ACT ON THE LOW FREQUENCIES: PREFIX EXECUTION OF VISION-LANGUAGE-ACTION CHUNKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language-action policies predict chunks of robot controls and ship every chunk in full, and how much of a chunk a robot needs to finish its task has remained unmeasured in closed loop. We introduce *frequency-prefix execution* (FPE), a training-free scheme that transforms each finished chunk into a temporal cosine basis, transmits a fixed number of its lowest-frequency rows, and lets the robot reconstruct and execute the whole chunk under the unchanged policy, horizon and replanning schedule. The basis makes the discarded detail exactly accountable: the reconstruction error of a prefix equals the energy of the omitted rows, and under a contraction condition the accumulated state deviation stays within the largest single-chunk residual. On the official LIBERO protocol with OpenVLA-OFT, two cosine rows keep 98.8% success on Spatial against 98.6% for quantized full codes at 54% of the native payload, and four rows keep 92.6% against 92.2% on LIBERO-10 at 72% of the payload, both passing a registered non-inferiority test. On Goal the tolerance to truncation follows the task, which motivates PAIRGAP, a shortest-path certificate that bounds, at a single restored state, the regret of executing a prefix against the best option in a fixed bank of prefixes and full codes. Together they show that a robot can act on a fraction of each predicted chunk, and they identify the tasks and states that call for the full code.

1 INTRODUCTION

A vision-language-action policy reads a camera image and a language instruction, and predicts a short sequence of robot controls, called an action chunk (Zhao et al., 2023; Black et al., 2025a). The robot carries out that whole sequence before it asks for another prediction (Kim et al., 2025; Zhao et al., 2026). Each request moves a block of numbers from the policy to the robot, and across an episode these transfers consume bandwidth and time. What decides the outcome of the episode is the motion the robot actually performs.

A compact code can stand for a full motion. Frequency transforms and learned ordered codes represent a control sequence with a few leading coefficients, and they are judged by how closely they reconstruct the original action (Pertsch et al., 2025; Liu et al., 2026a; Lian et al., 2026). A robot that executes a partial code closes a loop around it: the reconstruction error shifts where the robot ends up, together with every later observation and prediction. Closeness of reconstruction and success on the task are therefore separate questions, and the second one has not been measured under a fixed execution protocol.

We propose *frequency-prefix execution* (FPE), a training-free scheme that answers this second question for one family of codes. Once the policy has finished a prediction, an encoder transforms the chunk along time into a cosine basis and transmits a fixed number of its lowest-frequency rows. The receiver zero-fills the missing coefficients, reconstructs the entire chunk, and the robot executes all of it. The policy weights, the execution horizon and the replanning schedule stay fixed, so a single integer sets the message size of every request. Figure 1 places this transmission decision between prediction and execution. We count transmitted bits and policy calls separately, so each intervention carries its own cost.

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.

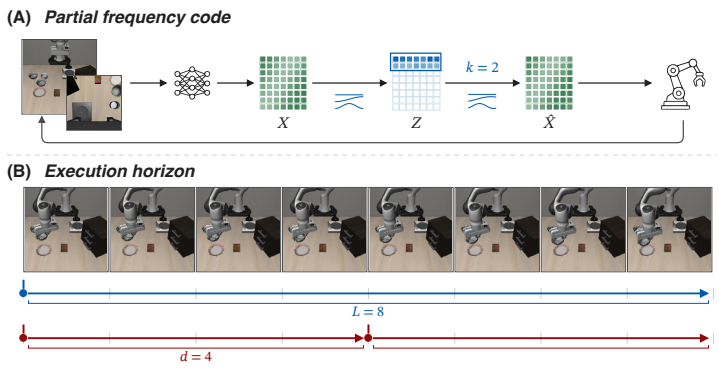

Figure 1: Partial frequency code and execution horizon. (A) A k-row prefix of the coefficients Z reconstructs the full chunk $\hat{X}$. (B) Frequency and time truncation schedules.

The cosine basis makes the discarded detail exactly accountable. Each transmitted row carries a known share of the energy of the chunk, and the reconstruction error of a prefix equals the energy of the rows left behind. Repeated truncation feeds that error back into the state, and under a contraction condition on the full-code loop the accumulated deviation is bounded by the largest single-chunk residual. The experiments measure task success over complete episodes and leave the constants unestimated.

We evaluate fixed frequency prefixes on the official LIBERO protocol (Liu et al., 2023) with a released open vision-language-action checkpoint (Kim et al., 2025). Native actions, quantized full codes and a shorter execution horizon serve as controls. On Spatial, task success is retained and the payload of every request falls below the full-code cost. On Goal, success moves with the task, which locates the sensitivity in the task itself. Single-decision experiments on restored states isolate one truncated chunk, and spectral measurements connect those differences to retained energy and reconstruction error.

The variation across tasks raises a local question: at one state, how much evidence is needed before a partial code can be executed on purpose? We study that question with PAIRGAP, a graph-based procedure that bounds the additional expected failure cost of a prefix against the best option in a fixed bank of candidate prefixes and full codes. Restored simulator rollouts supply the evidence, and shortest paths in a single difference-constraint graph give the tightest bound those edges support. Applying the procedure inside every closed-loop request remains open.

2 RELATED WORK

Action representations. Action tokenizers turn a continuous control sequence into a compact code that a policy generates and a robot decodes (Pertsch et al., 2025; Liu et al., 2026a; Lian et al., 2026). These methods compress an action sequence for training and storage, and they assess a code by how closely it reconstructs the original action. We use the same cosine representation as the executable interface, transmit a fixed number of its low-frequency rows, and measure task success over complete episodes of the official protocol. Quantized full codes and a shorter execution horizon enter as separate controls, so truncation and quantization carry their own comparisons. Appendix D covers streaming and chunk-refinement variants of this pipeline.

Execution horizons and real-time control. Chunked policies must choose how long to execute before incorporating another observation, and recent methods set that length from the overlap of inference with execution or from the estimated state difficulty (Black et al., 2025b; Zhao et al., 2026; Chen et al., 2026a). Each of these interventions changes when feedback reaches the controller. Our scheme keeps the execution horizon and the replanning schedule of the underlying policy and varies the transmitted code alone, with a time-truncation control varying the horizon under that same frozen policy. Total episode calls and serialized bits count the resource consequences of both choices.

Reconstruction error and closed-loop stability. The discrete cosine transform supplies a fixed orthogonal basis for temporal signals (Ahmed et al., 1974), whose orthogonality expresses the energy a prefix discards as a sum of retained row energies. The behavioral effect of that residual depends on the feedback dynamics of the closed loop. Analyses of action chunking bound error propagation through open-loop incremental stability and policy regularity (Zhang et al., 2025), and input-to-state stability describes how a bounded perturbation propagates through a contractive system (Jiang & Wang, 2001). We apply this reasoning to repeated truncation at a fixed horizon, where the resulting conditional bound takes the reconstruction residual of one chunk as its input and the accumulated state deviation as its output. Task success in contact-rich environments supplies the behavioral evidence for the same scheme.

Rollout-based certification. Rollout-based procedures attach statistical evidence to decisions about robot policies (Snyder et al., 2026; Chen & Weng, 2026). Certifying a partial action code fixes two further objects: the comparator set, which contains the other candidate motions and code depths available at the same state, and the accounting of received bits. We represent bounds between these options as edges of a difference-constraint graph and compose them into a regret certificate, then compare the simulator cost of that certificate with certification from an option's own failure outcomes and with receiving every candidate code in full. Appendix D discusses broader action-generation and adaptive-compute work.

3 Frequency-prefix execution

3.1 A partial code for a full execution horizon

Frequency-prefix execution sends a partial action code and executes the complete action horizon at every policy call. A policy observes the state and predicts a chunk $X \in \mathbb{R}^{L\times m}$ containing L control steps and m action dimensions, and the receiver retains a fixed number k of low-frequency rows throughout an episode and executes all L reconstructed steps before calling the policy again. The stopping rule and the step limit match the full-code arm.

To form the code, we apply the orthonormal type-II cosine transform C along time, $Z = CX$, following the representation used in FAST (Pertsch et al., 2025). Row r of Z holds the r-th frequency of every action dimension, and the codec quantizes the coefficients and serializes the rows in increasing frequency. Before quantization, the first k rows reconstruct

$$\hat{X}_k = C^\top P_k Z, \qquad P_k = \operatorname{diag}(\underbrace{1,\dots,1}_{k},0,\dots,0), \tag{1}$$

an executable chunk of all L steps. The projection P_k keeps the leading k rows and discards the rest. The implemented decoder uses quantized coefficients and zero-fills the omitted rows, and quantization contributes its own reconstruction error.

The time-truncation control executes the leading d steps before replanning. Both interventions follow a completed policy prediction. We measure episode success, policy calls and serialized bits, and leave policy-generation and network latency outside this evaluation.

3.2 Communication and policy-call costs

The payload $b(k)$ counts the bits the receiver obtains for k rows. A header of β_h bits carries the shape, the quantization scale and the candidate binding. Rows travel in geometric bands, each opened by a segment of β_s bits, and every row carries a β_r-bit index and m coefficients of β_c bits each (Figure 2A). A prefix of $k = 2^j \le L$ rows spans $1 + \log_2 k$ bands, so

$$b(k) = \beta_h + \beta_s\,(1 + \log_2 k) + (\beta_r + m\,\beta_c)\,k. \tag{2}$$

Proposition 2 derives these constants, and the leading d steps coded as their own horizon-d chunk also carry $b(d)$. Over H executed steps, a controller that executes d of L steps per call issues $N(d) = \lceil H/d \rceil$ calls and receives $B(d) = N(d)\,b(d)$ bits, so for d and L dividing H (Proposition 3)

$$\frac{N(d)}{N(L)} = \frac{L}{d}, \qquad \frac{B(d)}{B(L)} = \frac{L}{d}\cdot\frac{b(d)}{b(L)}. \tag{3}$$

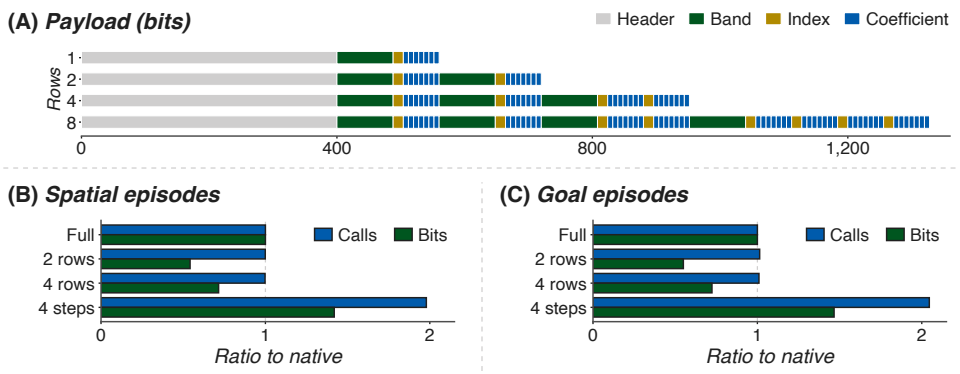

Figure 2: Code layout and episode cost. (A) Wire fields per retained-row count. (B, C) Calls and bits relative to native execution.

For the same H, frequency truncation preserves the full-code call count and scales bits by $b(k)/b(L)$, and time truncation multiplies calls by L/d. Episode lengths vary with success and termination, so results use actual call totals and their serialized bits. Native execution is priced through the full-chunk codec as the common baseline.

3.3 Reconstruction error and closed-loop deviation

Discarded frequency rows determine the reconstruction error before quantization. Let $\Sigma = \mathbb{E}[XX^\top]$ be the temporal second-moment matrix of the chunks pooled over action dimensions, c_r the r-th cosine basis vector and $e_r = c_r^\top \Sigma c_r$ the energy of row r.

Lemma 1 (Energy of a frequency prefix)**.** *For every chunk, $\|X - \hat{X}_k\|_F^2 = \sum_{r>k} \|Z_r\|^2$ before quantization, and $\mathbb{E}\|X - \hat{X}_k\|_F^2 = \sum_{r>k} e_r$. Among all rank-$k$ orthogonal projections Q along time, $\mathbb{E}\|X - QX\|_F^2 \geq \sum_{r>k} \lambda_r$, where $\lambda_1 \geq \cdots \geq \lambda_L$ are the eigenvalues of Σ, with equality for the Karhunen–Loève projection onto the leading eigenvectors. If the row energies are non-increasing, $e_1 \geq e_2 \geq \cdots \geq e_L$, the k-row prefix has the smallest expected residual among all choices of k cosine rows.*

Parseval's identity gives the residual, and the Ky Fan maximum principle gives the optimal rank-k projection. For a stationary first-order Markov signal with lag-one correlation near one, the cosine basis approaches the Karhunen–Loève basis (Ahmed et al., 1974), so a cosine prefix nearly attains the lower bound of Lemma 1.

Repeated prefix execution changes subsequent states. Input-to-state stability relates this accumulated deviation to the one-chunk residual. Let x_t be the state at the t-th chunk boundary, F the chunk dynamics under shared exogenous randomness, $\bar{u}(x)$ and $u_k(x)$ the chunks executed from the full code and from k rows, and $\varepsilon_t = \|u_k(x_t) - \bar{u}(x_t)\|$ the implemented residual, including quantization, at the visited state.

Proposition 1 (Deviation tube of every-chunk truncation)**.** *Suppose the full-code closed loop $G(x) = F(x, \bar{u}(x))$ contracts in a norm $\|\cdot\|_P$ on a common invariant state domain, $\|G(x) - G(y)\|_P \leq \rho\|x - y\|_P$ with $\rho < 1$, and $\|F(x,u) - F(x,u')\|_P \leq \ell\|u - u'\|$. Started from the same state, the truncated and full-code trajectories satisfy $d_{t+1} \leq \rho\, d_t + \ell\, \varepsilon_t$ for $d_t = \|x_t - \bar{x}_t\|_P$ with $\bar{x}_t$ the full-code trajectory, hence*

$$\|x_t - \bar{x}_t\|_P \leq \ell \sum_{s<t} \rho^{\,t-1-s} \varepsilon_s \leq \frac{\ell \, \sup_s \varepsilon_s}{1-\rho}. \tag{4}$$

The recursion follows from the triangle inequality through $F(x_t, \bar{u}(x_t))$. Equation (4) gives the geometric gain of discrete-time input-to-state stability (Jiang & Wang, 2001) (Appendix A). The largest one-chunk residual sets the tube radius; Lemma 1 relates this residual to dropped energy up to quantization error. For a terminal goal-set criterion, success transfers whenever the full-code

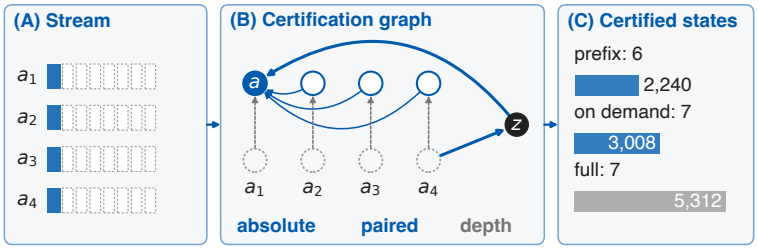

Figure 3: Certified execution depth. (A) Row one of four candidates has arrived. (B) The certification graph and the witness path through z. (C) Certified states of eight, with bits per decision.

terminal state lies inside the goal set by more than this radius. Trajectory-wide constraints require the corresponding clearance at every relevant time.

The constants ρ and ℓ are hard to estimate for contact-rich manipulation, so we measure the behavioral consequence directly over complete episodes. A prefix is *behaviorally sufficient at tolerance* τ when its closed-loop success is at least that of the full code minus τ, and we test this suite-level criterion with paired rollouts from shared states and seeds.

