5	58.67	86.19	+0.67	45.57	90.32	+0.00
Video-MME	AVP	Qwen3.5	36.65	73.62	−0.44	19.76	76.00	+2.22
Video-MME	AVP	GLM-4.6V	27.87	45.07	−2.44	14.89	48.43	−1.33
Video-MME	AVP	InternVL3.5	32.98	83.96	+6.89	17.29	84.10	+4.00
LongVideoBench	VideoSeek	Qwen3.5	61.70	83.04	+8.58	47.36	87.49	+8.21
LongVideoBench	VideoSeek	GLM-4.6V	74.52	68.06	+1.87	50.42	74.08	+1.49
LongVideoBench	VideoSeek	InternVL3.5	50.09	90.34	+10.82	35.92	94.19	+5.22
LongVideoBench	AVP	Qwen3.5	33.36	72.84	+4.85	17.48	73.55	+4.48
LongVideoBench	AVP	GLM-4.6V	26.21	47.11	+0.00	13.73	47.32	+0.75
LongVideoBench	AVP	InternVL3.5	34.01	85.01	+17.91	17.84	85.40	+24.63
MLVU	VideoSeek	Qwen3.5	59.95	75.18	+4.37	42.07	80.83	+8.26
MLVU	VideoSeek	GLM-4.6V	69.62	62.03	+1.15	52.51	72.44	−2.75
MLVU	VideoSeek	InternVL3.5	46.83	69.67	+7.82	31.62	75.17	+7.34
MLVU	AVP	Qwen3.5	32.32	77.75	+2.99	17.40	79.26	+4.59
MLVU	AVP	GLM-4.6V	27.42	37.48	+0.00	14.41	39.74	+0.92
MLVU	AVP	InternVL3.5	42.95	69.88	+20.46	24.00	72.52	+21.10

The Top20% slices account for 48.08%, 46.65%, and 46.51% of native Model Token on average for Video-MME, LongVideoBench, and MLVU. CASE removes 70.72%, 74.40%, and 65.33% of the corresponding model-token cost. The Top10% slices show the same concentration more strongly.

C.6 Where stopping occurs

Figure 5 shows the first valid stopping checkpoint for every question. Checkpoint 0 is the first eligible nonterminal state; the native-terminal segment also includes Short videos, for which the main protocol leaves the host agent unchanged.

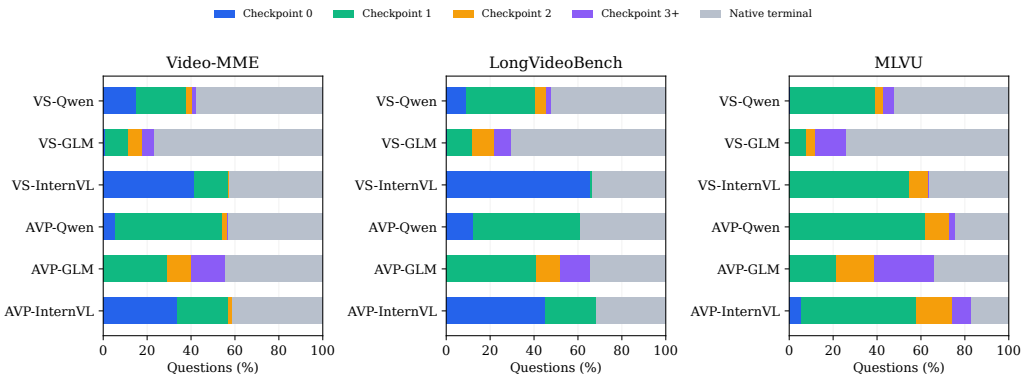

Figure 5: **Where CASE terminates.** Each bar partitions questions by the checkpoint of the first valid STOP decision or the Native-terminal outcome.

The mean stop rates are 49.09%, 56.53%, and 60.40% on Video-MME, LongVideoBench, and MLVU. In most Video-MME and LongVideoBench cells, more than half of the stopped questions terminate at checkpoint 0 or 1. Table 13 places these early stops in the context of the native trajectory length.

Table 13: **Native trajectory length and early-stop suffix.** Native checkpoints are counted on Medium+Long questions where CASE is active. Early denotes stops at checkpoint 0 or 1.

Benchmark	Agent	Model	Native ckpts. mean / med.	Early among stops	Bypassed after early stop
Video-MME	VideoSeek	Qwen3.5	2.92 / 2	89.9	3.28
Video-MME	VideoSeek	GLM-4.6V	3.78 / 2	49.0	5.32
Video-MME	VideoSeek	InternVL3.5	3.14 / 2	99.0	3.19
Video-MME	AVP	Qwen3.5	4.10 / 5	95.4	3.74
Video-MME	AVP	GLM-4.6V	4.79 / 5	52.4	3.89
Video-MME	AVP	InternVL3.5	4.09 / 5	96.4	4.07
LongVideoBench	VideoSeek	Qwen3.5	3.92 / 3	84.3	4.21
LongVideoBench	VideoSeek	GLM-4.6V	4.95 / 2	40.3	6.61
LongVideoBench	VideoSeek	InternVL3.5	2.69 / 2	100.0	2.93
LongVideoBench	AVP	Qwen3.5	3.87 / 5	99.9	3.63
LongVideoBench	AVP	GLM-4.6V	4.93 / 5	62.1	3.96
LongVideoBench	AVP	InternVL3.5	4.08 / 5	99.6	3.94
MLVU	VideoSeek	Qwen3.5	2.90 / 2	81.7	3.76
MLVU	VideoSeek	GLM-4.6V	4.16 / 2	30.0	4.96
MLVU	VideoSeek	InternVL3.5	2.65 / 2	86.4	1.98
MLVU	AVP	Qwen3.5	3.72 / 5	82.1	3.51
MLVU	AVP	GLM-4.6V	4.42 / 5	32.2	3.72
MLVU	AVP	InternVL3.5	4.23 / 5	70.1	3.83

Across the current cells, checkpoint-0/1 stops bypass several later native decision checkpoints on average. The effect is largest on the longer VideoSeek trajectories, where early stops remove a substantial native suffix.

C.7 Supervision-acquisition cost

Learning the stopping rule requires complete native training trajectories together with option probes and reachable-prefix finalizers on the 50-video training role. Table 14 audits this one-time acquisition cost separately from deployment.

Table 14: **One-time supervision-acquisition cost on the 50-video training role.** Native trajectory, probe, and reachable-prefix finalizer costs are reported in millions of Model Tokens. Parentheses give the number of auxiliary model calls. Ridge fitting itself uses the cached features/targets and requires no VLM calls.

Agent	Model	Native traj. (M)	Probes (M; calls)	Finalizers (M; calls)	Total (M)
VideoSeek	Qwen3.5	5.50	0.57; 288	1.11; 288	7.19
VideoSeek	GLM-4.6V	5.11	0.45; 304	1.16; 304	6.71
VideoSeek	InternVL3.5	2.67	0.09; 132	0.26; 132	3.02
AVP	Qwen3.5	4.68	0.48; 406	0.53; 406	5.69
AVP	GLM-4.6V	6.82	0.71; 630	0.78; 630	8.32
AVP	InternVL3.5	7.95	0.39; 405	0.44; 405	8.78

Across the six agent–model cells, collecting the 150 training-question supervision requires 3.02–8.78 million Model Tokens in total. The number of cached probe/finalizer pairs ranges from 132 to 630, reflecting the different native trajectory lengths exposed by the two hosts and three VLMs. The cached supervision supports feature construction, target construction, and Ridge fitting, and the uniform cost ledger provides the call and token accounting in Table 14.

D Baseline Comparison and Operating-Point Calibration

This section gives the complete baseline results and the calibration analysis behind the operating-point rule used throughout the paper.

D.1 STOPPING BASELINES

The six controls span position-only, cost-only, confidence-based, stability-based, and learned stopping rules. **Tuned checkpoint** stops at a globally calibrated checkpoint index k. **Tuned budget** stops at a calibrated cumulative Model Token budget B. **Top-option confidence** and **option entropy** threshold the calibrated option distribution Schuster et al. (2022). **TASR-Video** adapts TASR's answer-stability gate to our multiple-choice setting by replacing TASR's calibrated answer-token logit margin with the external option probe's top-two probability margin; its threshold is selected by the shared calibration rule Kieback et al. (2026). **Forward-Q** implements the Monte Carlo continuation target from Stop-RAG Park et al. (2025). Two weighted Ridge predictors operate on the checkpoint state, estimating current-prefix correctness and the maximum correctness among later prefixes and the Native terminal answer; their difference is thresholded. With the shared features, trajectory weights, standardization, and Ridge regularization used here, this score is the sign-reversed quality-only score in Appendix E.2, so the two controls use threshold sets paired by sign. This is a lightweight video adaptation of the Monte Carlo target, whereas Stop-RAG's main method trains a text encoder with annealed forward-view $Q(\lambda)$ targets. Every tunable method uses the same Video-MME calibration rule described below, and the selected point is frozen for zero-shot evaluation on LongVideoBench and MLVU. Tuned checkpoint and tuned budget do not invoke the option probe; all other auxiliary calls are charged only when used.