4 A PER-STATE CERTIFICATE FOR THE EXECUTABLE PREFIX

The every-chunk scheme of Section 3 holds one row count for a whole episode, and the cost of a prefix depends on the state the robot occupies. A controller that faces one state at a time benefits from a bound on the regret of executing a prefix there. This section presents PAIRGAP, a procedure that decides whether an executable prefix can be committed at one state, and Figure 3 traces one decision. Appendix F gives the setup and the algorithm, Appendix A the proofs and Appendix G the full experiments.

4.1 CERTIFYING A PREFIX BY A GRAPH OF COST DIFFERENCES

At a restored state a fixed policy proposes K candidate action chunks, and an *execution option* $a = (i, k)$ is the k-row prefix of candidate i, with binary failure cost $Y_a \in \{0, 1\}$ and mean $m_a = \mathbb{E}[Y_a]$. The comparator set $\mathcal{A}$ holds the registered prefix and full-code options, including those whose full rows have not arrived, and the received bits enter as a separate decision-level cost. Executing a incurs the regret $r(a) = m_a - \min_{b \in \mathcal{A}} m_b$ against every candidate at every depth. An ϵ certificate is a bound $r(a) \le \epsilon$ that holds for every cost vector consistent with the evidence, so it covers the full codes in flight.

PAIRGAP turns each piece of evidence into a bound on a cost difference. A directed edge $b \to a$ with weight $w(b, a)$ records $m_a - m_b \le w(b, a)$ on the vertex set $\mathcal{A} \cup \{z\}$, where the reference vertex z has $m_z = 0$. Reference edges carry $\mu_b \ge 0$ and need no data, an absolute edge carries an upper confidence bound on m_a from n samples of the executed option, a signed paired edge between received options runs both from the same n seeds so common noise cancels, and a depth edge connects an unreceived full code to its received prefix. Summing the inequality along a path bounds $m_a - m_b$ by the path length, so the exact evidence-consistent regret reads as a shortest-path distance,

$$R_w(a) = \max_{b \in \mathcal{A}} d(b, a), \qquad a \text{ carries an } \epsilon \text{ certificate} \iff R_w(a) \le \epsilon. \tag{5}$$

Theorem 1 proves that (5) is exact for these constraints, so the bound is attained and no tighter value follows from the same evidence. The maximizing source and its shortest path form the *witness path*. A union bound over the allocated edge errors gives the state-marginal validity of Proposition 4.

The evidence a certificate needs is the evidence its witness path uses. Relational evidence decides exactly on the sources whose reference path exceeds ϵ, an unreceived full code sits at most the depth-edge weight from its received prefix, and a rival that never disagrees with the executed option contributes no shorter path than the absolute bound. Proposition 5 states these conditions, and Appendix F measures which witness path attains each certificate.

The evaluated on-demand rule tests a pre-registered candidate at prefix depth and requests that candidate's full code when the prefix bound exceeds a fixed receipt threshold η, then selects the depth with the smaller bound. The codes order rows by energy, and the two depth tests split the online error budget equally. Shortest paths come from Floyd–Warshall on $2K + 1$ vertices, $O(K^3)$ per update, negligible next to one rollout.

5 EXPERIMENTS

We evaluate whether a fixed frequency prefix remains executable over complete episodes, under a policy, execution horizon, stopping rule and initial states shared by every arm. Four questions organize the experiments. **Q1**, does a fixed frequency prefix preserve task success and reduce the transmitted bits of an episode (Section 5.2)? **Q2**, what does one truncated chunk change at a restored state (Section 5.3)? **Q3**, how much energy does a prefix retain, and what does the short message cost to encode (Section 5.4)? **Q4**, how much simulator evidence does a state-level certificate require (Section 5.5)?

5.1 POLICIES, PROTOCOLS AND REPORTING

Encoders and decoders run on the official LIBERO protocol (Liu et al., 2023) with released checkpoints. OpenVLA-OFT (Kim et al., 2025; 2024) supplies the closed-loop measurements at seed 7 and fifty initial states for each of ten tasks, which gives 500 episodes per suite and arm. LaRA-VLA (Bai et al., 2026) and Flex-π (Yan et al., 2026) supply the single-decision tests, which restore a state, truncate its first chunk, and resume the standard policy over sixteen states, four candidates per state and four shared continuation seeds per candidate and depth. The LIBERO-10 panel uses Flex-π at one denoising step, 40 states across 10 tasks, and 64 shared continuations per candidate and depth.

The codec of Equation (2) uses a $\beta_h = 400$-bit header, $\beta_s = 88$-bit band segments, $\beta_r = 16$-bit row indices and one-byte coefficients, $\beta_c = 8$, so with $m = 7$, retaining one, two, four and eight rows gives 560, 720, 952 and 1,328 bits, as verified on every official OpenVLA-OFT call. Suite non-inferiority uses the registered tolerance $\tau = 2$ percentage points and a one-sided, task-level sign-flip test at 0.025 (Appendix C), and the certificate study tests the one-row DC prefix with receipt threshold $\eta = 0.05$ and reports $\epsilon \in \{0.05, 0.10\}$. Bold and underlined numbers mark the best and second distinct success percentages and bit costs within each population, and each reported contrast names the arm it compares against.

5.2 EVERY-CHUNK EXECUTION

Table 1 reports the five registered arms on both suites, which share the initial states, seed, stopping rule and step limit.

Table 1: Five registered arms on the official protocol. k counts retained rows and d executed steps per call.

Execution	k	d	Success (%) ↑	Calls	Bit ratio ↓
Spatial, 500 episodes per arm					
Native full	–	8	98.4	6,652	1.00
Quantized full	8	8	<u>98.6</u>	6,665	1.00
Frequency prefix	2	8	**98.8**	6,646	**0.54**
Frequency prefix	4	8	**98.8**	6,637	<u>0.72</u>
Time prefix	–	4	98.2	13,164	1.42
Goal, 500 episodes per arm					
Native full	–	8	<u>97.0</u>	7,295	1.00
Quantized full	8	8	**97.6**	7,317	1.00
Frequency prefix	2	8	96.6	7,396	**0.55**
Frequency prefix	4	8	96.8	7,365	<u>0.72</u>
Time prefix	–	4	95.6	14,924	1.47

5.2.1 Two rows keep Spatial success at about half the payload

Both frequency arms meet the registered non-inferiority criterion on Spatial (Figure 4A), and two rows transmit 0.54 of native bits. This saving is the per-call ratio $b(2)/b(8)$ of Equation (2), because frequency truncation leaves the call total at the native level. Two rows reach 98.8% success against 98.6% for quantized full codes, and four rows carry 0.72. The non-inferiority tests give $P = 0.0117$ and $P = 0.0020$. Quantized full execution also meets the criterion against native execution ($P = 0.0039$), so rounding the coefficients and removing rows each leave Spatial success intact. Figure 4C resolves these totals into tasks, and panel B counts the episode-level pairs.

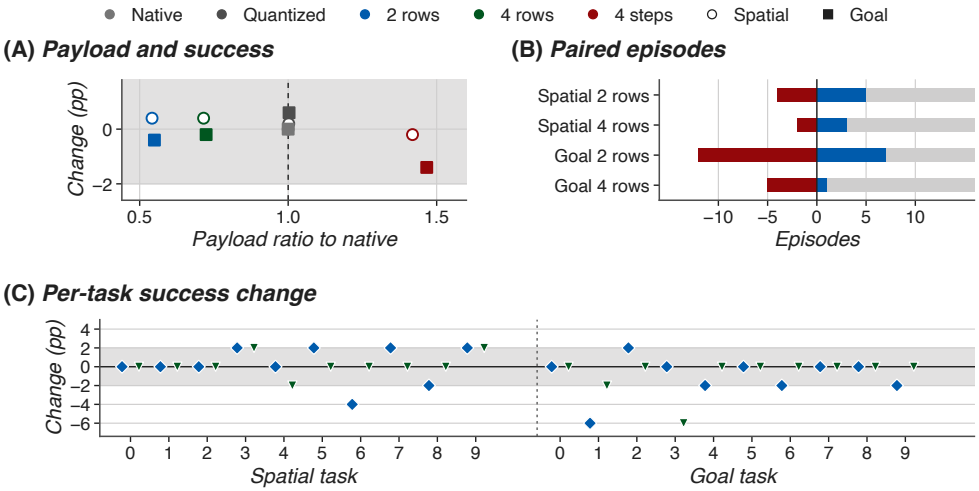

Figure 4: Cost and success. (A) Payload and success relative to native execution. (B) Paired episodes and (C) per-task success change against quantized full codes.

5.2.2 Goal sensitivity depends on the task

On Goal, two rows lose one point of suite success against the quantized full code (Figure 4A), below the two-point decline that would trigger the registered switch to four rows, and neither frequency arm establishes non-inferiority at 0.025 (Appendix C). Figure 4C locates the losses: task 1 declines by six points at two rows and task 3 by six points at four rows, so sensitivity to truncation follows the task.

5.2.3 A longer horizon moves the operating point to four rows

The long-horizon suite answers the same question over ten multi-stage tasks (Appendix L.3). Two rows reach 90.0% against 92.2% for the quantized full code, which is -2.2 points and fails the registered criterion. Four rows reach 92.6% and meet it ($P = 0.0020$). Quantized full execution itself falls -1.6 points below native execution, so this suite separates the two compressions that the shorter suites keep together.

5.2.4 The constant row carries the Goal result, and a discrete gripper changes nothing

A mechanism comparison at its own quantized-full anchor (Appendix L.2) separates payload from row count. On Goal the single 560-bit DC row reaches 97.8% against 97.0% for two rows and 97.0% for the quantized full code, and on Spatial it reaches 97.8%, below two rows by -1.0 points. The Goal outcome therefore needs no temporal row beyond the constant. Replacing the seven-axis continuous gripper by ternary commands at the two-row payload leaves success at 97.2% on Goal and 98.0% on Spatial, so the residual of a two-row prefix lies in the continuous axes.

5.2.5 SHORTENING THE HORIZON COSTS MORE THAN TRUNCATING THE CODE

Executing four of eight steps per call multiplies Spatial calls by 1.98, close to the prediction of Equation (3), and two frequency rows carry 54% of the full payload per call. Goal ratios exceed this prediction because more of its time-truncated episodes reach the step limit (Figure 2C). Time truncation adds feedback at the cost of extra calls and bits, and frequency truncation lowers the payload at the original feedback rate. Appendix L reports per-task counts.

5.3 SINGLE-DECISION PREFIXES ON RESTORED STATES

A single truncated chunk isolates the effect of partial-code execution at a restored state (Figure 6A). These diagnostic banks order quantized cosine rows by energy, and the sixteen-state ladders encode eight steps for LaRA-VLA and ten for Flex-π.

5.3.1 ONE TRUNCATED CHUNK RARELY CHANGES THE OUTCOME

On LaRA-VLA, two rows match the 254 successes of the full code among 256 continuations at 0.54 of its payload, and one row on Flex-π reaches 255 successes against 253 for the ten-row code (Figure 6A). The standard policy resumes after the truncated chunk, so these ladders measure one chunk's deviation, and Section 5.2 measures its accumulation.

On LIBERO-10, two rows fail less often than the eight-row reference over 10,240 paired continuations per depth, and the reference reconstructs every complete quantized ten-step chunk. Figure 5 resolves the pairs over 160 candidates and forty states (IDs in Table 19). At most two additional failures among 256 continuations appear at any state, and the reference failures concentrate in task 6, initial state 49, where both depths fail on most continuations, so those failures come from the state itself. Appendix L gives the state-level counts, and repeated truncation throughout LIBERO-10 episodes remains a separate evaluation.

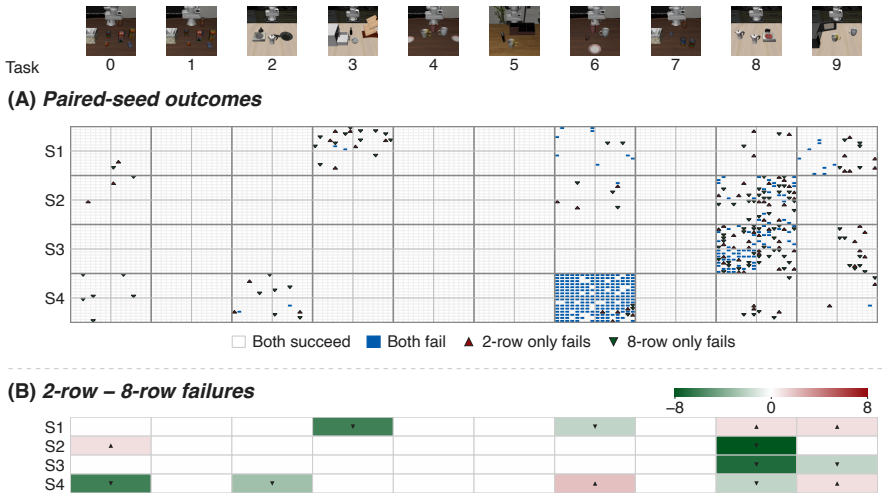

Figure 5: Single-decision Flex-π truncation on LIBERO-10. (A) Paired-seed outcomes at states S1–S4. (B) 2-row minus 8-row failures per state.

5.4 PAYLOAD, ENERGY AND RECONSTRUCTION

5.4.1 SHORT COMPRESSED MESSAGES PAY A CODING OVERHEAD

Priced at 11 bits per token of the released FAST+ vocabulary, a two-row prefix saves 22% of the full LaRA-VLA stream, and every partial prefix pays more bits per retained row than the full code (Figure 15B), since entropy coding already makes the omitted high-frequency rows cheap.

5.4.2 TWO COSINE ROWS HOLD NEARLY ALL CHUNK ENERGY

The energy atlas excludes the constant gripper, preserves the six continuous axes' logged scales and normalizes each chunk by its six-axis energy. Two cosine rows contain 99.69% of LaRA-VLA energy, close to the leading Karhunen–Loève pair on both policies (Figure 15C), so the fixed cosine basis retains almost as much energy as the data-dependent optimum of Lemma 1. Appendix L.7 shows that the mean component contributes materially to this concentration.

5.4.3 RESIDUAL TAILS SET THE DEVIATION BOUND

Two rows raise the mean relative reconstruction error $\|\hat{X} - X\|_F / \|X\|_F$ above that of the quantized full code, and the largest two-row residual reaches 10.0% of the chunk norm on Flex-π. Proposition 1 depends on that largest residual, so these tails decide any calibrated deviation guarantee.

5.5 STATE-LEVEL CERTIFICATION COSTS

5.5.1 ONE-CANDIDATE RECEIPT CERTIFIES AS MANY STATES AS FULL RECEIPT WITH FEWER BITS

Figure 6B compares DC-prefix certification on 8 restored LaRA-VLA states with four candidate chunks each, $\epsilon = 0.10$ and total error $\delta = 0.05$. Both receipt rules certify the same states. The one-candidate rule completes only the registered candidate's code and executes the prefix at 6 of them, and full receipt receives every full code and certifies each state at full depth, so one suffix reaches the same certificates at a lower bit cost.

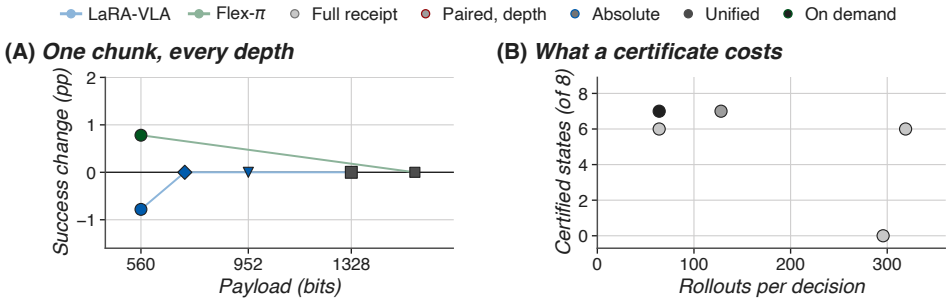

Figure 6: Single-decision truncation. (A) One chunk, every depth, as success change against the full code. (B) What a certificate costs at tolerance 0.10.

5.5.2 RELATIONAL EDGES DO NOT DECIDE ON THIS SATURATED PANEL

Paired and depth edges certify no state at this budget: the candidates succeed on almost every shared seed, so a concordant pair is no shorter than the absolute bound (Appendix G.1.2). The binding constraint is the zero-failure sample size, where the receipt threshold $\eta = 0.05$ requires 86 rollouts per depth test against the 64-sample budget, and no arm certifies at $\epsilon = 0.05$. Held-out audits find no certified pair with empirical regret above 0.10 (Appendix G.1.1).

6 CONCLUSION

This work asks how much of a predicted action chunk a robot must receive to complete the task. Frequency-prefix execution answers that question for one executable code: a fixed number of low-frequency rows of a finished prediction travels to the robot, and the receiver reconstructs and executes the entire chunk under the original policy, horizon and replanning schedule. The cosine basis makes the discarded detail exactly accountable, so the payload of each row count follows from the wire layout and the residual from the energy of the omitted rows. Spatial retains task success below the full-code payload, Goal success changes with the task, and a discrete gripper holds both. PAIRGAP bounds a state-level prefix's regret from restored-simulator rollouts.

AI USE STATEMENT

Generative AI tools assisted with literature discovery, hypothesis development, proof review, implementation, experiment orchestration, result interpretation, visualization, drafting, and editing. The authors verified the primary sources, mathematical arguments, code, measurements, citations, tables, figures, and final rendering. The authors take responsibility for the manuscript.

REPRODUCIBILITY STATEMENT

Section 3 defines fixed-prefix execution and its payload accounting, and Section 4 defines the state-level certificate. Appendix C specifies the official paired episode protocol and its non-inferiority rule. Appendix G provides the independent certification protocol, and Appendix I details state selection, continuation seeds, and held-out evaluation. Machine-readable results generate the manuscript's numerical tables and figures. Code and analysis outputs will be released with the paper.

REFERENCES

N. Ahmed, T. Natarajan, and K. R. Rao. Discrete Cosine Transform. *IEEE Transactions on Computers*, C-23(1):90–93, 1974. doi: 10.1109/T-C.1974.223784.

Sining Ang, Yuguang Yang, and Yan Wang. Adaptive-WAM: Quality-Guided Early-Exit Planning from Intermediate Video-Diffusion Features, 2026. URL https://arxiv.org/abs/2608.06008.

Aniri, Chen Yilin, Jinhe Bi, Junfei Guo, Donglai Ran, Xu Bian, Zengjie Jin, Yujun Wang, Yijun Tian, Volker Tresp, Fei Shen, Tat-Seng Chua, and Yunpu Ma. FabriMAE I Trust Myself? Self-Evaluating VLA Action Generation with Markov Attention Entropy, 2026. URL https://arxiv.org/abs/2608.16697.

Shuanghao Bai, Jing Lyu, Wanqi Zhou, Zhe Li, Dakai Wang, Lei Xing, Xiaoguang Zhao, Pengwei Wang, Zhongyuan Wang, Cheng Chi, Badong Chen, and Shanghang Zhang. Latent Reasoning VLA: Latent Thinking and Prediction for Vision-Language-Action Models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.01166.

Kapilan Balagopalan, Yinan Li, Yao Zhao, Tuan Nguyen, Anton Daitche, Houssam Nassif, and Kwang-Sung Jun. Fixed Budget is No Harder Than Fixed Confidence in Best-Arm Identification up to Logarithmic Factors. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.03972.

Richard Bellman. On a Routing Problem. *Quarterly of Applied Mathematics*, 16(1):87–90, 1958. doi: 10.1090/qam/102435. URL https://doi.org/10.1090/qam/102435.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control. In *Robotics: Science and Systems XXI*, 2025a. doi: 10.15607/RSS.2025.XXI.010. URL https://www.roboticsproceedings.org/rss21/p010.html.

Kevin Black, Manuel Y. Galliker, and Sergey Levine. Real-Time Execution of Action Chunking Flow Policies. In *Advances in Neural Information Processing Systems*, volume 38, pp. 37596–37620, 2025b. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/300ccb2187dedd4edcc07f7e76d8e553-Abstract-Conference.html.

Xiaowei Cai, Yunuo Cai, Bingao Chen, Jingxiao Chen, Zhi Chen, Siyuan Feng, Tengyu Hou, Jingshun Huang, Han Jiang, Runkun Ju, et al. τ_0-VLA: a Hierarchical Robot Foundation Model with World-Model-Guided Test-Time Computation, 2026. URL https://arxiv.org/abs/2608.16885.

Feng Chen, Xianghui Wang, Yuxuan Chen, Boying Li, Yefei He, Zeyu Zhang, and Yicheng Wu. Dynamic Execution Commitment of Vision-Language-Action Models, 2026a. URL https://arxiv.org/abs/2605.11567.

Lingling Chen, Zongyao Lyu, and William J. Beksi. ReconVLA: An Uncertainty-Guided and Failure-Aware Vision-Language-Action Framework for Robotic Control, 2026b. URL https://arxiv.org/abs/2604.16677.

Yujia Chen and Bowen Weng. Betting for Sim-to-Real Performance Certificates, 2026. URL https://arxiv.org/abs/2608.21572.

C. J. Clopper and E. S. Pearson. The Use of Confidence or Fiducial Limits Illustrated in the Case of the Binomial. *Biometrika*, 26(4):404–413, 1934. doi: 10.1093/biomet/26.4.404.

XPolicyLab Community, Tianxing Chen, Yue Chen, Tian Nian, Zijian Cai, Guangyu Chen, Wenwei Lin, Qiwei Liang, Zanxin Chen, Peicheng Xiang, et al. XPolicyLab: A Unified Standard and Open Ecosystem for Robot Policy Evaluation and Deployment, 2026. URL https://arxiv.org/abs/2608.09892.

Mingtong Dai, Lingbo Liu, Yongjie Bai, Yang Liu, Zhouxia Wang, Rui SU, Chunjie Chen, Liang Lin, and Xinyu Wu. RoVer: Robot Reward Model as Test-Time Verifier for Vision-Language-Action Model, 2025. URL https://arxiv.org/abs/2510.10975.

Rianne de Heide. Where Does the Union Bound Go? Best-Arm Identification and Strong FWER Control, 2026. URL https://arxiv.org/abs/2608.19903.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action Elimination and Stopping Conditions for the Multi-Armed Bandit and Reinforcement Learning Problems. *Journal of Machine Learning Research*, 7:1079–1105, 2006. URL https://www.jmlr.org/papers/v7/evendar06a.html.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. LIBERO-Plus: In-depth Robustness Analysis of Vision-Language-Action Models, 2025. URL https://arxiv.org/abs/2510.13626.

Victor Gabillon, Mohammad Ghavamzadeh, and Alessandro Lazaric. Best Arm Identification: A Unified Approach to Fixed Budget and Fixed Confidence. In *Advances in Neural Information Processing Systems 25*, pp. 3212–3220, 2012. URL https://papers.nips.cc/paper/2012/hash/8b0d268963dd0cfb808aac48a549829f-Abstract.html.

Jake Gonzales, Kazuki Mizuta, Karen Leung, and Lillian J. Ratliff. Safe Probabilistic Planning for Human-Robot Interaction using Conformal Risk Control. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 18676–18683, 2025. doi: 10.1109/IROS60139.2025.11247339. URL https://arxiv.org/abs/2603.10392.

Wenkai Guo, Guanxing Lu, Haoyuan Deng, Zhenyu Wu, Yansong Tang, and Ziwei Wang. VLA-Reasoner: Empowering Vision-Language-Action Models with Reasoning via Online Monte Carlo Tree Search. In *IEEE International Conference on Robotics and Automation*, 2026. URL https://arxiv.org/abs/2509.22643.

Chia-Yu Hsu and Shubhanshu Shekhar. Efficient Sequential Evaluation of Large Language Models, 2026. URL https://arxiv.org/abs/2607.17409.

Suhyeok Jang, Dongyoung Kim, Changyeon Kim, Youngsuk Kim, and Jinwoo Shin. Verifier-free Test-Time Sampling for Vision-Language-Action Models, 2025. URL https://arxiv.org/abs/2510.05681.

Yuhwan Jeong and Kuk-Jin Yoon. Reuse Before You Retrieve: Diagnosing Headroom and Complementarity for Test-Time Augmentation of Embodied Multimodal Policies. In *European Conference on Computer Vision Workshops*, 2026. URL https://arxiv.org/abs/2608.17484.

Zhong-Ping Jiang and Yuan Wang. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37(6):857–869, 2001. doi: 10.1016/S0005-1098(01)00028-0.

Marc Jourdan, Rémy Degenne, and Emilie Kaufmann. An ε-best-arm identification algorithm for fixed-confidence and beyond. In *Advances in Neural Information Processing Systems*, volume 36, pp. 16578–16649, 2023. doi: 10.52202/075280-0727.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. PAC Subset Selection in Stochastic Multi-armed Bandits. In *Proceedings of the 29th International Conference on Machine Learning*, 2012. URL http://icml.cc/2012/papers/359.pdf.

Zohar Karnin, Tomer Koren, and Oren Somekh. Almost Optimal Exploration in Multi-Armed Bandits. In Sanjoy Dasgupta and David McAllester (eds.), *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 1238–1246, Atlanta, Georgia, USA, 2013. PMLR. URL https://proceedings.mlr.press/v28/karnin13.html.

Valentin Kilian, Stefano Cortinovis, and François Caron. Asymptotically Log-Optimal Bayes-Assisted Confidence Sequences for Bounded Means, 2026. URL https://arxiv.org/abs/2605.07964.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag R. Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An Open-Source Vision-Language-Action Model. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713, 2024. URL https://proceedings.mlr.press/v270/kim25c.html.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-Tuning Vision-Language-Action Models: Optimizing Speed and Success. In *Robotics: Science and Systems XXI*, 2025. doi: 10.15607/RSS.2025.XXI.017. URL https://www.roboticsproceedings.org/rss21/p017.html.

Seong-Hee Kim and Barry L. Nelson. Chapter 17: Selecting the Best System. In Shane G. Henderson and Barry L. Nelson (eds.), *Simulation*, volume 13 of *Handbooks in Operations Research and Management Science*, pp. 501–534. Elsevier, 2006. doi: 10.1016/S0927-0507(06)13017-0. URL https://doi.org/10.1016/S0927-0507(06)13017-0.

Jacky Kwok, Christopher Agia, Rohan Sinha, Matt Foutter, Shulu Li, Ion Stoica, Azalia Mirhoseini, and Marco Pavone. RoboMonkey: Scaling Test-Time Sampling and Verification for Vision-Language-Action Models. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 3200–3217, 2025. URL https://proceedings.mlr.press/v305/kwok25a.html.

Jacky Kwok, Xilun Zhang, Mengdi Xu, Yuejiang Liu, Azalia Mirhoseini, Chelsea Finn, and Marco Pavone. Scaling Verification Can Be More Effective than Scaling Policy Learning for Vision-Language-Action Alignment. In *Computer Vision – ECCV 2026*, Lecture Notes in Computer Science, pp. 37–55, 2026. doi: 10.1007/978-3-032-37016-7_3. URL https://arxiv.org/abs/2602.12281.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-Free Predictive Inference for Regression. *Journal of the American Statistical Association*, 113 (523):1094–1111, 2018. doi: 10.1080/01621459.2017.1307116.

Andrew Zou Li, Gokul Swamy, Yonatan Bisk, and Andrea Bajcsy. ELASTIC: Efficiently Learning to Adaptively Scale Test-Time Compute for Generative Control Policies, 2026a. URL https://arxiv.org/abs/2606.31132.

Boning Li, Yu Chen, and Longbo Huang. AV-AIVAT: 74x Cheaper Agent Evaluation with Certified Anytime-Valid Stopping in Imperfect-Information Games, 2026b. URL https://arxiv.org/abs/2608.06362.

Huiqiong Li, Zhiting Mei, Anirudha Majumdar, Jingjing Chen, Yu-Gang Jiang, and Bin Zhu. LIBERO-VPro: Benchmarking Closed-Loop Visual Robustness of Robotic Foundation Models, 2026c. URL https://arxiv.org/abs/2609.24350.

Wenhao Li, Xiu Su, Yichao Cao, Hongyan Xu, Xiaobo Xia, Shan You, Yi Chen, and Chang Xu. VLA-ATTC: Adaptive Test-Time Compute for VLA Models with Relative Action Critic Model, 2026d. URL https://arxiv.org/abs/2605.01194.

Zekai Li, Jiaming Tang, and Zhijian Liu. FlashVLA: Streaming Action Decoding for Fast and Asynchronous VLA Inference, 2026e. URL https://arxiv.org/abs/2608.27384.

Shijie Lian, Bin Yu, Zhaolong Shen, Xiaopeng Lin, Yichao Du, Zhirui Zhang, Laurence T. Yang, and Kai Chen. ActionPiece: Rethinking Action Tokenization for Autoregressive Vision-Language-Action Models, 2026. URL https://arxiv.org/abs/2609.18487.

Andrew Liao, Hanchen Cui, Karthik Desingh, and Aryan Deshwal. Active Real-World Factor-Based Evaluation for Generalist Robot Policies, 2026. URL https://arxiv.org/abs/2607.14439.

Zhanran Lin, Wanteng Ma, Zhimei Ren, and Yuting Wei. Sample-Efficient Multiple Testing with Adaptive Data Collection, 2026. URL https://arxiv.org/abs/2609.26651.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking Knowledge Transfer for Lifelong Robot Learning. In *Advances in Neural Information Processing Systems: Datasets and Benchmarks Track*, volume 36, pp. 44776–44791, 2023. doi: 10.52202/075280-1939. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/8c3c666820ea055a77726d66fc7d447f-Abstract-Datasets_and_Benchmarks.html.

Chaoqi Liu, Xiaoshen Han, Jiawei Gao, Yue Zhao, Haonan Chen, and Yilun Du. OAT: Ordered Action Tokenization. In *Proceedings of Robotics: Science and Systems*, Sydney, Australia, 2026a. URL https://www.roboticsproceedings.org/rss22/p075.html.

Yicheng Liu, Zibin Dong, Baijun Ye, Tianyuan Yuan, Tao Jiang, Anqi Yang, Shicheng Cao, Haonan Liu, Yue Sun, Zihan Guo, et al. G0.5: One Autoregressive Stream for Robot Reasoning and Action, 2026b. URL https://arxiv.org/abs/2608.11739.

Yushan Liu, Peibo Sun, Xintao Chao, Zhenyang Yang, Yifan Xie, Lingfeng Zhang, Shoujie Li, Chenyu Tang, Fang Chen, Xiao-Ping Zhang, and Wenbo Ding. CheckVLA: Execution-Time Verification with Action-Conditioned World Model for Long-Horizon Mobile Manipulation, 2026c. URL https://arxiv.org/abs/2607.26789.

Zeyu Lou, Tianran Zhang, Xinquan Yue, Ya Jing, and Chenyang Si. SafeLoop: Risk-Aware Rollback for Vision-Language-Action Manipulation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2026. URL https://arxiv.org/abs/2609.26313.