D.2 COMPLETE BASELINE RESULTS

Tables 15 and 16 expand the main baseline table with visual-token savings.

Table 15: **Complete stopping-baseline results for VideoSeek.** Entries report accuracy change (pp), Model Token saving (M), and visual-token saving (V). LongVideoBench and MLVU use the frozen Video-MME operating point.

Model	Method	Video-MME			LongVideoBench			MLVU		
		M	V	Δ	M	V	Δ	M	V	Δ
Qwen3.5	Tuned checkpoint	42.57	42.17	+2.00	57.15	52.11	+3.89	59.14	62.99	+2.39
Qwen3.5	Tuned budget	36.73	34.58	**+2.36**	51.99	45.10	**+4.19**	42.34	48.70	+2.16
Qwen3.5	Confidence	36.13	42.66	+1.69	37.57	39.57	+2.69	48.31	57.58	+2.21
Qwen3.5	Entropy	33.72	40.13	+1.51	42.52	44.58	+2.09	43.84	53.71	+2.58
Qwen3.5	TASR-Video	37.65	46.10	+1.56	52.77	54.10	+3.29	52.79	65.26	**+3.36**
Qwen3.5	Forward-Q	38.64	43.25	+0.93	50.47	49.59	+3.74	32.91	43.63	+2.44
Qwen3.5	CASE	**52.43**	**57.64**	+1.91	**64.24**	**64.03**	+3.22	**59.46**	**69.53**	+2.30
GLM-4.6V	Tuned checkpoint	30.51	29.92	+0.22	32.82	31.95	+0.15	31.77	30.95	**+0.55**
GLM-4.6V	Tuned budget	32.36	33.06	**+0.36**	35.14	34.54	**+0.45**	31.48	36.97	+0.41
GLM-4.6V	Confidence	34.64	43.10	−2.09	32.51	36.21	−2.09	**56.83**	**62.83**	−5.66
GLM-4.6V	Entropy	32.53	42.61	−1.82	36.92	40.14	−2.99	51.06	56.90	−5.24
GLM-4.6V	TASR-Video	26.86	42.77	−0.36	37.75	50.59	−0.75	44.08	58.41	−0.18
GLM-4.6V	Forward-Q	43.49	50.08	−0.13	50.19	56.58	+0.22	32.00	39.67	+0.46
GLM-4.6V	CASE	**49.77**	**58.07**	−0.71	**54.87**	**61.39**	−0.15	45.52	53.51	−0.28
InternVL3.5	Tuned checkpoint	44.72	41.88	+0.80	36.23	34.31	+3.59	38.84	40.72	+3.08
InternVL3.5	Tuned budget	28.01	20.86	+0.13	18.70	14.37	+0.22	21.74	21.94	+1.10
InternVL3.5	Confidence	33.90	38.85	+0.18	10.71	21.03	+2.47	35.87	40.06	+2.71
InternVL3.5	Entropy	30.91	35.50	+0.22	12.13	21.64	+2.69	30.29	34.14	+2.02
InternVL3.5	TASR-Video	48.30	55.65	**+1.60**	38.36	50.93	**+4.71**	52.17	**60.69**	+5.47
InternVL3.5	Forward-Q	54.53	59.96	+1.51	46.82	54.88	+4.04	35.38	40.29	+3.40
InternVL3.5	CASE	**69.85**	**77.06**	−0.18	**74.08**	**85.39**	+3.89	**54.20**	59.96	**+5.89**

Table 16: **Complete stopping-baseline results for AVP.** Entries report accuracy change (pp), Model Token saving (M), and visual-token saving (V). LongVideoBench and MLVU use the frozen Video-MME operating point.

Model	Method	Video-MME			LongVideoBench			MLVU		
		M	V	Δ	M	V	Δ	M	V	Δ
Qwen3.5	Tuned checkpoint	25.27	32.53	**+1.96**	23.75	32.47	**+4.41**	31.23	40.49	**+5.75**
Qwen3.5	Tuned budget	**64.11**	**73.24**	−0.49	**72.06**	**80.59**	−2.54	**71.89**	**87.01**	+4.78
Qwen3.5	Confidence	28.04	38.97	−0.76	9.49	24.66	+0.07	28.99	39.89	−0.64
Qwen3.5	Entropy	29.59	40.23	−1.11	13.30	28.26	+0.00	29.57	40.45	−0.51
Qwen3.5	TASR-Video	45.72	61.45	+0.76	39.85	60.66	+3.66	57.53	74.50	+5.47
Qwen3.5	Forward-Q	54.11	67.36	+0.49	53.15	68.72	+2.99	61.00	77.04	+4.65
Qwen3.5	CASE	56.67	69.67	+0.27	55.83	71.40	+2.69	65.70	81.88	+4.46
GLM-4.6V	Tuned checkpoint	23.01	26.06	−0.13	24.13	27.77	**+3.96**	33.01	38.07	+1.70
GLM-4.6V	Tuned budget	30.00	29.02	**+0.09**	**39.05**	40.06	+3.66	26.57	32.41	**+1.89**
GLM-4.6V	Confidence	27.24	32.71	−0.71	21.04	29.52	+1.65	44.83	52.11	+0.28
GLM-4.6V	Entropy	27.79	33.22	−0.80	25.15	32.98	+1.65	**45.72**	53.13	−0.23
GLM-4.6V	TASR-Video	24.41	**34.00**	−0.40	20.43	34.52	+1.80	43.48	**53.89**	+1.15
GLM-4.6V	Forward-Q	**30.76**	32.60	−0.44	35.36	42.33	+3.07	32.92	40.56	+0.55
GLM-4.6V	CASE	30.72	32.36	−0.40	35.43	**42.33**	+3.07	31.19	38.73	+0.41
InternVL3.5	Tuned checkpoint	54.30	56.13	**+4.04**	58.66	62.90	+8.90	56.56	56.75	+14.03
InternVL3.5	Tuned budget	**67.70**	**69.63**	+1.69	**78.13**	**84.14**	+6.58	**61.49**	**62.76**	+14.12
InternVL3.5	Confidence	28.35	32.40	+0.58	14.36	22.33	+3.89	29.88	33.64	+7.27
InternVL3.5	Entropy	25.68	29.70	+0.98	16.34	24.45	+3.89	26.08	29.73	+5.75
InternVL3.5	TASR-Video	46.07	51.06	+3.87	48.28	58.47	**+9.65**	48.30	50.36	+14.54
InternVL3.5	Forward-Q	59.85	62.91	+3.73	64.44	71.82	+7.93	51.90	53.81	+14.49
InternVL3.5	CASE	62.35	65.26	+3.16	68.23	75.43	+7.55	51.58	53.26	**+14.67**

The baseline families occupy different parts of the accuracy–cost plane. Fixed position and budget controls are competitive in several cells, while confidence and entropy thresholds vary more strongly with the host and model.

D.3 Common-budget comparison

Table 17 compares the seven stopping methods after selecting operating points toward the same pre-specified calibration saving targets and then evaluating them under the common terminal-finalizer protocol from Appendix C.4. This controls both stopping aggressiveness at selection time and the finalization rule at evaluation time; test labels do not affect operating-point selection.

Table 17: **Common-budget comparison under a common terminal finalizer.** Each cell is test ΔAcc (pp) / realized Model Token saving (%), both relative to Native+F. For each requested saving target, method parameters are selected using Video-MME calibration only; evaluation labels never affect selection. CASE, Forward-Q, tuned budget, confidence, and entropy match all six agent–model calibration targets within the preset tolerance. Tuned checkpoint matches 5/6, 1/6, 5/6, and 2/6 cells at 30/40/50/60%, and TASR-Video matches 6/6, 6/6, 5/6, and 4/6; unmatched entries use the nearest calibration point without extrapolation. Values macro-average the 18 agent–model–benchmark configurations.