Zuxing Lu, Hongjia Zhai, Guanzhi Wang, Huajian Zeng, Jiaqi Yang, Jingyu Liu, Lei Cheng, Yuantai Zhang, Yuheng Qiu, Zezhou Cheng, Ivan Laptev, Danfei Xu, Benjamin Riviere, Giuseppe Loianno, Eric Xing, and Xingxing Zuo. World-Action Models for Robot Learning and Control: A Survey, 2026. URL https://arxiv.org/abs/2609.16074.

Khang Luong, Nam Nguyen, Hoang Ta, Hung The Tran, and Tuan Dam. Variance Driven Exploration: A Provable and Efficient Methodology for Pure Exploration in Highly Stochastic Environments. In *Proceedings of the 43rd International Conference on Machine Learning*, Proceedings of Machine Learning Research, 2026. URL https://arxiv.org/abs/2608.21995.

Arnab Maiti, Yunbei Xu, and Kevin Jamieson. On the Power of Adaptivity for ε-Best Arm Identification in Linear Bandits. In *Proceedings of the 39th Annual Conference on Learning Theory*, 2026. URL https://arxiv.org/abs/2605.15663.

Diego Martinez-Taboada and Aaditya Ramdas. Gaussian-efficient testing by betting on the mean of bounded data, 2026. URL https://arxiv.org/abs/2608.21694.

Andreas Maurer and Massimiliano Pontil. Empirical Bernstein Bounds and Sample Variance Penalization. In *Proceedings of the 22nd Annual Conference on Learning Theory*, 2009. URL https://arxiv.org/abs/0907.3740.

Pierre Ménard. Gradient Ascent for Active Exploration in Bandit Problems, 2019. URL https://arxiv.org/abs/1905.08165.

Mitsuhiko Nakamoto, Oier Mees, Aviral Kumar, and Sergey Levine. Steering Your Generalists: Improving Robotic Foundation Models via Value Guidance. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 4996–5013, 2024. URL https://proceedings.mlr.press/v270/nakamoto25a.html.

Seongheon Park, Wendi Li, Changdae Oh, Samuel Yeh, Zsolt Kira, Michael Hagenow, and Sharon Li. Hide-and-Seek in Trajectories: Discovering Failure Signals for VLA Runtime Monitoring, 2026. URL https://arxiv.org/abs/2605.30834.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. FAST: Efficient Action Tokenization for Vision-Language-Action Models. In *Proceedings of Robotics: Science and Systems*, Los Angeles, California, USA, 2025. doi: 10.15607/RSS.2025.XXI.012. URL https://www.roboticsproceedings.org/rss21/p012.html.

Kevin Regan and Craig Boutilier. Regret-based Reward Elicitation for Markov Decision Processes. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, pp. 444–451, 2009. URL https://www.auai.org/uai2009/proceedings.html.

Ralf Römer, Maximilian Seeliger, Saida Liu, Ben Sturgis, Marco Bagatella, Daniel Marta, Andreas Krause, and Angela P. Schoellig. Uncertainty Quantification for Flow-Based Vision-Language-Action Models, 2026. URL https://arxiv.org/abs/2606.18043.

Daniel Russo. Simple Bayesian algorithms for best-arm identification. *Operations Research*, 68(6): 1625–1647, 2020. doi: 10.1287/opre.2019.1911.

Rohit Kumar Salla, Manoj Saravanan, and Simon Stepputtis. Learning When to Stop: Prefix-Optimal Dynamic Diffusion Policies for Continuous Control, 2026. URL https://arxiv.org/abs/2608.05084.

Yuliy Sannikov and Weijie Zhong. Exploration and Stopping, 2026. URL https://arxiv.org/abs/2608.10274.

Anya Singh, Cabrel Happi, Jai Relan, Varun Nair, and Vidyut Baradwaj. BOKBO (Best of K Bad Options): Calibrated Abstention for VLA Policies, 2026. URL https://arxiv.org/abs/2605.30660.

Achyuthan Sivasankar. Adaptive Compute in Latent World Models: When Depth Helps, Hurts, or Doesn't Matter, 2026. URL https://arxiv.org/abs/2607.10203.

David Snyder, Apurva Badithela, Nikolai Matni, George J. Pappas, Anirudha Majumdar, Masha Itkina, and Haruki Nishimura. Beyond Binary Success: Sample-Efficient and Statistically Rigorous Robot Policy Comparison. In *Robotics: Science and Systems XXII*, 2026. doi: 10.15607/RSS.2026.XXII.076. URL https://www.roboticsproceedings.org/rss22/p076.html.

Zhen Sun, Yongjian Guo, Haoran Sun, Luqiao Wang, Wei Lu, Jiachi Ji, Shengzhe Ji, Junwu Xiong, and Zhijun Meng. Pre-VLA: Preemptive Runtime Verification for Reliable Vision-Language-Action and World-Model Rollouts, 2026. URL https://arxiv.org/abs/2605.22446.

Yanchuan Tang, Taowen Wang, Yuefei Chen, Boxuan Zhang, Qiang Guan, and Ruixiang Tang. Shifting Uncertainty to Critical Moments: Towards Reliable Uncertainty Quantification for VLA Model, 2026. URL https://arxiv.org/abs/2603.18342.

Yang Wan, Xihang Yue, Zhirui Liu, Ziyuan Chu, Shuxun Wang, Yuhan Chen, Xiaonan Jiang, Xukun Zhu, Yubo Dong, and Linchao Zhu. No Free Checker: A Survey of Verifiers for Robot Policies, 2026. URL https://arxiv.org/abs/2609.09250.

Chenyang Wang, Yuntian Wang, Xiaoxiong Yang, Dingde Jiang, Siao Liu, and Yang Liu. Uncertainty-Guided Sparse Refinement for Action Chunking Transformer Policies, 2026a. URL https://arxiv.org/abs/2609.15840.

Linhan Wang, Zijian An, Mingyuan Zhang, Chen Dai, Yi Xu, Can Cui, Zichong Yang, Yinlin Chen, Lifeng Zhou, and Chang-Tien Lu. GlanceWAM: Sparse Test-Time Imagination for World-Action Models, 2026b. URL https://arxiv.org/abs/2608.23927.

Shuo Wang, Hanyuan Xu, Yingdong Hu, Fanqi Lin, and Yang Gao. A Practical Recipe Towards Improving Sim-and-Real Correlation for VLA Evaluation, 2026c. URL https://arxiv.org/abs/2606.10366.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 2024. doi: 10.1093/jrsssb/qkad009.

Skyler Wu, Yash Nair, and Emmanuel J. Candès. Efficient Evaluation of LLM Performance with Statistical Guarantees, 2026. URL https://arxiv.org/abs/2601.20251.

Changhua Xu, En Yu, Junyu Xuan, and Jie Lu. VGAS: Value-Guided Action-Chunk Selection for Few-Shot Vision-Language-Action Adaptation. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence*, pp. 5110–5118, 2026. doi: 10.24963/ijcai.2026/569. URL https://www.ijcai.org/proceedings/2026/569.

Ge Yan, Jinghao Liu, Yuzhi Fan, Lei Cai, Minwen Liao, Jesse Zhang, and Dieter Fox. Flex-π: A Multi-Stream World-Action Model with Compute Flexibility, 2026. URL https://arxiv.org/abs/2608.10860.

Jinghan Yang, Yunchao Zhang, Wang Yuan, Haolun Wan, Jiaming Zhang, Zhengyang Hu, and Yanchao Yang. Tri-Info: Generalizable, Interpretable Failure Prediction for VLA Models via Information Theory, 2026a. URL https://arxiv.org/abs/2606.19998.

Shiyuan Yang, Borong Zhang, Jizheng Zhang, Zhijia Tao, Junfei Guo, Donglai Ran, Xu Bian, and Qingbiao Li. FabriVLA: A Lightweight Vision-Language-Action Model with Conformal Action Chunk Uncertainty, 2026b. URL https://arxiv.org/abs/2607.08575.

Siyuan Yang, Yang Zhang, Haoran He, Ling Pan, Xiu Li, Chenjia Bai, and Xuelong Li. Steering Vision-Language-Action Models as Anti-Exploration: A Test-Time Scaling Approach, 2025. URL https://arxiv.org/abs/2512.02834.

Jiarui Yao, Jiaxi Zhao, and Xiangxin Zhou. Gap Entropy and Almost Instance-Wise Optimal Best-Arm Identification, 2026. URL https://arxiv.org/abs/2609.13703.

Jessie Yuan, Yilin Wu, and Andrea Bajcsy. When to Act, Ask, or Learn: Uncertainty-Aware Policy Steering. In *Robotics: Science and Systems XXII*, 2026a. URL https://arxiv.org/abs/2602.22474.

Jianhao Yuan, Yu Yuan, Benjamin Ramtoula, Lukas Vierling, Paul Newman, Lars Kunze, Philip Torr, and Daniele De Martini. Beyond Visual Quality: A Study of Test-Time Planning with World Action Models, 2026b. URL https://arxiv.org/abs/2609.24745.

Haoran Zhang, Yifu Lu, Boyang Wang, Xuhui Kang, Yen-Ling Kuo, Zezhou Cheng, Mengdi Wang, and Odest Chadwicke Jenkins. Foresight: Failure Detection for Long-Horizon Robotic Manipulation with Action-Conditioned World Model Latents, 2026. URL https://arxiv.org/abs/2606.23085.

Thomas T. Zhang, Daniel Pfrommer, Chaoyi Pan, Nikolai Matni, and Max Simchowitz. Action Chunking and Exploratory Data Collection Yield Exponential Improvements in Behavior Cloning for Continuous Control, 2025. URL https://arxiv.org/abs/2507.09061.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, 2023. URL https://www.roboticsproceedings.org/rss19/p016.html.

Yuchi Zhao, Miroslav Bogdanovic, Arjun Sohal, Liyu Tao, Kourosh Darvish, Alán Aspuru-Guzik, Florian Shkurti, and Animesh Garg. Dynamic Execution Horizon Prediction for Chunk-based Robot Policies, 2026. URL https://arxiv.org/abs/2606.11408.

Yixin Zheng, Jiangran Lyu, Yuntian Deng, Kai Liu, Yizhou Zhou, Yizhou Wang, Xiaoguang Zhao, He Wang, and Zhizheng Zhang. DualWAM: Dual-System World Action Models for Asynchronous Global Planning and Local Refinement, 2026. URL https://arxiv.org/abs/2609.24868.

Kaiwen Zhu, Dongfang Liu, and Liangkai Liu. VLA-Scope: Shift-Aware Failure Prediction for Vision-Language-Action Models, 2026. URL https://arxiv.org/abs/2609.21246.

Vilém Zouhar, Julia Kreutzer, Alon Lavie, Tom Kocmi, Matt Post, Ondřej Bojar, and Mrinmaya Sachan. Dynamically Allocating Evaluation Effort for Model Ranking. In *Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing*, 2026. URL https://arxiv.org/abs/2608.03437.

A PROOFS

This appendix proves the results the main text uses, in the order the main text uses them. Each result is stated before its proof, and main-body statements keep their numbers.

A.1 PROOFS OF PAYLOAD AND CALL ACCOUNTING

Proposition 2 (Serialized payload)**.** *The serialized k-row prefix of a chunk with m action dimensions, $k \in \{1, 2, 4, 8\}$, carries $b(k)$ bits as in* (2) *with $\beta_h = 400$, $\beta_s = 88$, $\beta_r = 16$ and $\beta_c = 8$ whenever every quantized coefficient q satisfies $-64 \le q \le 63$. The leading d steps coded as their own horizon-d chunk also carry $b(d)$ bits.*

Proof. The stream opens with a header of a four-byte magic string, a version byte, a level count, two 16-bit dimensions, an eight-byte scale and a 32-byte candidate binding, 50 bytes in total. Geometric bands end at $1, 2, 4, \ldots$, so the first $k \in \{1, 2, 4, 8\}$ rows occupy $1 + \log_2 k$ bands, and each band opens with an 11-byte segment record. Each row carries a two-byte frequency index and its m coefficients. A zigzag varint maps q to $2q$ for $q \ge 0$ and to $-2q - 1$ for $q < 0$. It writes one byte exactly when $-64 \le q \le 63$. Summing gives $8\,[50 + 11(1 + \log_2 k) + (2 + m)k]$ bits, which is (2) with $\beta_h = 400$, $\beta_s = 88$, $\beta_r = 16$ and $\beta_c = 8$. None of these fields depends on the coefficient values beyond the one-byte condition. The leading d steps encoded with horizon d produce the bands ending at $1, 2, \ldots, d$ and d rows of m coefficients, the same field counts as a d-row prefix of the horizon-L code, so they cost $b(d)$. □

Proposition 3 (Call and bit counts)**.** *A controller that executes d steps of each arrived chunk over H actuated steps issues $\lceil H/d \rceil$ policy calls and receives $\lceil H/d \rceil\, b(d)$ bits. For depths $d < L$ that both divide H, the call and bit ratios are those of* (3)*, and the bit ratio is at most one exactly when $b(d)/d \le b(L)/L$.*

Proof. The controller executes the arrived steps before requesting the next chunk, so the arrivals partition the H actuated steps into blocks of d steps with a final partial block, giving $\lceil H/d \rceil$ arrivals and one policy call each. Every arrival carries $b(d)$ bits, which yields the episode total $\lceil H/d \rceil\, b(d)$ and establishes the two counts.

For the comparison, let $d < L$ both divide H. The call counts are H/d and H/L, whose ratio is L/d, and the bit totals are $Hb(d)/d$ and $Hb(L)/L$. Dividing the first by the second gives the bit ratio $(L/d)\, b(d)/b(L)$, and this ratio is at most one exactly when $b(d)/d \le b(L)/L$, with value one at equality, which is the stated criterion. □

A.2 PROOFS OF THE FREQUENCY-PREFIX RESULTS

Lemma 1 (Energy of a frequency prefix)**.** *For every chunk, $\|X - \hat{X}_k\|_F^2 = \sum_{r>k} \|Z_r\|^2$ before quantization, and $\mathbb{E}\|X - \hat{X}_k\|_F^2 = \sum_{r>k} e_r$. Among all rank-$k$ orthogonal projections Q along time, $\mathbb{E}\|X - QX\|_F^2 \ge \sum_{r>k} \lambda_r$, where $\lambda_1 \ge \cdots \ge \lambda_L$ are the eigenvalues of Σ, with equality for the Karhunen–Loève projection onto the leading eigenvectors. If the row energies are non-increasing, $e_1 \ge e_2 \ge \cdots \ge e_L$, the k-row prefix has the smallest expected residual among all choices of k cosine rows.*

Proof. Write $X = C^\top Z$ with C orthogonal. Then $\hat{X}_k = C^\top P_k Z$ and $X - \hat{X}_k = C^\top (I - P_k) Z$, so orthogonality of C gives $\|X - \hat{X}_k\|_F^2 = \|(I - P_k)Z\|_F^2 = \sum_{r>k} \|Z_r\|^2$, which is Parseval's identity for the discarded rows. Since $Z_r = c_r^\top X$, taking expectations gives $\mathbb{E}\|Z_r\|^2 = c_r^\top \Sigma c_r = e_r$ and the second identity. For a rank-k orthogonal projection Q along time, $\mathbb{E}\|X - QX\|_F^2 = \operatorname{tr}\Sigma - \operatorname{tr}(Q\Sigma)$, and the Ky Fan maximum principle gives $\operatorname{tr}(Q\Sigma) \le \sum_{r \le k} \lambda_r$, with equality for the projection onto the leading k eigenvectors. Any choice S of k cosine rows has expected residual $\sum_{r \notin S} e_r$, which non-increasing energies minimize at $S = \{1, \ldots, k\}$. □