Method	30% target	40% target	50% target	60% target
Tuned checkpoint	−0.36/27.55	−0.59/31.39	−0.84/42.93	−1.40/49.16
Tuned budget	−0.26/25.07	−0.47/32.27	−0.85/42.68	−1.33/52.09
Confidence	−1.17/23.76	−1.58/31.75	−2.26/40.80	−3.05/50.73
Entropy	−1.41/26.02	−1.80/32.86	−2.44/43.72	−3.38/53.26
TASR-Video	−0.72/26.15	−0.99/34.05	−1.20/42.08	−1.39/49.39
Forward-Q	−0.48/22.39	−0.83/30.76	−1.08/40.78	−1.35/48.61
CASE	−0.39/21.43	−0.73/29.95	−1.05/40.38	−1.40/47.97

At comparable calibration budgets, CASE, Forward-Q, and TASR-Video have very similar stopping accuracy, while tuned budget is modestly higher at some intermediate targets. The result complements

the independently selected operating points: the heterogeneous-cost correction primarily shifts CASE toward a more aggressive saving regime rather than creating a large fixed-budget accuracy advantage.

D.4 Pareto stability under evaluation resampling

PFC summarizes frontier membership over finite evaluation sets, so we additionally characterize how the frontier changes under video-level resampling. Within each of the 18 agent–model–benchmark configurations, we resample videos with replacement and use the same cluster draw for every method, then recompute accuracy change, Model Token saving, Pareto membership, and Pareto rank. The calibrated operating points remain frozen throughout this analysis.

Table 18: **Pareto-frontier stability under evaluation resampling.** We use 10,000 paired video-cluster bootstrap draws (seed 42) within each agent–model–benchmark configuration. Operating points remain frozen; each draw recomputes accuracy change, Model Token saving, frontier membership, and Pareto rank. PFC follows Eq. 10. Bold marks the best bootstrap mean and underlining the second-best.

Method	Point PFC	Bootstrap PFC	Central 95% interval	Point rank	Bootstrap rank (95% interval)
Tuned checkpoint	61.1	49.8	[33.3, 66.7]	1.44	<u>1.67</u> [1.44, 1.94]
Tuned budget	72.2	<u>63.8</u>	[50.0, 77.8]	1.72	1.85 [1.61, 2.11]
Confidence	11.1	12.0	[5.6, 16.7]	2.72	2.93 [2.61, 3.22]
Entropy	11.1	10.5	[5.6, 16.7]	2.67	2.92 [2.61, 3.22]
TASR-Video	50.0	46.8	[33.3, 61.1]	1.67	1.82 [1.56, 2.06]
Forward-Q	61.1	51.1	[33.3, 66.7]	1.56	1.72 [1.50, 2.00]
CASE	83.3	**78.2**	[66.7, 88.9]	1.22	**1.28** [1.17, 1.44]

Across 10,000 paired bootstrap draws, CASE has mean PFC 78.2% with a central 95% bootstrap interval of [66.7%, 88.9%] and mean Pareto rank 1.28 with interval [1.17, 1.44]. Its PFC is strictly the highest among the compared methods in 90.0% of draws and is highest or tied-highest in 97.5%. Thus the resampled frontier shows the same broad pattern as the point-estimate analysis while making the evaluation-sample variation explicit.

D.5 Unified calibration rule

For a method's Video-MME calibration candidates, let s_c, Δ_c, and ℓ_c denote Model Token saving, accuracy change, and the one-sided video-cluster accuracy lower bound. We first form

$$\mathcal{C}_{\text{strict}} = \{c : s_c \geq 0.55,\ \ell_c \geq -0.01\}.$$

If this set is nonempty, let $\Delta^* = \max_{c \in \mathcal{C}_{\text{strict}}} \Delta_c$ and retain candidates satisfying $\Delta_c \geq \Delta^* - 0.02$. Among them we select the point with the greatest saving, breaking ties by accuracy, the lower-bound statistic, and candidate index. If the strict set is empty, we select the candidate whose calibration saving is closest to 40%, with ties broken by the lower-bound statistic, accuracy, and candidate index.

D.6 Calibration sensitivity

Table 19 compares nearby strict saving floors. Raising the floor from 50% to 55% leaves CASE feasible in the same five of six agent–model cells, whereas a 60% floor reduces that count to four.

Table 19: **Strict-set feasibility on Video-MME calibration.** Entries give the number of agent–model cells with at least one candidate satisfying the saving floor and the -1 pp accuracy-LCB constraint.

Method	50%	55%	60%
Tuned checkpoint	<u>4/6</u>	2/6	1/6
Tuned budget	3/6	2/6	2/6
Confidence	0/6	0/6	0/6
Entropy	0/6	0/6	0/6
TASR-Video	3/6	3/6	1/6
Forward-Q	<u>4/6</u>	<u>4/6</u>	<u>3/6</u>
CASE	**5/6**	**5/6**	**4/6**

Figure 6 varies the other two calibration choices. For CASE, widening the near-optimality band from 0 to 2 pp raises mean calibration Model Token saving from 55.98% to 65.77%; further widening to 2.5 or 3 pp changes the mean saving only to 65.97%. For methods that enter the fallback branch, calibration accuracy is similar across the 35–45% target range, so we use its round-number midpoint, 40%.

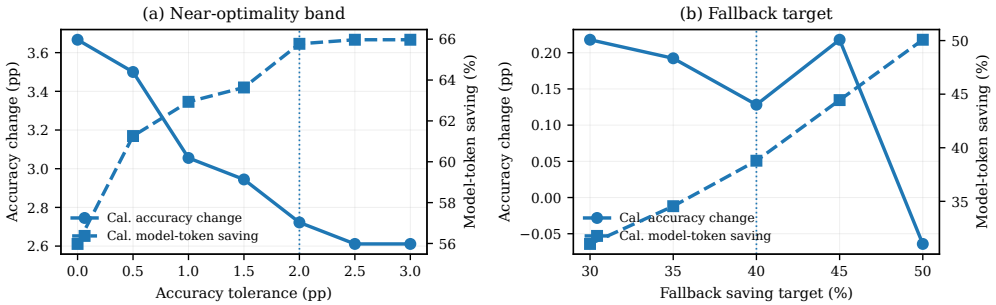

Figure 6: **Calibration sensitivity of the operating-point rule.** (a) CASE mean calibration accuracy change and model-token saving as the near-optimality band is widened, with the 55% saving floor and -1 pp LCB constraint fixed. (b) Mean calibration behavior over fallback method–cell pairs as the fallback target changes, with the 2-pp band fixed. Dotted lines mark the paper settings.

D.7 Branch usage and frozen operating points

Table 20 gives every frozen parameter together with its calibration statistics; S/F superscripts identify the strict and 40% fallback branches. CASE uses the strict branch in five of six cells, with AVP–GLM using the fallback.

D.8 CASE calibration candidates

For each agent–model pair, CASE evaluates five cost weights and 81 boundaries, yielding 405 calibration candidates. Figure 7 shows the strict-feasible and near-optimal regions together with the selected point.

E Extended Ablations

We extend the main component analysis along cost modeling, target horizon, state representation, fitting procedure, training-set size, and probe design. The main-paper ablation treats every variant as a complete stopping method: each model selects its own Video-MME operating point with the shared calibration rule before zero-shot transfer. In this appendix, saving-matched diagnostics serve a different purpose: they hold stopping aggressiveness approximately fixed to isolate accuracy differences at similar computation savings. Unless stated otherwise, all alternatives use the same training/calibration roles and the complete deployment cost ledger.

E.1 Independent end-to-end ablations

Table 21 expands the five independently selected variants from the main paper with visual-token saving, stop rate, and strict-calibration frequency.

Execution-state-only excludes the answer-support block and therefore does not invoke the option probe; its continuation-cost target uses the corresponding no-probe execution path. It reaches $+2.21$ points with 55.30% Model Token saving overall, compared with $+2.88$/54.56% for Full.

E.2 Quality and continuation-cost correction

Equation 6 separates the quality advantage from the correction induced by heterogeneous continuation cost without changing the Full predictor. Across all six agent–model calibration sets and all five

Table 20: **Frozen Video-MME calibration operating points.** Superscripts S/F mark the strict and fallback branches. Calibration columns report accuracy change, one-sided accuracy LCB, and Model Token saving.

(a) VideoSeek

Model	Method	Parameter	Cal. M	Cal. Δ	LCB
Qwen3.5	Tuned checkpoint	$k = 2^F$	46.15	+0.00	−3.31
Qwen3.5	Tuned budget	$B = 34.2\text{k}^F$	38.06	+0.33	−2.83
Qwen3.5	Confidence	$\tau = +0.525^F$	39.32	+2.33	−1.42
Qwen3.5	Entropy	$\tau = +0.8^F$	34.50	+1.33	−2.23
Qwen3.5	TASR-Video	$m = 0.05^F$	40.31	−0.33	−4.19
Qwen3.5	Forward-Q	$\tau = -0.135^F$	39.86	+2.33	−1.09
Qwen3.5	CASE	$\lambda = 0.2,\ b = +0.195^S$	**58.37**	**+3.67**	**−0.87**
GLM-4.6V	Tuned checkpoint	$k = 9^F$	35.89	−1.00	−2.82
GLM-4.6V	Tuned budget	$B = 118.0\text{k}^F$	41.11	−0.67	−2.22
GLM-4.6V	Confidence	$\tau = +0.425^F$	39.82	+0.00	−3.12
GLM-4.6V	Entropy	$\tau = +0.9^F$	31.01	+0.67	−2.54
GLM-4.6V	TASR-Video	$m = 0.1^F$	38.60	−2.67	−5.65
GLM-4.6V	Forward-Q	$\tau = -0.035^S$	58.19	+2.00	**−0.68**
GLM-4.6V	CASE	$\lambda = 0.2,\ b = -0.010^S$	**63.06**	**+2.33**	−0.90
InternVL3.5	Tuned checkpoint	$k = 2^S$	57.66	+2.33	−0.39
InternVL3.5	Tuned budget	$B = 67.3\text{k}^F$	41.26	+1.33	−0.40
InternVL3.5	Confidence	$\tau = +0.65^F$	40.84	+1.67	−1.39
InternVL3.5	Entropy	$\tau = +0.65^F$	40.95	+1.00	−1.96
InternVL3.5	TASR-Video	$m = 0^S$	60.13	**+5.00**	**+1.23**
InternVL3.5	Forward-Q	$\tau = -0.14^S$	63.89	+3.33	−0.60
InternVL3.5	CASE	$\lambda = 0.8,\ b = -0.175^S$	**82.84**	+3.67	−0.87

(b) AVP

Model	Method	Parameter	Cal. M	Cal. Δ	LCB
Qwen3.5	Tuned checkpoint	$k = 3^F$	31.79	**+4.33**	**+1.45**
Qwen3.5	Tuned budget	$B = 4.9\text{k}^S$	**78.22**	+3.00	−0.90
Qwen3.5	Confidence	$\tau = +0.675^F$	38.97	−3.67	−6.24
Qwen3.5	Entropy	$\tau = +0.7^F$	40.23	−3.67	−6.24
Qwen3.5	TASR-Video	$m = 0^S$	57.33	+4.00	+0.67
Qwen3.5	Forward-Q	$\tau = -0.195^S$	67.39	+2.67	−0.61
Qwen3.5	CASE	$\lambda = 0.4,\ b = +0.070^S$	69.48	+3.00	−0.58
GLM-4.6V	Tuned checkpoint	$k = 3^F$	31.68	−2.33	−5.75
GLM-4.6V	Tuned budget	$B = 28.9\text{k}^F$	39.78	−2.00	−5.47
GLM-4.6V	Confidence	$\tau = +0.55^F$	40.44	+0.33	−2.19
GLM-4.6V	Entropy	$\tau = +0.8^F$	**41.35**	**+1.33**	**−1.36**
GLM-4.6V	TASR-Video	$m = 0.2^F$	39.62	+0.33	−2.42
GLM-4.6V	Forward-Q	$\tau = -0.09^F$	40.29	+0.33	−2.31
GLM-4.6V	CASE	$\lambda = 0.05,\ b = +0.070^F$	39.95	+0.00	−2.58
InternVL3.5	Tuned checkpoint	$k = 1^S$	70.02	**+4.00**	+0.24
InternVL3.5	Tuned budget	$B = 3.3\text{k}^S$	**85.75**	+3.67	−0.30
InternVL3.5	Confidence	$\tau = +0.55^F$	40.65	+1.00	−1.13
InternVL3.5	Entropy	$\tau = +0.8^F$	36.14	+1.00	−0.82
InternVL3.5	TASR-Video	$m = 0^S$	59.16	**+4.00**	**+0.49**
InternVL3.5	Forward-Q	$\tau = -0.19^S$	76.90	+3.33	−0.44
InternVL3.5	CASE	$\lambda = 0.8,\ b = -0.190^S$	80.94	+3.67	−0.30

cost weights, the factorized score in Equation 8 agrees with direct Gap-Ridge fitting to within floating-point precision (maximum absolute difference 5.6×10^{-15}).

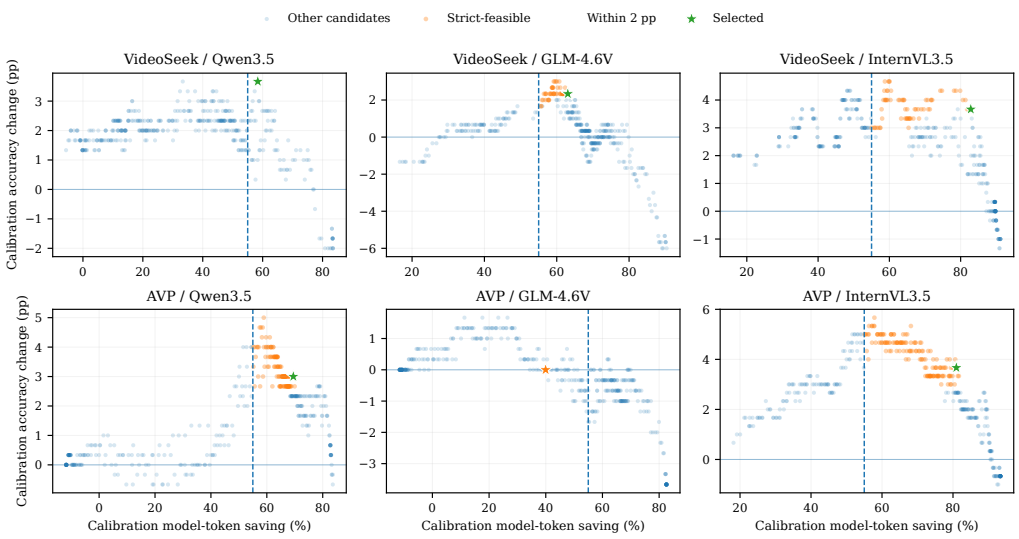

Figure 7: **CASE calibration candidate clouds.** Color separates strict-feasible candidates ($\geq 55\%$ saving and LCB ≥ -1 pp) from the remaining grid; open markers show strict candidates within 2 pp of the best strict calibration accuracy, and stars mark the selected operating points. AVP–GLM uses the 40% fallback because its strict set is empty.

Table 21: **Complete end-to-end ablation metrics.** These are the same five independently selected variants as Table 4. Δ is accuracy change in percentage points; M and V are model- and visual-token savings; Stop is the early-stop rate. Strict counts agent–model cells selected through the primary calibration branch. Bold marks the best value and underlining the second-best.

(a) Video-MME

Variant	M	V	Δ	Stop
Full	53.63	**60.01**	+0.67	**49.09**
Belief-only	36.90	44.20	+0.38	36.80
Exec-state-only	**55.21**	58.05	+0.09	47.10
Current-corr.	35.99	42.21	−0.19	35.51
w/o cost correction	46.90	52.69	**+1.01**	44.35

(b) LongVideoBench

Variant	M	V	Δ	Stop
Full	58.78	66.66	+3.38	**56.53**
Belief-only	33.46	41.55	+2.49	36.37
Exec-state-only	**64.48**	**68.15**	+2.26	56.20
Current-corr.	31.55	39.19	+2.02	35.13
w/o cost correction	50.07	57.32	**+3.66**	49.98

(c) MLVU

Variant	M	V	Δ	Stop
Full	**51.28**	**59.48**	**+4.58**	**60.40**
Belief-only	49.28	57.19	+2.70	54.88
Exec-state-only	46.20	51.36	+4.27	53.83
Current-corr.	40.57	47.96	+1.34	46.93
w/o cost correction	41.02	49.17	+4.33	53.79

(d) Overall summary

Variant	M	V	Δ	Stop	Strict /6
Full	54.56	**62.05**	+2.88	**55.34**	**5/6**
Belief-only	39.88	47.65	+1.86	42.68	2/6
Exec-state-only	**55.30**	59.19	+2.21	52.37	4/6
Current-corr.	36.04	43.12	+1.06	39.19	1/6
w/o cost correction	46.00	53.06	**+3.00**	49.37	4/6

Table 22: **Quality and continuation-cost components under independent selection.** Each variant selects its own Video-MME operating point with the shared rule and transfers the frozen policy to LongVideoBench and MLVU. Entries are Model Token saving (%) / accuracy change (pp). The factorized Full predictor agrees with the original direct Gap-Ridge to a maximum absolute calibration-score difference of 5.6×10^{-15}.