Proposition 1 (Deviation tube of every-chunk truncation)**.** *Suppose the full-code closed loop $G(x) = F(x, \bar{u}(x))$ contracts in a norm $\|\cdot\|_P$ on a common invariant state domain, $\|G(x) - G(y)\|_P \le \rho \|x - y\|_P$ with $\rho < 1$, and $\|F(x, u) - F(x, u')\|_P \le \ell \|u - u'\|$. Started from the same*

state, the truncated and full-code trajectories satisfy $d_{t+1} \le \rho\, d_t + \ell\, \varepsilon_t$ *for* $d_t = \|x_t - \bar{x}_t\|_P$ *with* $\bar{x}_t$ *the full-code trajectory, hence*

$$\|x_t - \bar{x}_t\|_P \le \ell \sum_{s<t} \rho^{\,t-1-s} \varepsilon_s \le \frac{\ell\, \sup_s \varepsilon_s}{1-\rho}. \tag{4}$$

Proof. Write $x_{t+1} = F(x_t, u_k(x_t))$ and $\bar{x}_{t+1} = G(\bar{x}_t)$ with the same exogenous randomness. Then

$$d_{t+1} \le \|F(x_t, u_k(x_t)) - F(x_t, \bar{u}(x_t))\|_P + \|G(x_t) - G(\bar{x}_t)\|_P \le \ell\, \varepsilon_t + \rho\, d_t.$$

With $d_0 = 0$, induction gives $d_t \le \ell \sum_{s<t} \rho^{t-1-s} \varepsilon_s$, and bounding each ε_s by its supremum and summing the geometric series gives $\ell \sup_s \varepsilon_s/(1-\rho)$. This is the linear-gain case of discrete-time input-to-state stability (Jiang & Wang, 2001), with the residual as the input. □

A.3 Proofs of the certificate results

Theorem 1 (Exact certificate)**.** *Let* $\mathcal{A}$ *be finite, let the graph on* $\mathcal{A} \cup \{z\}$ *have real weights and a path between every ordered pair of vertices, and let* $F(w)$ *be nonempty. Then for every* $a \in \mathcal{A}$ *and* $b \in \mathcal{A} \cup \{z\}$*,*

$$\sup_{m \in F(w)} (m_a - m_b) = d(b,a), \qquad \sup_{m \in F(w)} \Big(m_a - \min_{b \in \mathcal{A}} m_b \Big) = R_w(a), \tag{6}$$

and the potential $m_c = d(b^\star, c) - d(b^\star, z)$ *with* $b^\star \in \arg\max_{v \in \mathcal{A}} d(v,a)$ *lies in* $F(w)$ *and attains the regret supremum.*

Proof. Fix an edge field w with a nonempty feasible set and all pairs reachable. Summing the edge inequalities along a walk $b = v_0 \to \cdots \to v_r = a$ gives $m_a - m_b \le \sum_t w(v_t, v_{t+1})$. Minimizing over walks yields $m_a - m_b \le d(b,a)$. To attain this bound, set $m_c = d(b,c) - d(b,z)$, where z is the reference vertex. The triangle inequality gives $m_v - m_u = d(b,v) - d(b,u) \le w(u,v)$, so m is feasible and $m_z = 0$. Feasibility excludes negative cycles and gives $d(b,b) = 0$, hence $m_a - m_b = d(b,a)$. For any feasible assignment, regret is bounded by $\max_{b \in \mathcal{A}} d(b,a)$. Choosing a maximizing rival $b \in \mathcal{A}$ in the construction attains that bound. This proves both identities. □

Proposition 4 (State-marginal validity)**.** *Let each sampled edge hold with probability at least* $1 - \delta_e$*, and let the depth edges of a state drawn exchangeably with the calibration states hold with probability at least* $1 - \delta_q$*. Then*

$$\Pr\big(\textsc{PairGap}\ \textit{certifies}\ a\ \textit{and}\ r(a) > \epsilon\big) \le \delta_q + \textstyle\sum_e \delta_e. \tag{7}$$

Proof. Let G be the event that every sampled edge and every depth edge of the state holds. Reference edges and trivial edges hold deterministically. On G the true cost vector lies in $F(w)$, so Theorem 1 gives $r(a) \le R_w(a)$, and a certificate $R_w(a) \le \epsilon$ implies $r(a) \le \epsilon$. The event in (7) is therefore contained in the complement of G, whose probability is at most $\delta_q + \sum_e \delta_e$ by the union bound. Each sampled edge uses a preallocated error at a budget n fixed before any rollout, and the executed option is fixed in advance, so no edge is chosen after its outcomes are seen. A receiver that tests several budgets in sequence divides each edge's error over the budgets it tests. □

Proposition 5 (Composition)**.** *Consider the binary-cost graph with* $|\mathcal{A}| \ge 2$ *and* $F(w)$ *nonempty, where every option has a zero-weight edge into* z*, every option-to-option edge has weight at most* C*, and every edge out of* z *has weight* C *except the sampled absolute edge* $z \to a$ *of weight* $U_a \in [0, C]$*. Let* $d_{\neg z}(b,a)$ *be the shortest distance over paths that avoid* z*, the* relational *distance, let* $R^-(a)$ *be the regret certificate of the graph retaining the default, reference and absolute edges, and let* $R^+(a)$ *be that of the graph with the paired and depth edges added. Then: (i) the default edges* $z \to v$ *for* $v \ne a$ *are redundant for option-to-option shortest distances, and* $R^+(a) = \max_{b \in \mathcal{A}} \min\{d(b,z) + U_a,\ d_{\neg z}(b,a)\}$ *with* $d(b,z) \le 0$*, and* $R^-(a) = U_a$*; (ii) at equal edge weights,* $R^+(a) \le R^-(a)$*, and when* $U_a > \epsilon$*, relational evidence certifies a new option,* $R^+(a) \le \epsilon < R^-(a)$*, exactly when* $\min\{d(b,z) + U_a,\ d_{\neg z}(b,a)\} \le \epsilon$ *for every source* $b \in \mathcal{A}$*; (iii) for an unreceived full option,* $d_{\neg z}((i,D),a) \le q + d_{\neg z}((i,k),a)$*, so the unreceived full code of candidate* i *lies at most* q *farther from* a *than its received prefix; (iv) if* a *has zero failures and* a *and a received rival* b *have no*

discordant seed among n shared seeds, the signed edge (13) *equals $u(0, n, \delta_p/2)$, which is at least the absolute weight $u(0, n, \delta_a)$ whenever $\delta_p \leq 2\delta_a$, so when b reaches a through its paired edge alone, its relational path is no shorter than the reference path.*

Proof. (i) A path fragment $u \to z \to v$ with $u, v \in \mathcal{A}$ and $v \neq a$ has weight C. The direct edge $u \to v$ has weight at most C, so replacing the fragment preserves or reduces the path length. Repeated replacement removes every use of a default edge out of z. Feasibility excludes negative cycles, so a shortest path into z can end at its first visit to z. Deleting the default edges $z \to v$, $v \neq a$, therefore preserves option-to-option distances and distances into z. The reduced graph has one outgoing reference edge, $z \to a$. A path from b to a in this graph that visits z reaches a through $z \to a$ after its last visit to z, and its length is at least $d(b, z) + U_a$, attained by a shortest path into z followed by $z \to a$. Every other path avoids z and has length at least $d_{\neg z}(b, a)$. Taking the minimum of the two classes and the maximum over sources gives the formula. The reference edges $b \to z$ have weight 0, so $d(b, z) \leq 0$. In the absolute graph, $d(b, z) = 0$ and $d_{\neg z}(b, a) = C \geq U_a$ for $b \neq a$; $d_{\neg z}(a, a) = 0$. There is at least one rival and $U_a \geq 0$, so $R^-(a) = U_a$. *(ii)* Adding edges never lengthens a shortest path, which gives the inequality. With $U_a > \epsilon$, $R^-(a) > \epsilon$, and $R^+(a) \leq \epsilon$ holds exactly when each term of the maximum in (i) is at most ϵ. *(iii)* The path $(i, D) \to (i, k)$ followed by a shortest z-avoiding path from (i, k) to a avoids z and has length $q + d_{\neg z}((i, k), a)$. *(iv)* Zero failures of a and zero discordances give $n_+ = n_- = 0$ and $U_a = u(0, n, \delta_a)$. Equation (13) reads $u(0, n, \delta_p/2) - \ell(0, n, \delta_p/2) = u(0, n, \delta_p/2) = 1 - (\delta_p/2)^{1/n}$, and $u(0, n, \delta_a) = 1 - \delta_a^{1/n}$ is decreasing in the error, so $\delta_p/2 \leq \delta_a$ gives the claim. When b reaches a through its paired edge alone, $d_{\neg z}(b, a) = w(b, a) \geq U_a \geq d(b, z) + U_a$. □

B FIXED-ROW EXECUTION AND CERTIFICATE PROTOCOLS

The every-chunk experiments fix k before the episode, decode all L steps from those rows, execute the chunk and repeat at the next state. Algorithm 1 gives the separately evaluated on-demand receipt interface at one restored state. The interface tests the pre-registered first candidate at two depths.

Algorithm 1 Two-depth on-demand receipt at one state

Require: fixed candidates, prefix depth k, full depth D, fixed budget n, report tolerance ϵ
1: fix candidate 1 before sampling; receive all K prefixes
2: set receipt threshold $\eta = 0.05$ and online error $\delta_o = 0.025$
3: estimate the prefix failure upper bound U_k from n rollouts at error $\delta_o/2$
4: $a \leftarrow (1, k)$; $R \leftarrow U_k$
5: **if** $U_k > \eta$ **then**
6: receive candidate 1's remaining rows; estimate U_D from n full-depth rollouts at error $\delta_o/2$
7: **if** $U_D < U_k$ **then**
8: $a \leftarrow (1, D)$; $R \leftarrow U_D$
9: **end if**
10: **end if**
11: **return** recommendation a, bound R, certification status $R \leq \epsilon$, bits and rollout count

The replay reports $\epsilon \in \{0.05, 0.10\}$ with the same receipt threshold η. It selects the prefix on a tie and retains an uncertified recommendation for the held-out audit. At $n = 48$, every state receives candidate 1's full code before the final depth choice. The two absolute edges share δ_o, and the panel reserves the other 0.025 of its total state error for depth calibration; this receipt arm uses absolute edges. Each budget is evaluated as a fixed-budget experiment. Repeated budget tests require an error allocation across those tests.

Receipt cost and zero-failure sample requirements. Both receipt arms sample the pre-fixed first candidate. Full receipt obtains all four full codes and samples that candidate's full depth. The two-depth rule receives four prefixes and completes the first candidate when its prefix bound exceeds $\eta = 0.05$. Its 3,008-bit cost is $4 \times 560 + (1{,}328 - 560)$; full receipt costs $4 \times 1{,}328 = 5{,}312$ bits. The six prefix-depth certificates identify the final execution depth after full-code receipt. No evaluated state stops receipt at prefix depth.

For a binary cost with zero observed failures, the one-sided exact upper bound at edge error α is $U(n) = 1 - \alpha^{1/n}$. The corresponding sample requirement for $U(n) \leq \epsilon$ is

$$n \geq \left\lceil \frac{\log \alpha}{\log(1-\epsilon)} \right\rceil. \tag{8}$$

With $\alpha = 0.0125$ for each on-demand depth test, a bound of 0.05 requires 86 zero-failure rollouts. This exceeds the evaluated maximum of 64 and explains why the receipt gate remains open. At $\epsilon = 0.10$, the zero-failure requirement is 42, covered by the next evaluated budget of 48. The first-certifying budgets of Table 2 are read from separate fixed-budget evaluations. Sequentially testing those budgets would require an additional allocation of the error probability across tests.

The whole-episode experiments use fixed-row execution with no simulator check at policy calls. The two-depth interface is evaluated by replay at individual states. Its integration into every-chunk execution and a deployed controller's recovery action after an uncertified recommendation remain unevaluated.

Exploratory state-level sign-flip diagnostic. The LIBERO-10 diagnostic pools four candidates and 64 shared-seed continuations per state and depth. Its 40 success differences, shifted by two percentage points, are all positive; their minimum is 1.21875 points. The all-positive assignment therefore uniquely maximizes the signed sum, giving $P = 2^{-40}$. The conditional reference distribution assumes coordinatewise sign invariance across states at the null boundary. Four states share each task, and the implemented diagnostic treats all 40 states as separate sign units. The panel design was frozen under version control before the recorded continuations; this state-level test entered the analysis afterward. The main text reports the observed failure counts and state-level changes.

C Registered every-chunk frequency-truncation design

The every-chunk comparison on the official protocol was registered before any of its episodes ran. It uses the OpenVLA-OFT checkpoints, revisions and seed of Section 5.2 on LIBERO Spatial and Goal, ten tasks per suite and fifty official initial states per task, which gives 500 episodes per suite and arm, paired across arms on the suite, task, initial-state index and seed. Five arms enter the table. *Native full* and *time-4* are two of the arms of Figure 4. *Quantized full* passes every chunk through the orthonormal temporal cosine transform and the scalar quantizer of the code with all eight rows kept, and *freq-2* and *freq-4* keep the first two and four rows of the same code with the remaining rows zero; all three execute eight steps per call. The table reports policy calls, measured payload bits per call and success. The primary contrasts are freq-2 and freq-4 against quantized full, which isolates coefficient removal from quantization, and quantized full against native full. A frequency arm is non-inferior on a suite when a one-sided exact sign-flip permutation test over the ten task-level paired differences rejects, at level 0.025, the hypothesis that the mean difference is at most -2 points; episode-level paired wins and losses are reported beside it. Freq-2 is the reported operating point unless its Goal success falls more than two points below quantized full, in which case freq-4 is. On Spatial, quantized full reaches 98.6% and freq-2 98.8%, and both contrasts reject at $P = 0.0039$ and $P = 0.0117$; freq-4 rejects against quantized full at $P = 0.0020$ (Section 5.2).

Completed Goal comparison. The registered operating-point rule above remains fixed. Goal success is 97.6% for quantized full, 96.6% for freq-2 and 96.8% for freq-4. The freq-2 change is -1.0 points, so the decline stays within two points and the rule retains freq-2. The non-inferiority tests give $P = 0.1406$ and $P = 0.0742$. Both exceed the registered level 0.025, leaving non-inferiority unestablished for both frequency arms on Goal. The operating-point rule selects a row count from the observed success change; the test supplies the separate statistical assessment.

Registration identity and inference. The protocol was frozen under version control before the first frequency-arm episode started, and the anonymized artifact records both events. For each of the ten tasks, let Δ_j be the paired success-rate difference in percentage points and let $q_j = \Delta_j + 2$. The analyzer computes $2^{-10} \sum_{s \in \{-1,1\}^{10}} \mathbf{1}\{\sum_j s_j q_j \geq \sum_j q_j\}$. This enumeration is exact conditional on the observed magnitudes. A null-boundary interpretation requires joint invariance under these sign changes; independent, symmetric task-level differences are a sufficient condition. The ten official tasks define the evaluated suite, and the single policy-evaluation seed fixes this study's replication scope.

D EXTENDED RELATED WORK

Vision-language-action policies and chunked execution. Action chunking with transformers and flow-based generalist policies emit several control steps per call (Zhao et al., 2023; Black et al., 2025a), and recent systems stream reasoning and action in one sequence or survey world-action models that predict both (Liu et al., 2026b; Lu et al., 2026). Uncertainty-guided sparse refinement revisits only parts of an action chunk (Wang et al., 2026a). Our analysis applies to every such policy whose chunk passes through an ordered code before execution.

Early exit and adaptive test-time compute. Adaptive-WAM exits planning early from intermediate video-diffusion states, prefix-optimal diffusion policies learn when to stop denoising, and ELASTIC learns to scale test-time compute (Ang et al., 2026; Salla et al., 2026; Li et al., 2026a). Glance-WAM, DualWAM and latent world models vary imagination and depth at test time (Wang et al., 2026b; Zheng et al., 2026; Sivasankar, 2026), and VLA-ATTC adapts compute with a relative action critic (Li et al., 2026d). These methods decide how much generation to spend. The frequency and time axes of this paper decide how much of a generated chunk to send and execute, and the two decisions compose.