Variant	Video-MME	LongVideoBench	MLVU	Overall
Quality + heterogeneous-cost correction	**53.63** / +0.67	**58.78** / +3.38	**51.28 / +4.58**	**54.56** / +2.88
Quality only (w/o cost correction)	46.90 / **+1.01**	50.07 / **+3.66**	41.02 / +4.33	46.00 / **+3.00**
Quality + constant-step cost	49.17 / +0.84	52.35 / +3.40	45.57 / +4.20	49.03 / +2.82

The cost component changes the selected efficiency regime rather than increasing accuracy. Quality-only selection gives $+3.00$ points at 46.00% Model Token saving, whereas Full gives $+2.88$ at 54.56%: an accuracy difference of about -0.13 points accompanies an 8.57-point increase in saving. Under the paired thresholds above, the linear Forward-Q control selects the same policies and obtains the same aggregate result as Quality-only. Replacing heterogeneous native continuation cost with a constant per-step cost reaches $+2.82$/49.03%, recovering part but not all of the efficiency shift. Thus the heterogeneous cost correction aligns the quality predictor with a more aggressive computation-saving operating point while leaving answer quality nearly unchanged.

E.3 SAVING-MATCHED CORE DIAGNOSTICS

Table 23 controls stopping aggressiveness by evaluating each core alternative at the calibration point whose Model Token saving is closest to the corresponding Full policy. This saving-matched view isolates answer-quality differences at comparable early-stopping intensity; deployed operating points are the independently selected variants in Table 21.

Table 23: **Core saving-matched diagnostics.** Alternatives are selected at the calibration Model Token saving closest to the corresponding Full policy, with no additional saving floor. This controls stopping aggressiveness and isolates accuracy differences at similar saving. Entries report accuracy change (pp), Model Token saving (M), and stop rate (S).

(a) VideoSeek

Model	Variant	Video-MME			LongVideoBench			MLVU		
		M	Δ	S	M	Δ	S	M	Δ	S
Qwen3.5	Full	**52.43**	+1.91	42.18	64.24	**+3.22**	48.02	59.46	+2.30	47.84
Qwen3.5	Belief-only	50.16	**+2.00**	42.09	64.06	+1.87	**49.96**	68.95	+2.12	**55.29**
Qwen3.5	Execution-state-only	51.83	+0.98	37.82	64.72	+2.62	46.00	68.23	+1.10	52.44
Qwen3.5	Current-correctness	50.81	+1.64	**42.84**	60.74	+2.09	46.45	51.02	**+2.94**	45.26
Qwen3.5	One-step target	51.31	+0.93	40.49	**65.43**	**+3.22**	49.29	**68.98**	+1.70	54.83
GLM-4.6V	Full	49.77	−0.71	23.42	54.87	−0.15	29.54	45.52	−0.28	26.03
GLM-4.6V	Belief-only	51.94	−2.09	31.47	52.23	−1.72	**33.21**	71.01	−5.84	58.92
GLM-4.6V	Execution-state-only	48.79	−0.67	17.60	**58.98**	**+0.37**	26.03	27.01	**+0.28**	9.29
GLM-4.6V	Current-correctness	**52.38**	−2.89	**34.67**	41.89	−1.42	29.17	**76.67**	−9.66	**69.00**
GLM-4.6V	One-step target	52.20	**−0.53**	24.76	55.71	−0.15	29.77	60.57	−1.61	39.10
InternVL3.5	Full	69.85	−0.18	57.47	74.08	+3.89	66.57	54.20	+5.89	63.57
InternVL3.5	Belief-only	70.08	−0.89	58.13	64.45	**+4.26**	58.86	86.01	+3.86	90.89
InternVL3.5	Execution-state-only	70.97	**+0.31**	57.11	**76.79**	+4.11	**66.94**	65.23	+5.43	75.34
InternVL3.5	Current-correctness	**74.98**	−1.73	**62.62**	74.41	+4.04	**66.94**	**87.00**	+3.27	**91.49**
InternVL3.5	One-step target	69.56	−0.36	57.38	73.51	+4.11	66.27	58.86	**+6.16**	68.49

(b) AVP

Model	Variant	Video-MME			LongVideoBench			MLVU		
		M	Δ	S	M	Δ	S	M	Δ	S
Qwen3.5	Full	**56.67**	+0.27	56.80	55.83	+2.69	**61.11**	65.70	+4.46	75.76
Qwen3.5	Belief-only	54.74	−2.40	55.02	50.90	+1.65	58.56	58.79	+1.29	68.40
Qwen3.5	Execution-state-only	56.22	**+1.02**	55.20	**57.04**	**+3.29**	57.52	52.71	**+5.01**	71.07
Qwen3.5	Current-correctness	55.97	−3.47	**57.11**	50.21	−0.30	57.37	**71.67**	+0.23	**83.07**
Qwen3.5	One-step target	56.49	+0.36	56.62	55.39	+2.77	60.28	64.20	+4.74	75.57
GLM-4.6V	Full	30.72	−0.40	**55.69**	35.43	+3.07	65.52	31.19	+0.41	**66.24**
GLM-4.6V	Belief-only	27.00	−0.62	42.62	17.92	+0.97	36.13	**45.53**	−0.46	63.57
GLM-4.6V	Execution-state-only	**31.97**	**+0.13**	53.02	**52.16**	+3.07	**71.95**	7.42	**+0.78**	29.48
GLM-4.6V	Current-correctness	26.77	−0.53	40.76	21.38	+0.97	39.64	38.16	−0.05	56.58
GLM-4.6V	One-step target	30.65	−0.22	54.76	33.86	**+3.14**	61.78	13.63	+0.60	49.68
InternVL3.5	Full	62.35	**+3.16**	58.98	68.23	**+7.55**	68.44	51.58	+14.67	82.98
InternVL3.5	Belief-only	63.32	+0.04	**62.09**	67.58	+7.18	68.51	**78.29**	+10.49	**92.50**
InternVL3.5	Execution-state-only	63.57	+2.84	57.91	72.42	**+7.55**	68.14	52.11	**+14.95**	82.70
InternVL3.5	Current-correctness	**65.39**	+0.00	61.16	**76.21**	+6.43	**72.85**	70.85	+10.44	81.42
InternVL3.5	One-step target	63.03	**+3.16**	59.16	70.98	+7.18	69.93	52.64	+14.44	82.61

Belief-only and Current-correctness remain lower than Full across the benchmark averages at comparable saving. Execution-state-only and the One-step horizon control remain closer, showing that these restricted variants preserve much of the local stopping signal.

E.4 Continuation horizon

Table 24 varies how far the realized suffix is exposed to the target while retaining the full state and Ridge learner. These controls are saving-matched to Full because the question here is the effect of supervision horizon at comparable stopping aggressiveness.

Table 24: **Continuation-horizon diagnostics.** Alternatives are saving-matched to Full when feasible so that the comparison isolates the supervision horizon at similar stopping aggressiveness. Means are over the feasible agent–model cells; evaluation always uses the real complete cost ledger.

Target	Feasible	VMME M	VMME Δ	LVB M	LVB Δ
Full suffix	6/6	53.63	+0.67	58.78	+3.38
Native-terminal only	6/6	**54.33**	+0.12	**62.90**	+2.61
Suffix horizon 1	6/6	53.87	+0.56	59.15	+3.38
Suffix horizon 2	6/6	53.60	**+0.71**	58.44	**+3.43**
Suffix horizon 4	6/6	53.49	+0.70	58.85	+3.34

Table 25: **Fine-grained state and fitting controls.** Panel (a) removes individual state channels or changes checkpoint weighting at saving-matched calibration points. Panel (b) changes only the Ridge regularization strength and uses the same calibration rule as Full.

(a) State channels and weighting

Variant	VMME M	VMME Δ	LVB M	LVB Δ
Full	53.63	**+0.67**	58.78	+3.38
w/o belief change	52.88	+0.67	57.73	+3.32
w/o consumed cost	53.00	+0.62	59.41	+3.10
w/o tool type	**53.68**	+0.60	59.70	**+3.52**
w/o duration + subtitle	53.59	+0.66	58.37	+3.24
Checkpoint-uniform weight	53.53	+0.59	**61.55**	+3.20

(b) Ridge regularization

α	VMME M	VMME Δ	LVB M	LVB Δ
0.1	51.05	+0.43	53.81	+2.87
1.0	51.29	+0.48	55.58	+3.12
10.0	**53.63**	+0.67	**58.78**	+3.38
100.0	51.52	+0.57	55.94	**+3.58**
1000.0	48.61	**+0.89**	52.05	+3.55

Native-terminal supervision is lower on both Video-MME and LongVideoBench. Finite-horizon suffix targets are also competitive: horizons 2 and 4 closely match the full-suffix result. These results indicate that the main gain comes from learning a continuation-aware stopping target; the detailed continuation-cost parameterization has a smaller effect on the benchmark averages.

E.5 State channels, trajectory weighting, and regularization

Table 25 removes individual groups of causal state variables and varies the Ridge fit.

Individual state channels are partly redundant: removing belief change, consumed cost, tool type, or the static duration/subtitle attributes changes the aggregate result only slightly. The larger Belief-only and Execution-state-only variants in Table 23 provide the stronger test of the two state blocks. Ridge regularization also changes the selected operating point; $\alpha = 10$ gives the paper's balance of Video-MME accuracy and saving, while nearby and larger values remain competitive.

E.6 Training-video scaling

We refit both the checkpoint-depth probe temperatures and Gap-Ridge using 10, 20, 30, or all 50 training videos. For 10–30 videos, three fixed random orders of the canonical 50-video training role define nested subsets; the 100-video calibration role is unchanged.

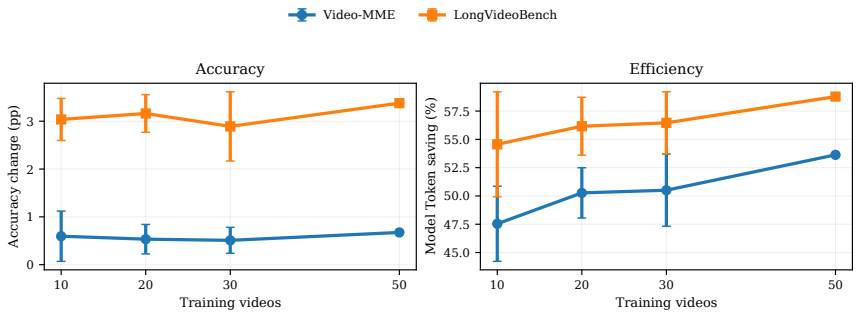

Figure 8: **Training-video scaling.** For 10, 20, and 30 videos, points show the mean over three fixed nested training orders and error bars show one standard deviation; the 50-video point is the canonical training role. Probe temperatures and Gap-Ridge are refit for every subset, while the 100-video calibration role remains fixed.

Table 26: **Probe cadence and probability calibration.** Values are means over all six agent–model cells under the shared calibration rule.

(a) Decision-check cadence

Cadence	VMME M	VMME Δ	LVB M	LVB Δ
Every checkpoint	**53.63**	+0.67	**58.78**	**+3.38**
Every 2 checkpoints	44.64	**+0.90**	51.02	+2.98
Every 3 checkpoints	46.10	+0.67	51.73	+2.31

(b) Option-probability calibration

Calibration	VMME M	VMME Δ	LVB M	LVB Δ
Raw ($T = 1$)	52.44	+0.51	**59.02**	+2.94
Single global T	51.45	+0.34	57.42	+2.84
Depth-group T	**53.63**	**+0.67**	58.78	**+3.38**

Table 27: **Held-out option-probability calibration.** Temperatures are fit only on the Video-MME training role. NLL and multiclass Brier score are pooled over eligible checkpoints of the six agent–model cells; lower is better.

Evaluation set	Checkpoints	Raw NLL	Depth-T NLL	Raw Brier	Depth-T Brier
Video-MME	45,620	1.945	**1.216**	0.793	**0.653**
LongVideoBench	30,460	1.879	**1.313**	0.767	**0.661**
MLVU	48,013	1.799	**1.197**	0.759	**0.640**

Useful stopping policies emerge from small subsets: the 10-video fits average $+0.60$ points / 47.54% saving on Video-MME and $+3.04$ / 54.56% on LongVideoBench. Using all 50 training videos gives $+0.67$ / 53.63% and $+3.38$ / 58.78%, respectively.

E.7 Probe cadence and probability calibration

Finally, we vary how often the stopping policy is evaluated and how the auxiliary option probabilities are calibrated.

Checking every second or third native checkpoint reduces both decision opportunities and realized saving under the current selector; LongVideoBench accuracy also falls. Evaluating every checkpoint therefore retains the strongest overall accuracy–efficiency profile. Raw logprobs and a single global temperature remain viable, while the checkpoint-depth temperatures used by CASE give the highest mean accuracy change on both benchmarks.

The depth-group temperatures also improve probability quality outside the fitting videos: held-out NLL and Brier score decrease on Video-MME and under zero-shot transfer to LongVideoBench and MLVU.

F Stopping-Behavior and Mechanism Analysis

The preceding sections establish the aggregate accuracy–efficiency trade-off. We now examine how that trade-off is produced at the level of individual answers, stopping locations, and learned state signals. The analysis uses the frozen evaluation trajectories and operating points.

F.1 Answer transitions

A net accuracy change can hide substantially different behavior across questions. Table 28 therefore reports the two directional transitions that determine the change in accuracy: Native wrong $\rightarrow$ CASE correct and Native correct $\rightarrow$ CASE wrong. Their difference, divided by the full evaluation size, is exactly the reported CASE$-$Native accuracy change.

Both directions occur across the three benchmarks. For example, VideoSeek–Qwen on Video-MME corrects 197 Native errors and changes 154 Native-correct answers to wrong, yielding the net $+1.91$-

Table 28: **Per-question answer transitions.** $W \to C$ and $C \to W$ count correctness changes from Native to CASE; their difference divided by the full benchmark size gives the net accuracy change.

(a) Video-MME

Agent	Model	$W \to C$	$C \to W$	Both correct	Net Δ (pp)
VideoSeek	Qwen3.5	197	154	1012	+1.91
VideoSeek	GLM-4.6V	62	78	1191	-0.71
VideoSeek	InternVL3.5	167	171	1018	-0.18
AVP	Qwen3.5	105	99	1364	+0.27
AVP	GLM-4.6V	72	81	1222	-0.40
AVP	InternVL3.5	189	118	1032	+3.16

(b) LongVideoBench

Agent	Model	$W \to C$	$C \to W$	Both correct	Net Δ (pp)
VideoSeek	Qwen3.5	154	111	436	+3.22
VideoSeek	GLM-4.6V	51	53	582	-0.15
VideoSeek	InternVL3.5	162	110	453	+3.89
AVP	Qwen3.5	110	74	715	+2.69
AVP	GLM-4.6V	86	45	641	+3.07
AVP	InternVL3.5	217	116	395	+7.55

(c) MLVU

Agent	Model	$W \to C$	$C \to W$	Both correct	Net Δ (pp)
VideoSeek	Qwen3.5	229	179	812	+2.30
VideoSeek	GLM-4.6V	53	59	1033	-0.28
VideoSeek	InternVL3.5	233	105	873	+5.89
AVP	Qwen3.5	199	102	1103	+4.46
AVP	GLM-4.6V	55	46	1109	+0.41
AVP	InternVL3.5	399	80	661	+14.67

point gain. The same table shows larger favorable flows in several zero-shot settings, especially AVP–InternVL on MLVU.

F.2 Risk–saving curves

As an auxiliary stopping-behavior diagnostic, we measure value-aware truncation risk (VTR). For questions stopped early by method m at threshold θ, VTR is the fraction whose recorded native continuation contains a later checkpoint with lower answer-risk–cost loss:

$$\mathrm{VTR}_m(\theta) = \frac{1}{|\mathcal{S}_m(\theta)|} \sum_{i \in \mathcal{S}_m(\theta)} \mathbb{1}\left[g_{i,\tau_i^{m,\theta}} > 0\right]. \tag{21}$$

Lower VTR is better at a fixed saving level. Figure 9 reports the complete VTR–saving curves across the realized saving range of each method.