Selection, verification and failure prediction. Test-time sampling and verification rank sampled chunks with learned scores (Kwok et al., 2025; Dai et al., 2025; Xu et al., 2026; Kwok et al., 2026), value guidance and anti-exploration steer generalist policies (Nakamoto et al., 2024; Yang et al., 2025; Jang et al., 2025), and reasoning and hierarchical planners search imagined futures (Guo et al., 2026; Cai et al., 2026; Yuan et al., 2026b; Jeong & Yoon, 2026). Calibrated abstention, conformal chunks and uncertainty estimates flag unreliable actions (Singh et al., 2026; Yang et al., 2026b; Römer et al., 2026; Yuan et al., 2026a), and critical-moment uncertainty, uncertainty-guided recovery and conformal planning extend them (Tang et al., 2026; Chen et al., 2026b; Gonzales et al., 2025). Runtime monitors predict failure from trajectories, information measures and world models (Sun et al., 2026; Park et al., 2026; Yang et al., 2026a; Zhang et al., 2026), and execution-time verification, rollback and self-evaluation act on those predictions (Liu et al., 2026c; Lou et al., 2026; Aniri et al., 2026; Zhu et al., 2026); a recent survey organizes these verifiers (Wan et al., 2026). The per-state check of Section 4 uses restored simulator rollouts as evidence and bounds regret over every candidate and depth.

Evaluation under a budget. Robustness benchmarks and sim-to-real recipes define the populations on which policies are compared (Fei et al., 2025; Li et al., 2026c; Wang et al., 2026c; Community et al., 2026), and active factor-based, sequential and dynamically allocated evaluations spend rollouts where they change a ranking (Liao et al., 2026; Hsu & Shekhar, 2026; Zouhar et al., 2026; Wu et al., 2026). Best-arm identification, fixed-budget and linear variants, and exploration with stopping supply the sampling rules we replay (Balagopalan et al., 2026; Maiti et al., 2026; Yao et al., 2026; Sannikov & Zhong, 2026), common random numbers and ranking-and-selection procedures justify paired seeds (Kim & Nelson, 2006), and empirical-Bernstein, betting and confidence-sequence bounds give the absolute edges (Maurer & Pontil, 2009; Waudby-Smith & Ramdas, 2024; Martinez-Taboada & Ramdas, 2026; Kilian et al., 2026), with adaptive multiple testing controlling many edges at once (Lin et al., 2026).

E ACQUISITION RULE AND EXECUTABLE SPECTRAL PREFIXES

An execution option $a = (i, k)$ identifies candidate i at token depth k. The set $\mathcal{A}$ contains all options, E contains receiver-executable options, S contains simulator-available options, m_a is task cost, $c_a = \lambda_b B_a$ is the remaining communication cost, and $\mu_a = m_a + c_a$ is total cost. An edge $b \to a$ bounds $\mu_a - \mu_b$; its shortest-path closure gives $R_w(a) = \max_b d(b, a)$. The reference vertex z has cost zero. Simulator seeds fix the continuation randomness shared by a pair.

The graph contains unreceived depths throughout acquisition. A token arrival changes executability. Sampling changes evidence. Shared tokens are charged once. The graph implementation uses exact rational arithmetic and retains a realizing path for each distance. The interval construction owns the validity event; the graph closure propagates that event to the returned decision.

E.1 Executable spectral prefixes

The evaluated representation applies an orthonormal temporal transform W, scalar quantization and descending-energy row transmission. Receiving row set I reconstructs a_I, and $\bar{a}$ denotes the decode of every row:

$$a_I = \Pi_{\mathcal{U}}\big(W^\top \widetilde{z}_I/s\big)\,, \qquad \|a_I - \bar{a}\|_2 \le \left(\sum_{r\notin I} \|\widehat{z}_r/s\|_2^2\right)^{1/2}.$$

Zero filling defines every partial decode. Parseval gives the coefficient-space identity, and projection onto the action box is nonexpansive. Exchanging a selected lower-energy row for an unselected higher-energy row decreases residual energy. One descending order therefore minimizes this bound at every depth. Packets identify the fixed candidate, decoder, quantization scale and coefficient rows. Reported bit counts include this metadata. Figure 7 shows one candidate through this path.

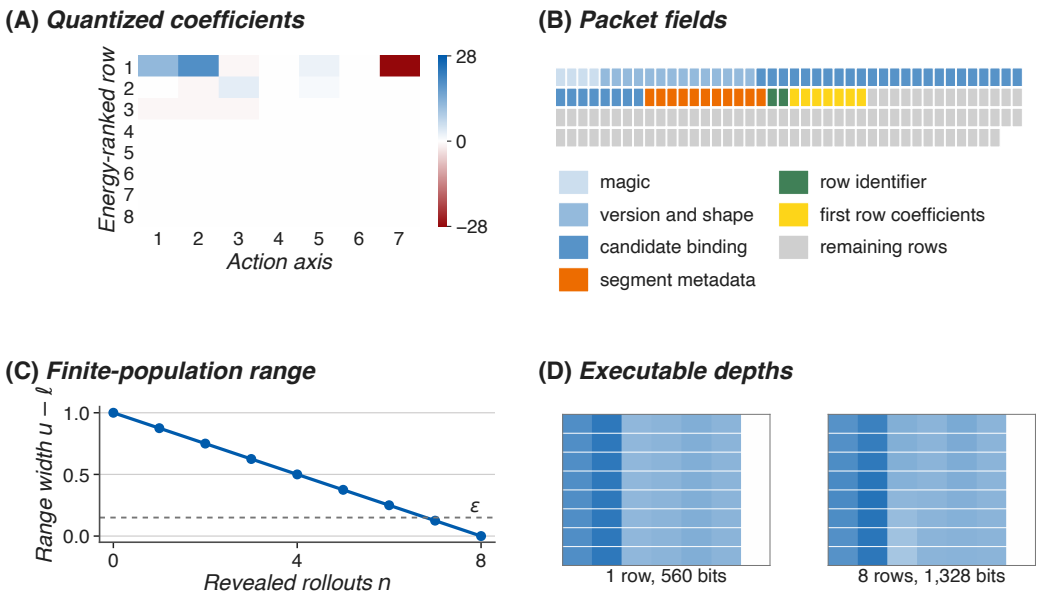

Figure 7: From coefficients to executable depths. (A) Quantized coefficients. (B) Packet fields of the 166-byte record and its 70-byte prefix. (C) Finite-population range. (D) Executable depths.

E.2 Evaluated allocation rules

Proposition 6 (Finite-population range)**.** *Let fixed outcomes $Y_{a,1},\dots,Y_{a,N} \in [L,U]$ be revealed on the index set I_a, with $n_a = |I_a|$ and $T_a = \sum_{j\in I_a} Y_{a,j}$. Under any adaptive allocation,*

$$\ell_a = \frac{T_a + (N-n_a)L}{N}, \qquad u_a = \frac{T_a + (N-n_a)U}{N}, \qquad m_a \in [\ell_a, u_a], \tag{9}$$

and revealing all N outcomes makes both endpoints equal to m_a.

Proof. Each unrevealed summand lies in $[L,U]$, giving pathwise validity under adaptive allocation. Revealing all N outcomes makes both endpoints equal to m_a. Known communication costs translate both endpoints. Taking a minimum of candidate values preserves the corresponding endpoint minima. □

The finite-population version applies Proposition 6. It begins with prefix support $[0, C]$ and full-action support $[c_i, C + c_i]$. It samples the executable option with smallest upper endpoint until

exhaustion, then resolves the unfinished competitor with smallest lower endpoint. Ties favor the prefix and lower candidate index. Tree refinement propagates prefix and full ranges through a two-level acquisition tree and assigns $\epsilon/2$ each to candidate selection and executable-depth selection. Uniform allocation advances every unfinished population under the same tree tests. A full-tolerance incumbent–rival comparator samples the incumbent and the competitor limiting its endpoint certificate. All rules share candidate actions, cost populations, token prices and reveal order.

F PER-STATE SUFFICIENCY CHECK

The per-state check certifies a prefix at one state. It reads the worst-case regret over every candidate and depth, received or unreceived, as a shortest-path distance in one graph of absolute, paired and depth evidence, and the evaluated on-demand rule can request the pre-registered candidate's full code after an uncertified prefix test. Section 4 summarizes the construction; this appendix gives the setup, the rule, the guarantees and the full experiments.

F.1 OPTIONS, COSTS AND THE REGRET OF A PREFIX

At state x, a fixed policy proposes K candidate action chunks. Candidate i is coded by D cosine rows in a fixed transmitted order, and its first k rows decode by zero filling and the inverse transform into an executable motion. An *execution option* is a pair $a = (i, k) \in \mathcal{A}$ with transmission cost B_a bits. The evaluated LaRA-VLA bank orders rows by quantized energy, and every one-row prefix is the DC row. The received set $E \subseteq \mathcal{A}$ contains the options whose rows have arrived. A simulator restores x and runs any received option from a seed j, returning binary task failure $Y_{a,j} \in \{0, 1\}$. The single-state certificate study uses $C = 1$ and bit price $\lambda_b = 0$, with bits reported separately. Option a has expected cost $m_a = \mathbb{E}_j Y_{a,j}$, and executing a incurs regret

$$r(a) = m_a - \min_{b \in \mathcal{A}} m_b. \tag{10}$$

The minimum ranges over every candidate at every depth, received or unreceived. An ϵ certificate for a is a bound $r(a) \le \epsilon$ that holds for every cost vector consistent with the evidence, so it covers the full codes in flight.

F.2 THE CERTIFICATION GRAPH AND THE FOUR EDGE KINDS

Each piece of evidence bounds a cost difference. A directed edge $b \to a$ with weight $w(b, a)$ records

$$m_a - m_b \le w(b, a), \tag{11}$$

on the vertex set $\mathcal{A} \cup \{z\}$, where the reference vertex z has $m_z = 0$. The cost vectors consistent with the evidence form the difference-constraint set $F(w) = \{m : m_z = 0,\ m_a - m_b \le w(b, a) \text{ for every edge}\}$, and $d(b, a)$ denotes the shortest-path distance. Summing (11) along a path bounds $m_a - m_b$ by the path length, so the worst-case regret is read from the graph as

$$R_w(a) = \max_{b \in \mathcal{A}} d(b, a), \qquad a \text{ carries an } \epsilon \text{ certificate } \iff R_w(a) \le \epsilon. \tag{12}$$

Four kinds of edges enter the graph. *Reference edges* $b \to z$ of weight 0 encode $m_b \ge 0$ and need no data. The *absolute edge* $z \to a$ carries an upper confidence bound U_a on m_a from n samples of the executed option, a Clopper–Pearson bound (Clopper & Pearson, 1934) for binary costs. A *signed paired edge* $b \to a$ between received options runs both from the same n seeds, and with n_+ seeds on which only a fails and n_- on which only b fails it carries

$$w(b, a) = u(n_+, n, \delta_p/2) - \ell(n_-, n, \delta_p/2), \tag{13}$$

where δ_p is the paired edge's error budget and u and ℓ are the upper and lower Clopper–Pearson limits at error $\delta_p/2$. Common seeds cancel the shared noise. A *depth edge* $(i, D) \to (i, k)$ runs from each unreceived full option into its received prefix with weight q, intended to bound $m_{(i,k)} - m_{(i,D)}$. The matched LaRA-VLA panel computes q by applying the Clopper–Pearson formula to pooled counts from the other states of the event that the prefix fails and the full code succeeds. This pooled calculation is a calibration diagnostic, and its state-marginal coverage is unestablished. Appendix F.3 states the score-coverage and exchangeability conditions for a split-conformal depth bound (Lei

Algorithm 2 PAIRGAP at one arrival

1: receive the depth-k prefixes of all K candidates; add reference and calibrated depth edges
2: fix the pre-registered option $a = (1, k)$, the budget n, and split δ_o over the sampled edges
3: add the absolute edge $z \to a$ and the signed paired edges $(i, k) \to a$ from n same-seed rollouts
4: compute shortest paths on $\mathcal{A} \cup \{z\}$ and $R_w(a) = \max_b d(b, a)$
5: **return** a, bound $R_w(a)$, certification status $R_w(a) \le \epsilon$, witness paths and bits

et al., 2018). Every other ordered pair carries the trivial weight C, including $z \to b$ for each option without a sampled absolute edge. Figure 3B draws the graph for four candidates at depths one and eight.

Algorithm 2 fixes the executed candidate before any rollout, so the certificate tests a pre-registered option and the error budget is spent only on the edges it samples. The simulator runs received options only, so the unreceived full options (i, D) reach a through two routes, the reference path $(i, D) \to z \to a$ of length U_a, and the relational path through the depth edge and the paired edge of the same candidate, of length $q + w((i, k), a)$. A received rival reaches a through the reference path or its paired edge. The certificate therefore needs no rollout of an unreceived option. The unified arm evaluates the received prefix and keeps its recommended depth fixed. Uncertified recommendations remain in the held-out audit. The shortest paths come from Floyd–Warshall on $2K + 1$ vertices, $O(K^3)$ per update, negligible next to one rollout.

Arms. All arms share the candidate bank, the calibration states, the seeds, the bit accounting and the total error $\delta = \delta_q + \delta_o$. *Absolute* keeps the reference and absolute edges, the minimal depth certifier. *Paired and depth* keeps the relational edges and drops the absolute edge. *Unified* is PAIRGAP with all four edge kinds, splitting δ_o over its K sampled edges. *On-demand receipt* tests the prefix through its absolute edge at error $\delta_o/2$ and, when that bound exceeds the fixed receipt threshold $\eta = 0.05$, receives the incumbent's full code and tests it at the other $\delta_o/2$. It recommends the option with the smaller bound, choosing the prefix on a tie, and reports certification at $\epsilon \in \{0.05, 0.10\}$. Algorithm 1 states this two-depth interface. *Receive, execute full* receives every full code and certifies the first candidate's full option through its absolute edge, the rule a receiver follows when it always waits for complete chunks.

F.3 EXACTNESS, COMPOSITION AND VALIDITY

The certificate is exact given the edges, a composition rule states which witness path decides, and a union bound over the construction controls the event of certifying a bad option.

Appendix A states and proves Theorem 1, Proposition 4 and Proposition 5. Shortest-path closure of difference constraints (Bellman, 1958) and regret over a polytope of feasible values (Regan & Boutilier, 2009) are classical. The statement here adds the reference vertex and options that the receiver has not received, so a certificate at depth k quantifies over every depth. The maximizing source $b^\star$ and its shortest path form the *witness path* (Figure 3B), the evidence that limits the certificate.

Part (ii) of Proposition 5 is the exact new-certification condition. Relational evidence matters exactly on the sources whose reference path exceeds ϵ, and every such source needs a relational path within ϵ. Part (iv) describes pairs whose samples all succeed, with the specified allocation of edge errors. On the evaluated panel every attained certificate uses the direct reference path, as measured in Appendix G.1.2. Splitting the online error over more edges lengthens that absolute bound at a fixed sample count.

The union bound of Proposition 4 over edges uses the declared coverage guarantees (de Heide, 2026). The experiments reserve $0.05/2$ each for online edges and the pooled depth calculation. Every reported certificate is attained by direct reference paths $b \to z \to a$, which rely on the absolute edge alone. Proposition 7 gives the joint budget $\delta_q = \delta_s + \delta_c$ of a conformal depth edge for score error and calibration error, and Appendix H.6 gives the sample size of a certificate on fresh paired differences.

Proposition 7 (Conformal depth budget)**.** *Let $g(x) = \max_i(m_{(i,k)} - m_{(i,D)})$, let a per-state score $S(x) \in [0, C]$ satisfy $\Pr(g(x) > S(x)) \le \delta_s$, and let calibration and test scores be exchangeable.*

Table 2: Certification at $\epsilon = 0.10$. n: first certifying per-edge budget.

Rule	n	States	Prefix	Bits	Rollouts
LaRA-VLA, LIBERO, 8 states					
Absolute only	48	6	6	2,240	48.0
One-candidate receipt	48	7	6	3,008	96.0
PAIRGAP unified	64	6	6	2,240	319.0
Full receipt	48	7	0	5,312	48.0
Paired and depth	64	0	0	2,240	295.8

With q the rank-$\lceil (M+1)(1-\delta_c) \rceil$ score of M calibration states, or $q = C$ when that rank exceeds M, $\Pr(g(x) > q) \le \delta_s + \delta_c$.

Proof. Split conformal calibration (Lei et al., 2018) gives $\Pr(S(x) > q) \le \delta_c$, and the support bound covers the case $q = C$. The event $g(x) > q$ is contained in the union of $g(x) > S(x)$ and $S(x) > q$, and the union bound gives the claim for all candidate depth gaps jointly. □

The implemented conformal calculation sets both component errors to 0.025, corresponding to a joint depth budget of 0.05. Table 2 uses the pooled calculation; it contains no conformal-calibrated result.

The matched-panel pooled calculation aggregates prefix-worse counts from other states. The conditional conformal construction of Proposition 7 allocates errors jointly and requires at least $1/\delta_c - 1$ calibration states for a finite rank within the calibration sample. Table 2 reports each certified state's regret on held-out seeds next to its certificate.

G CERTIFICATE EXPERIMENTS

The experiments ask three questions: at which depth and bit count a received prefix can be certified, which witness path decides each certificate, and what a shallower executed depth costs over an episode.

G.1 PROTOCOL

Panel. LaRA-VLA (Bai et al., 2026) proposes four candidate chunks at each of 8 matched LIBERO (Liu et al., 2023) states, restored simulator states at official initial-state indices. Each candidate is coded as eight frequency rows, with the one-row prefix at 560 bits and the full code at 1,328 bits. Every option carries logged continuation rollouts on a shared seed list, with task failure as the cost, so $C = 1$ and $\lambda_b = 0$ in the regret. Even seeds feed the certificate and odd seeds form a held-out audit on which we recompute the regret of each executed option. The depth edge of a state pools the prefix-worse discordance of the other seven states. The error budget is $\delta = 0.05$, split into $\delta_q = \delta_o = \delta/2$, and we report tolerances $\epsilon \in \{0.05, 0.10\}$ at $n \in \{16, 32, 48, 64\}$ samples per sampled edge.

Arms and replayed rules. The five arms of Appendix F.2 share every input. The absolute arm replays the one-policy certificate that betting and binomial intervals give from a policy's own rollouts (Chen & Weng, 2026; Snyder et al., 2026), the paired and depth arm replays anytime-valid paired comparison (Li et al., 2026b) composed with the depth edge, and the two receipt arms replay the complete-chunk receivers of streaming controllers (Black et al., 2025b; Li et al., 2026e). Table 2 reports, per arm, the smallest n at which it certifies any state, the states certified there, those certified at prefix depth, the bits per decision and the unique rollouts. The held-out audit evaluates whether a certified state's regret exceeds ϵ.

G.1.1 CERTIFIED DEPTH PER STATE

The one-row prefix is certified at 6 of 8 states with 2,240 bits per decision. The absolute arm certifies the first candidate's prefix at 6 states once $n = 48$ (Table 2), receiving four one-row prefixes and no suffix. The unified arm certifies the same 6 states at $n = 64$, and every one of its certificates

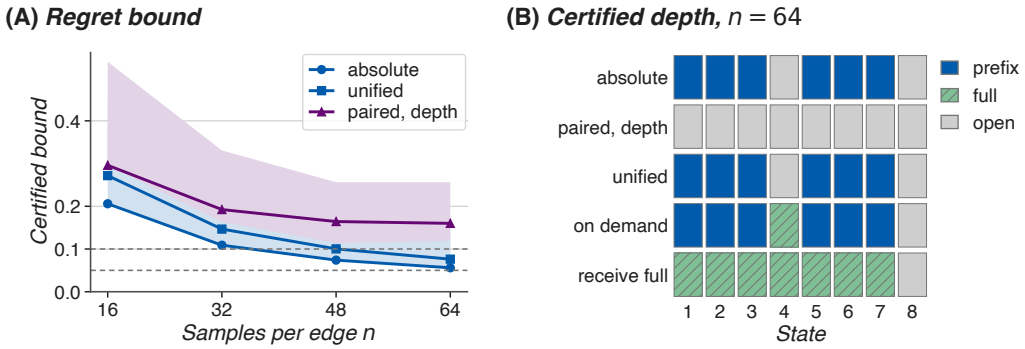

Figure 8: Per-state check on matched LaRA-VLA states. (A) Regret bound against samples per edge. (B) Certified depth at $\epsilon = 0.10$.

Table 3: States certified, of 8, by tolerance and samples n per edge.

	$\epsilon = 0.05$	$\epsilon = 0.10$			
n	≤ 64	16	32	48	64
Absolute	0	0	0	6	6
Paired, depth	0	0	0	0	0
Unified	0	0	0	0	6
On demand	0	0	0	7	7
Receive full	0	0	0	7	7

is attained on the reference path. One-candidate receipt completes the pre-registered candidate in every state and certifies 7 states at 3,008 bits, 6 of them at prefix depth. Receiving all four full codes and executing the pre-registered candidate certifies 7 states at 5,312 bits, all at full depth. On the held-out audit, 0 certified states exceed the tolerance across every arm, tolerance and budget. Figure 8B shows the certified depth of each state: state 8 stays open under every arm, and state 4 is certified only through its full code.

Tolerance 0.05 needs more than 64 samples of a failure-free option. No arm certifies at $\epsilon = 0.05$ (Table 3). With zero observed failures, the Clopper–Pearson bound is $1 - \delta^{1/n}$, which at $\delta = 0.025$ falls to 0.05 only after 72 samples; the grid therefore locates the budget each tolerance requires, and the certified bound falls with n for every arm (Figure 8A).

G.1.2 Which witness path decides

Every certificate on this panel is decided on the reference path, as Proposition 5 predicts for a saturated panel. Figure 10 shows the resulting arm bounds. At $n = 64$ the calibrated depth edges are at most 0.006, so each unreceived full code sits within that distance of its received prefix, and the binding sources are the received rivals. Their longest relational paths run from 0.12 to 0.26, since the executed option and its rivals succeed on almost every shared seed, and by part (iv) a concordant pair is no shorter than the absolute bound. The paired and depth arm, which has no absolute edge, therefore certifies no state, and the unified arm certifies exactly the states of the absolute arm. It reaches them one budget step later, because it divides the online error over four sampled edges and its reference path lengthens from 0.074 to 0.100 at $n = 48$. Part (ii) states what a panel needs for relational evidence to certify: states whose absolute bound exceeds ϵ and whose rivals are discordant with the executed option.

Depth relations are the most concordant relation class. On the 80 Flex-π diagnostic states of Appendix H, a prefix and its full code have mean discordance 0.036 across seeds, against 0.059 for two candidates at one depth, and 73 of the 80 states are saturated. The depth edge is therefore the cheapest edge to calibrate, and the witness path of a certificate is set by the evidence on received rivals.

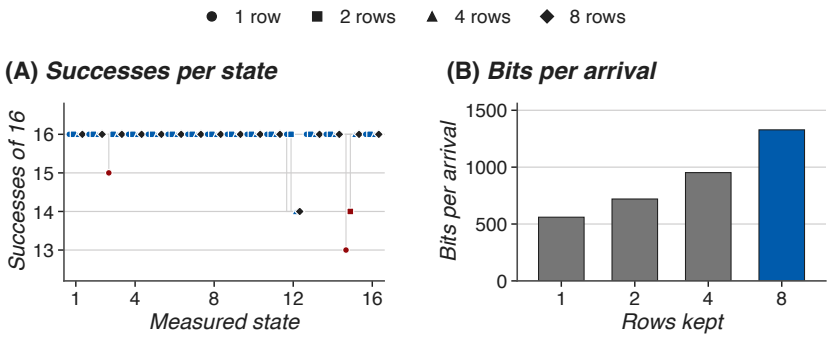

Figure 9: Single-decision outcomes at one to eight rows on sixteen LaRA-VLA states. (A) Successes per measured state. (B) Bits per arrival.

A two-row prefix reaches the full-depth success rate. On the sixteen LaRA-VLA states that execute one, two, four and eight rows at 560, 720, 952 and 1,328 bits (Figure 9), held-out success rises from 98.4% at one row to 99.2% at two rows and equals 99.2% at the full code, so a two-row prefix carries the full-depth rate at 54% of its bits.

G.1.3 Cost and scope of the certificate

Relational edges cost rollouts on every rival. The absolute arm spends 48 unique rollouts per decision at its certifying budget, all on the executed option, and on-demand receipt spends 96 by adding the incumbent's full code. The unified arm spends 319 at $n = 64$, because every signed paired edge runs a rival from the same seeds, and the paired and depth arm spends 296. The depth edge costs no decision-time rollout, since its calibration reuses the logged outcomes of other states once per panel. Four prefixes cost 2,240 bits per decision. Completing one of them adds $1{,}328 - 560 = 768$ bits, giving 3,008 bits for one-candidate receipt. The graph has $2K + 1 = 9$ vertices, so one Floyd–Warshall update takes $9^3 = 729$ relaxations.

Saturated logged populations close on the incumbent. The logged-population record of Appendix H shows the same regime at a finer tolerance: over the 80 Flex-π states with eight logged rollouts per option and $\epsilon = 0.01$, a certificate that reads the incumbent's own population first closes at 16.8 rollouts per decision, against 21.2 for an incumbent–rival rule and 50.2 to 51.3 for EB-TC$_\epsilon$, top-two Thompson sampling and UGapE under one exact stopping rule (Jourdan et al., 2023; Russo, 2020; Gabillon et al., 2012). At eight seeds per option each certified edge reads its whole population (Proposition 9), so the order in which options are read sets the rollout count.

What a relational edge needs to decide. By (13), a rival b on which the executed option never fails alone ($n_+ = 0$) has edge $u(0, n, \delta_p/2) - \ell(n_-, n, \delta_p/2)$, which becomes negative once the rival fails alone on enough seeds, $\ell(n_-, n, \delta_p/2) > u(0, n, \delta_p/2)$. A negative edge shortens every path through b, and part (ii) of Proposition 5 then certifies a whenever every such rival and every depth edge into it reach a within ϵ, including states where the executed option has failures and its absolute bound stays open. On the matched panel the rivals fail alone on few seeds, so the reference path remains the shorter one, and the count tells a deployer how many discordant seeds a harder task must show before paired rollouts of rivals repay their cost.

Scope. The certificate holds for the fixed candidate bank, the restorable simulator and the stated error construction, and its audit uses held-out seeds of the same states. The matched LaRA-VLA panel is saturated, so the composition proposition states the evidence a harder panel must supply for paired and depth edges to decide. The closed-loop price uses accounting bits on a continuous-action policy, and the break-even condition of Equation (3) applies to any per-arrival price.

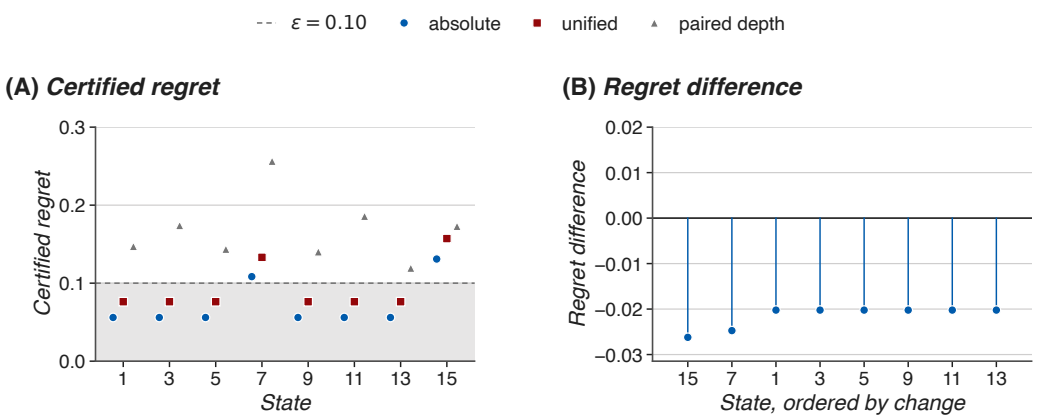

Figure 10: State-level certificate bounds at $n = 64$. (A) Certified regret per arm. (B) Absolute minus unified regret.

H LOGGED-POPULATION RECORD IN THE SATURATED REGIME

This appendix keeps the logged-population comparison of the earlier diagnostic panel. Every certificate here closes on endpoint intervals of the incumbent's own logged population, and the fresh-certificate price of Theorem 2 sets the rollout count that a fresh-seed certificate needs.

H.1 RESOLUTION OF A LOGGED CERTIFICATE

A seed population of size N fixes the certificate's evidence. Option a has per-seed support $[\ell_a, u_a]$, observed seeds I_a and endpoint interval $L_a = \big(\sum_{j\in I_a} Y_{a,j} + (N - |I_a|)\ell_a\big)/N$, $U_a = \big(\sum_{j\in I_a} Y_{a,j} + (N - |I_a|)u_a\big)/N$. A same-seed pair (a, b) observed on I_{ab} with joint support $S_{ab} \subseteq [\ell_a, u_a]\times[\ell_b, u_b]$ gives the edge

$$w(b \to a) = \frac{1}{N}\Big(\sum_{j\in I_{ab}} (Y_{a,j} - Y_{b,j}) + (N - |I_{ab}|)\, s_{ab}\Big), \qquad s_{ab} = \sup_{(y,y')\in S_{ab}} (y - y'),$$

which bounds $m_a - m_b$. The two statements below give the resolution of such an edge on a logged population.

Proposition 8 (Rectangle support removes the pairing gain)**.** *If $I_a = I_b = I_{ab}$ and $S_{ab} = [\ell_a, u_a] \times [\ell_b, u_b]$, then $w(b \to a) = U_a - L_b$. For $|I_{ab}| < N$ the paired edge is strictly tighter than $U_a - L_b$ exactly when $s_{ab} < u_a - \ell_b$.*

Proof. The observed terms of $w(b \to a)$ and $U_a - L_b$ coincide on the shared seeds, and each unobserved seed contributes s_{ab}/N to the first and $(u_a - \ell_b)/N$ to the second. A rectangle attains $\sup(y - y') = u_a - \ell_b$, and any joint support inside it has $s_{ab} \le u_a - \ell_b$. □

Proposition 9 (Resolution of a logged certificate)**.** *Let $Y \in \{0, 1\}$ and let every observed pair (a, b) be concordant. If the unobserved seeds admit a discordant outcome, $s_{ab} = 1$, then $w(b \to a) = (N - |I_{ab}|)/N$, and a certificate at $\epsilon < 1/N$ requires $|I_{ab}| = N$.*

Proof. Concordant observations contribute zero, and each unobserved seed contributes $1/N$. The bound is at most $\epsilon < 1/N$ only with no unobserved seed. □

At $N = 8$ and $\epsilon = 0.01$ every logged certificate therefore reads the whole population of each edge it closes, which is the saturated-incumbent regime this appendix records.