F.3 Cost concentration and saved suffixes

CASE saving rises strongly with Native Model Token cost. We rank questions within each agent–model–benchmark cell by Native cost, partition them into deciles, and average the frozen CASE behavior within each decile. The bins provide a post-hoc diagnostic of cost concentration.

CASE changes the cheapest trajectories little and concentrates saving on the expensive tail. Mean Model Token saving rises from 1.52% to 74.16% across the lowest-to-highest Video-MME cost deciles, from 1.21% to 77.02% on LongVideoBench, and from 5.32% to 70.04% on MLVU. Stop rates increase in the same direction. Together with Figure 5, this shows that the efficiency gain comes mainly from removing long Native suffixes on expensive trajectories.

F.4 Policy signals and stopping dynamics

Figure 5 shows substantial variation in the first STOP location across questions, models, and hosts. To inspect the fitted rule itself, Figure 11 visualizes the coefficients of the six selected Ridge policies after feature standardization.

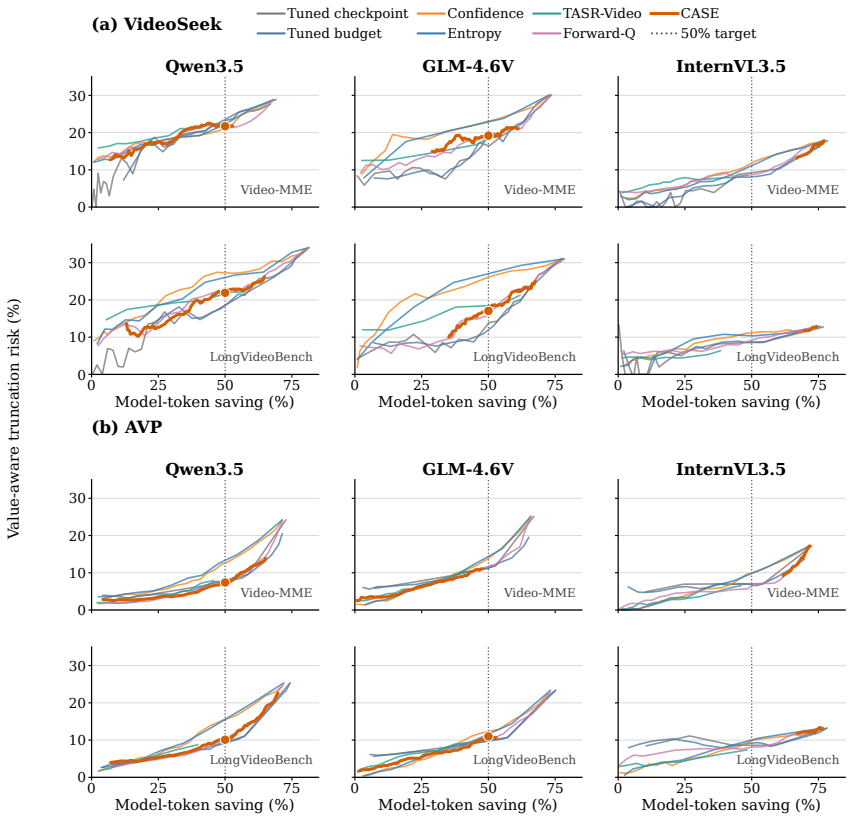

Figure 9: **Value-aware truncation risk versus model-token saving.** Columns are Qwen3.5, GLM-4.6V, and InternVL3.5; rows cover Video-MME and LongVideoBench for VideoSeek and AVP. Lower VTR is better at a fixed saving level.

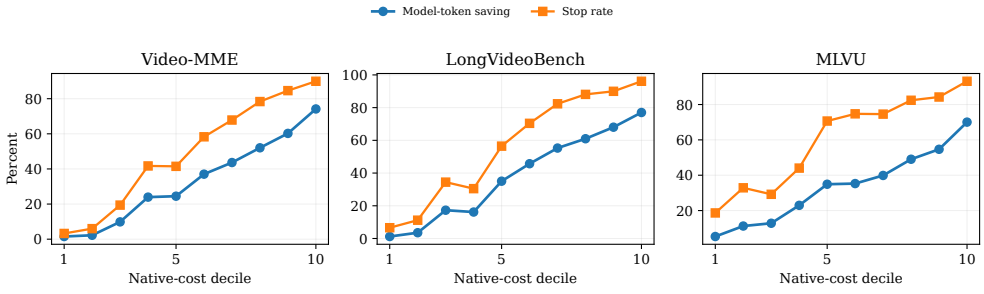

Figure 10: **Cost concentration of CASE stopping.** Questions are partitioned by Native Model Token cost within each agent–model–benchmark cell and then averaged over the six cells.

The learned coefficients vary across hosts and models. Belief variables, checkpoint depth, accumulated cost/evidence, and host-specific execution channels contribute in different combinations, consistent with the feature-removal results in Section E.5.

G Robustness

G.1 Sensitivity to the training/calibration assignment

We form nine balanced alternative assignments of the same 150 Video-MME development videos into 50 training and 100 calibration videos. Each assignment refits the probe temperatures and Gap-Ridge

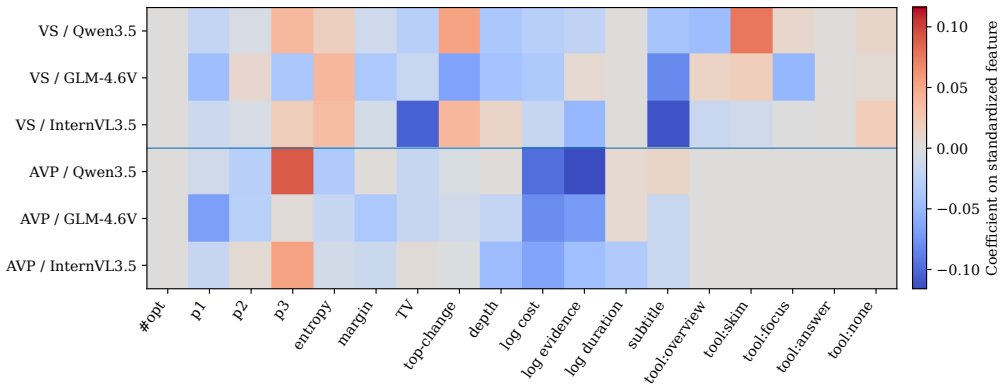

Figure 11: **Standardized Ridge coefficients of the six selected CASE policies.** Features are standardized within each fitted policy; the plot summarizes the learned linear stopping signals across hosts and models.

Table 29: **Balanced development-role resampling.** Mean ± sample standard deviation over nine alternative 50-train/100-calibration assignments of the same 150 Video-MME development videos. Every split is refitted and selected with the same calibration rule; LongVideoBench uses the frozen Video-MME policy.

Agent	Model	VMM M	VMM Δ	LVB M	LVB Δ	Strict splits
VideoSeek	Qwen3.5	51.82 ± 5.00	+1.39 ± 0.64	68.82 ± 7.03	+1.60 ± 1.88	2/9
VideoSeek	GLM-4.6V	45.22 ± 8.13	-1.30 ± 0.90	46.62 ± 23.52	-1.60 ± 1.69	2/9
VideoSeek	InternVL3.5	61.03 ± 16.43	+0.07 ± 0.91	59.28 ± 24.51	+3.56 ± 1.00	7/9
AVP	Qwen3.5	61.75 ± 8.31	-1.06 ± 1.22	61.07 ± 14.60	-0.79 ± 2.92	8/9
AVP	GLM-4.6V	40.75 ± 1.83	-0.49 ± 0.18	41.05 ± 7.78	+2.88 ± 0.59	0/9
AVP	InternVL3.5	65.07 ± 3.03	+1.93 ± 1.25	74.12 ± 2.81	+7.01 ± 0.48	9/9

models and then applies the same calibration rule; the 750 Video-MME evaluation videos remain unchanged.

Across the 54 alternative split–cell combinations, 28 use the primary branch and 26 use the 40% fallback. Averaged across all six agent–model pairs and nine assignments, Video-MME gives $+0.09$ points with 54.27% Model Token saving, while LongVideoBench gives $+2.11$ points with 58.50% saving. The per-cell standard deviations in Table 29 reflect both refitting and the selected stopping aggressiveness.

G.2 Trajectory-seed robustness

We isolate sensitivity to stochastic host trajectories by freezing the canonical seed-42 CASE controller and applying it unchanged to native trajectories generated with seeds 43 and 3407. Probe temperatures, feature standardization, Ridge coefficients, the selected cost weight λ, boundary b, and the Medium/Long activation rule are all fixed from seed 42; the target seeds are used only to generate the trajectories on which the controller is replayed and scored. No target-seed fitting, calibration, or operating-point selection is performed. The seed-42 member of each three-seed summary is the canonical paper run, so all three seeds are evaluated with the same controller parameters.

Table 30: **Trajectory-seed robustness.** Mean ± sample standard deviation over native-trajectory seeds 42, 43, and 3407. The complete CASE controller learned and selected with seed 42 is held fixed for all three seeds; only the native host trajectories change. Δ is accuracy change in percentage points and M is Model Token saving (%).

Agent	Model	Video-MME		LongVideoBench		MLVU	
		M	Δ	M	Δ	M	Δ
VideoSeek	Qwen3.5	51.89 ± 3.25	+1.59 ± 2.33	61.42 ± 4.13	+2.37 ± 2.01	56.92 ± 2.30	+2.30 ± 0.83
VideoSeek	GLM-4.6V	42.17 ± 8.59	-0.56 ± 1.34	49.35 ± 6.82	-0.20 ± 1.12	41.97 ± 5.94	-0.78 ± 0.51
VideoSeek	InternVL3.5	59.50 ± 10.00	-0.21 ± 0.18	71.01 ± 4.10	+5.68 ± 1.80	56.47 ± 6.52	+7.41 ± 1.43
AVP	Qwen3.5	56.36 ± 0.34	+0.06 ± 0.22	55.94 ± 1.01	+1.84 ± 1.22	67.14 ± 1.56	+3.99 ± 0.48
AVP	GLM-4.6V	30.13 ± 0.70	+0.10 ± 0.51	34.76 ± 0.90	+3.86 ± 0.87	30.50 ± 0.62	+0.44 ± 0.46
AVP	InternVL3.5	61.55 ± 0.87	+1.69 ± 1.28	66.41 ± 2.31	+4.06 ± 4.05	48.46 ± 4.58	+8.95 ± 5.00

Across the 18 agent–model–benchmark cells, the frozen controller averages +2.37 accuracy points and 52.33% Model Token saving. Averaging the per-cell variability across the three trajectory seeds gives a standard deviation of 1.42 points for accuracy change and 3.59 points for Model Token saving. Thus the deployed controller retains a similar efficiency profile when the native agent follows different stochastic trajectories.

H Experimental Protocol and Reproducibility

This section records the experimental roles and runtime configuration used throughout the paper. The protocol is fixed at the video level: all questions associated with one Video-MME video remain in the same role, and the learned stopping policy is selected on Video-MME before zero-shot evaluation on LongVideoBench and MLVU.

H.1 Video-MME partitions and data roles

Video-MME contains 900 videos with three questions per video. We use 50 videos (150 questions) to fit probe calibration and the stopping score, 100 disjoint videos (300 questions) to select the operating point, and the remaining 750 videos (2250 questions) for the reported Video-MME evaluation. Table 31 summarizes both the partition sizes and the stage at which each role is used.

Table 31: **Data roles in the experimental pipeline.** A check mark indicates that labels from that partition enter the corresponding stage. LongVideoBench and MLVU provide frozen-policy evaluation.

Partition	Videos	Questions	Probe temp.	Ridge fit	(λ, b) selection	Reported score
Video-MME train	50	150	✓	✓	–	–
Video-MME calibration	100	300	–	–	✓	–
Video-MME evaluation	750	2250	–	–	–	✓
LongVideoBench validation	753	1337	–	–	–	✓
MLVU multiple-choice	1242	2174	–	–	–	✓

The cost scale S, four checkpoint-depth probe temperatures, and the five Ridge models associated with the cost-weight grid are fit on the 50-video training role. The 100-video calibration role then evaluates the 405 candidate (λ, b) pairs and applies Algorithm 3. The resulting coefficients, probe temperatures, λ, and boundary remain frozen for Video-MME evaluation, LongVideoBench, and MLVU.

H.2 Zero-shot benchmark transfer

LongVideoBench and MLVU are evaluated after the complete Video-MME stopping policy is frozen. For each agent–model pair, the transferred object includes the probe calibration, 18-dimensional feature definition, Ridge coefficients, selected λ and b, Medium+Long activation rule, prefix-finalizer

interface, parser, and cost accounting. LongVideoBench contains 1337 scored questions over 753 videos. The MLVU multiple-choice set contains 2175 raw questions over 1242 videos; one objectively blank item is excluded for both Native and stopping methods, leaving 2174 scored questions. Target-benchmark labels enter at the final scoring stage.

H.3 Models and host-agent configurations

Table 32 gives the exact local checkpoint identities used for the three VLM backends. Revision hashes come directly from the Hugging Face download metadata stored with each checkpoint.

Table 32: **Vision-language model checkpoints.** Service inference uses bfloat16 compute for all three backends; Qwen retains its FP8 checkpoint weights.

Paper name	Upstream checkpoint	Revision	Weights
Qwen3.5-35B-A3B-FP8	`Qwen/Qwen3.5-35B-A3B-FP8`	`9d1823d2dee688a6b25e77009dc727688c44936e`	FP8, 128×128 blocks
GLM-4.6V-Flash	`zai-org/GLM-4.6V-Flash`	`411bb4d77144a3f03accbf4b780f5acb8b7cde4e`	bfloat16
InternVL3.5-30B-A3B-HF	`OpenGVLab/InternVL3_5-30B-A3B-HF`	`647fbc96f5fc9798f9f5811ddf622f07a09ccb9f`	bfloat16

For each host, the host-agent configuration is fixed across the three VLM backends and shared by Native, CASE, and all stopping baselines. CASE intervenes only in termination and otherwise leaves native evidence acquisition unchanged. Canonical reported results use trajectory seed 42. Appendix G.2 keeps the complete seed-42 CASE controller fixed while changing only the native trajectory seed to 43 or 3407. In both hosts, CASE is active for Medium and Long videos while Short videos retain native execution.

H.4 Runtime and software environment

The three model services use separate model-specific environments but the same core inference stack: vLLM 0.23.0, Transformers 5.12.1, and PyTorch 2.11.0+cu130. Qwen3.5 and GLM-4.6V-Flash run with tensor parallelism 1; InternVL3.5 uses tensor parallelism 2. The service context is 65,536 tokens for Qwen/GLM and 32,768 tokens for InternVL. For AVP–InternVL3.5, a context-length rejection triggers one retry after resampling the requested visual frames to 70%; the context window is unchanged.

The experiments run on an NVIDIA server containing four RTX 5880 Ada Generation GPUs (49,140 MiB each) and two RTX PRO 6000 Blackwell Server Edition GPUs (97,887 MiB each). The installed driver is 580.65.06; `nvidia-smi` reports CUDA Version 13.0. The final full-service launchers use single-GPU services for Qwen/GLM and two GPUs for InternVL, consistent with the tensor-parallel settings above.

H.5 Statistical protocol

Accuracy uncertainty estimates respect the video grouping. For a fixed stopping policy, we form paired Native–CASE per-question correctness differences, group all questions from the same video, and perform 10,000 video-clustered bootstrap draws with seed 42. Each draw resamples videos with replacement and retains every question belonging to each sampled video; the reported 95% interval uses the 2.5th and 97.5th percentiles. Calibration also records a one-sided 95% lower bound from the mean and standard error of the per-video paired accuracy differences. The strict branch of Algorithm 3 uses an LCB threshold of -1 pp, and an empty strict set activates the 40% fallback.

The development-role experiment in Appendix G measures a different source of variation: it rebuilds the fitted policy after reassigning the same 150 development videos between the 50-video training and 100-video calibration roles. Thus the clustered bootstrap summarizes evaluation-sample uncertainty for a frozen policy, whereas the balanced resplits summarize how the fitted and selected stopping policy changes with the development partition.

H.6 REPRODUCTION ASSETS

The anonymous supplementary code package provides the CASE implementation used for the reported experiments, the fixed data-role protocol and six operating points in `configs/paper_protocol.json`, the serialized online policies, and the host-agent integration interfaces. Public datasets, model weights, upstream host-agent repositories, and large cached trajectory artifacts are not duplicated. Appendix B contains the causal stopping protocol and prompts used by the termination framework.