H.2 SETUP

We evaluate two vision-language-action policies on the LIBERO manipulation suites (Liu et al., 2023): Flex-π (Yan et al., 2026), contributing 80 states from eight tasks across four suites, and LaRA-VLA (Bai et al., 2026), contributing 16 matched states at its native eight-step horizon. The unit of evaluation is a *state*, a restored simulator state with a fixed bank of candidates, nested from two candidates to four. Continuation cost is $Y = 1 - g + 0.01 \min\{h, H\}/H$ for satisfied-goal fraction g, executed steps h and task horizon H, so $C = 1.01$ and $\epsilon = 0.01$ equals the full step-time term. Every selector draws on the same logged population of eight continuation rollouts per option, chooses which outcomes to acquire, and certifies regret over that fixed population. After selection freezes, four fresh seeds per state give 320 Flex-π and 64 LaRA-VLA held-out outcomes per selector.

Panel. Every Flex-π and LaRA-VLA number below, Table 6 included, comes from a *diagnostic panel*: four suites, two tasks per suite drawn by a seeded hash over task names before any outcome was observed, and official initial-state indices 20–29, 4% of the official LIBERO protocol of ten tasks per suite at fifty initial states. Held-out success moves in steps of one episode, 0.31 points in 320 Flex-π outcomes and 1.56 points in 64 LaRA-VLA outcomes. In Table 6, success uses the initial held-out draw, and Δ pairs each selector with the first-candidate control on the shared state and continuation seed. Section 5.2 runs OpenVLA-OFT on the official protocol, all ten tasks of Spatial and Goal at fifty initial states per task.

Baselines. *Fixed prefix* and *fixed full* always transmit one depth, *tree refinement* certifies through a stop-and-refine tree with separate candidate and depth tests (Even-Dar et al., 2006), *uniform* advances every population in lockstep, the full-tolerance *incumbent–rival* rule samples the incumbent and the competitor limiting its endpoint certificate (Appendix E), an *oracle* reads every population, and the *first candidate* control fixes the action before any rollout. Rollouts and bits are per decision, at a bit price of 5×10^{-8} and a tolerance of $\epsilon = 0.01$ in those same cost units. At this price the communication term is at most 3.1×10^{-4}; Figure 12D counts the choices that other prices move, and Appendix J sweeps fourteen prices.

H.3 THE COST OF THE CERTIFICATE

An ϵ certificate costs 16.8 rollouts against 21.2 and 57.8 (Tables 6 and 4). Against the strongest full-tolerance comparator the rollout ratio is $21.2/16.8 = 1.26$. Against tree refinement it grows from 2.51 at two candidates to 3.44 at four (Table 7), because PAIRGAP exhausts one runnable incumbent before touching its rivals, and the ratio holds over all 24 candidate orders (Figure 12B).
Modern ϵ-best-arm rules need three to seven times the rollouts. We give EB-TC$_\epsilon$, top-two Thompson sampling and UGapE the same exact stopping rule, so only the choice of the next outcome differs. They spend 50.2 to 51.3 rollouts on Flex-π and 51.2 to 55.6 on LaRA-VLA (Table 4). These rules balance samples between a leader and a challenger, and the certificate here closes once the incumbent's own population is read: the tolerance $\epsilon = 0.01$ equals the full step-time term of Y, so an incumbent whose continuations all succeed has worst-case cost within ϵ of the zero floor. At $\epsilon = 0.005$ PAIRGAP spends 20.9 rollouts against 23.0 for the incumbent–rival rule and 50.0 for EB-TC$_\epsilon$, and Table 5 sweeps the tolerance. Every certificate-producing selector certifies the same tolerance at all 80 states. On LaRA-VLA, every state is a saturated incumbent (Proposition 10), and PAIRGAP spends 8.0 against 59.8 and executes the first candidate. Figure 11 shows the saving at every state.

Proposition 10 (Saturated incumbent)**.** *In the two-level setting with $K \geq 2$, nonnegative bit prices and $\mu_{P_1} \leq \epsilon < C/N$, the prefix-first finite-population implementation uses exactly N rollouts and returns P_1.*

Proof. Initially, P_1 has the smallest upper endpoint under the prefix-first tie rule. After n of its outcomes have been observed,

$$U_{P_1}(n) = \frac{T(n) + (N - n)C}{N} \leq C.$$

Unqueried prefixes retain upper endpoint C and full candidates retain upper endpoints at least C, so P_1 remains incumbent. For $n < N$, $U_{P_1}(n) \geq C/N > \epsilon$ and an unqueried prefix competitor

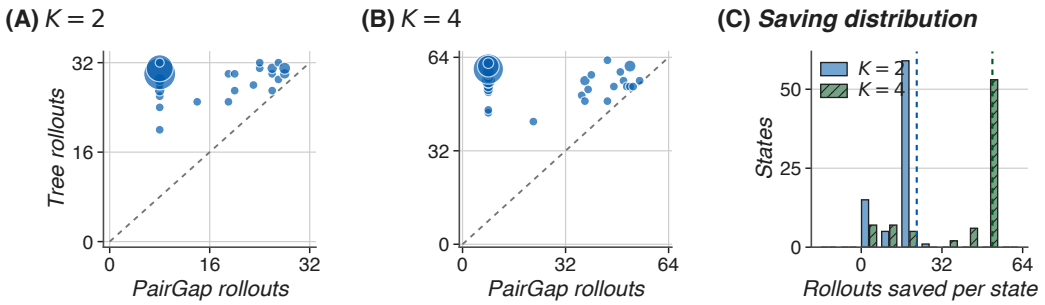

Figure 11: Rollouts per decision over 80 states at reveal order 17. (A, B) PAIRGAP and tree refinement at two and four candidates. (C) Saving distribution.

Table 4: Certifying selectors under candidate-order stress at $\epsilon = 0.01$.

	Rollouts		Candidate order	
Selector	Flex-π	LaRA	Mean	Same (%)
PAIRGAP	16.8	8.0	16.7	36.6
Incumbent–rival	21.2	18.4	21.5	46.5
EB-TC$_\epsilon$ (Jourdan et al., 2023)	50.6	55.6	50.8	84.2
Top-two Thompson (Russo, 2020)	50.2	51.2	50.0	28.6
UGapE (Gabillon et al., 2012)	51.3	55.6	51.6	61.1
Tree (Even-Dar et al., 2006)	57.8	59.8	58.1	91.0
Round robin	63.5	64.0	63.6	97.9

retains lower endpoint zero. That competitor remains active. At $n = N$, $U_{P_1}(N) = \mu_{P_1} \leq \epsilon$, and all competing lower endpoints are nonnegative. The certificate therefore holds exactly at exhaustion under every fixed reveal order. The condition $K \geq 2$ supplies the unqueried zero-price competitor. □

Table 6 lists the fixed-action controls and the certificate-producing selectors, and Tables 7 and 4 report the incumbent–rival rule on both policies. Its LaRA-VLA columns cover 16 matched states, all saturated incumbents, where the prefix-first rule of PAIRGAP reduces to the first-candidate control and tree refinement and uniform keep sampling. In Table 7, EB-TC$_\epsilon$ (Jourdan et al., 2023), top-two Thompson (Russo, 2020) and UGapE (Gabillon et al., 2012) choose the next outcome.

H.4 WHAT THE CERTIFICATE BUYS

Figure 12 collects the supporting views on Flex-π: held-out cost against the fresh rollout budget (A), mean rollouts for each of the 24 candidate orders on the logged populations (B), the empirical distribution of the paired variance ratio over all option pairs (C), and the number of the 80 executed choices that change when the bit price moves away from 5×10^{-8} (D).

PAIRGAP certifies at lower rollout cost under every allocation seed. PAIRGAP lowers the rollout cost of obtaining a certificate at all 80 states under each of three allocation seeds at four candidates and at 236 of 240 state–seed pairs at two candidates (Appendix J). Frozen selections evaluated on fresh continuations keep every selector of Table 6 within three Flex-π episodes of one another, and Appendix J.2 reports the paired held-out comparisons and three continuation-seed replicates.

Held-out cost gains come from fresh sampling rules, and pairing leaves the choice unchanged at this panel size. On all 20 fresh seeds per option, the initial draw and four continuation-seed replicates, of which Appendix J.2 compares the first three, half for acquisition and half held out over 20 splits, eleven sampling rules lower held-out cost from 0.0420 to 0.0360–0.0412 at 32 to 96 rollouts, at success within 0.3 points (Table 8, Figure 12A). The gap-driven rule samples the empirical incumbent together with the rival whose paired lower confidence bound on its gap is

Table 5: Rollouts per decision to a certificate across tolerances.

	Flex-π, ϵ			LaRA-VLA, ϵ		
Selector	0.0025	0.005	0.01	0.0025	0.005	0.01
PAIRGAP	36.6	20.9	16.8	35.2	13.6	8.0
Incumbent–rival	36.5	23.0	21.2	36.6	19.5	18.4
EB-TC$_\epsilon$ (Jourdan et al., 2023)	49.3	50.0	50.6	54.5	55.1	55.6
Top-two Thompson (Russo, 2020)	50.3	50.2	50.2	51.2	51.2	51.2
UGapE (Gabillon et al., 2012)	51.4	51.3	51.3	55.6	55.6	55.6
Tree (Even-Dar et al., 2006)	57.8	57.8	57.8	59.8	59.8	59.8
Round robin	63.5	63.5	63.5	64.0	64.0	64.0

Table 6: Held-out execution and acquisition cost. Δ: held-out episodes won minus lost against the first-candidate control.

	Flex-π				LaRA-VLA			
Selector	Succ. (%) ↑	Δ	Rollouts ↓	Bits ↓	Succ. (%) ↑	Δ	Rollouts ↓	Bits ↓
Fixed-action controls and certificate-producing selectors								
First candidate	96.9	0	0.0	2,240	96.9	0	0.0	2,240
Fixed prefix	96.6	−1	32.0	2,240	100.0	+2	32.0	2,240
Fixed full	95.9	−3	32.0	6,240	98.4	+1	32.0	5,312
Uniform	96.9	0	63.8	2,478	98.4	+1	64.0	2,288
Tree refinement (Even-Dar et al., 2006)	96.6	−1	57.8	2,415	98.4	+1	59.8	2,288
PAIRGAP	96.6	−1	**16.8**	2,340	96.9	0	**8.0**	2,240

smallest, and executes the lowest empirical mean. Paired and unpaired variants of each rule track one another, and the certificate rule executes the first candidate, consistent with the sample price of Theorem 2.

H.5 PAIRED-ROLLOUT VARIANCE

Pairing reduces variance, and the support width sets the fresh-sampling certificate. Across all Flex-π option pairs the paired variance ratio has median 0.66 (Figure 12C, Appendix K). The same-seed envelope width s sets the fresh price of Theorem 2 (Figure 13): under a width fitted to the observed differences, the 34 of 56 matched-panel edges whose differences stay below ϵ would need 2 to 36 pairs, and Appendix K.2 recomputes the certificate under the universal support.

H.6 FRESH-CERTIFICATE SAMPLE SIZE

A certificate for a fresh seed distribution pays for the width of the paired support, and among rules that certify on an all-nonpositive record the count it needs has a matching lower and upper bound.

Theorem 2 (Fresh-certificate sample size)**.** *Let fresh paired differences $\Delta = Y_a - Y_b$ take values in $[-s, s]$, with $0 < 2\epsilon < s$ and $0 < \alpha < 1$. A rule that certifies $m_a - m_b \le \epsilon$ on the all-nonpositive record of n pairs with error at most α needs $n \ge \log(1/\alpha)/(-\log(1 - 2\epsilon/s))$. The zero-record rule, which certifies exactly when all n differences are nonpositive, has error at most α once $n \ge \log(1/\alpha)/(-\log(1 - \epsilon/s))$. Both counts equal $(s/\epsilon)\log(1/\alpha)$ up to a factor of two.*

Proof. For the lower bound, let $\Delta = s$ with probability $p = 2\epsilon/s$ and $\Delta = 0$ otherwise, so $m_a - m_b = 2\epsilon$ exceeds the tolerance. Independent pairs produce the all-nonpositive record with probability $(1-p)^n$, and a rule that certifies on that record errs with at least this probability. Error at most α forces $(1-p)^n \le \alpha$, and taking logarithms gives $n \ge \log(1/\alpha)/(-\log(1-p))$. Since $-\log(1-p) \le p/(1-p)$, the count is at least $((1-p)/p)\log(1/\alpha)$.

For the upper bound, let the distribution of Δ have mean at least ϵ. Splitting the mean by the sign of Δ and using $\Delta \le s$ gives $\epsilon \le s\Pr(\Delta > 0)$, so $\Pr(\Delta \le 0) \le 1 - \epsilon/s$. The zero-record rule errs only when every pair is nonpositive, which has probability at most $(1-\epsilon/s)^n \le \alpha$ under the stated count. The two thresholds differ by the factor two in the Bernoulli parameter. The values below

Table 7: Rollouts per decision to an $\epsilon = 0.01$ certificate under one exact stopping rule.

Selector	Flex-π	LaRA-VLA
PAIRGAP	1.00	1.00
Incumbent–rival	1.26	2.30
EB-TC$_\epsilon$	3.02	6.95
Top-two Thompson	2.99	6.41
UGapE	3.06	6.95
Tree	3.44	7.47
Round robin	3.78	8.00

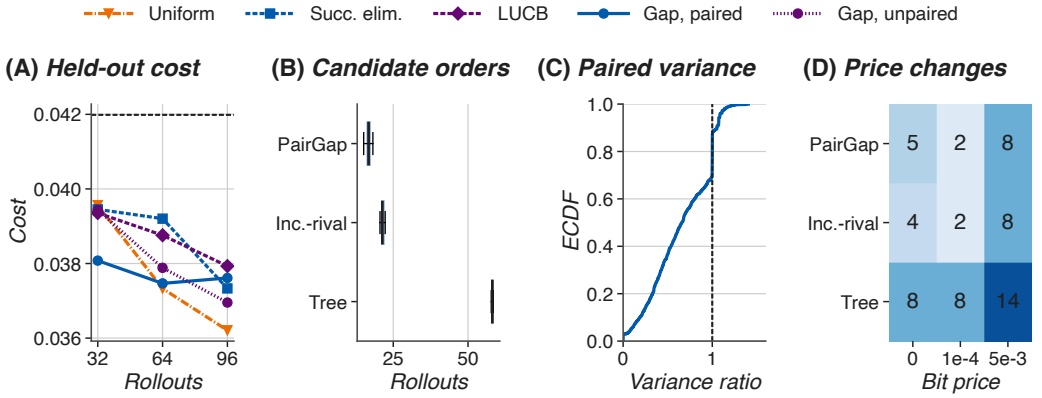

Figure 12: Fresh sampling and logged certification on Flex-π. (A) Held-out cost. (B) Mean rollouts over 24 candidate orders. (C) Paired variance ratio. (D) Choices that change with the bit price.

use $s = C = 1.01$ and $s = 0.05$ with $\epsilon = 0.01$ and $\alpha = 1/40$; the deterministic envelope below supplies a width of this kind from simulator sensitivity. □

Proposition 11 (Deterministic same-seed envelope)**.** *Suppose state distances satisfy $d_{t+1} \le Ld_t + Be_t$ and absolute stage-cost differences are at most $\kappa d_t + \eta e_t$ for nonnegative constants. Define $E_0 = \delta$, $E_{t+1} = LE_t + Be_t$ and $G_T = \sum_{t<T}(\kappa E_t + \eta e_t)$. Two same-seed rollouts started at distance δ satisfy $|Y_a - Y_b| \le G_T$, and with zero initial distance and constant action error r, $G_T(r) = rG_T(1)$.*

Proof. Induction gives $d_t \le E_t$, and summing stage-cost bounds gives $|Y_a - Y_b| \le G_T$. Nonnegative coefficients make the envelope monotone in initial distance and action errors. With zero initial distance and constant action error r, linearity yields $G_T(r) = rG_T(1)$. Averaging seedwise envelopes and adding $c_a - c_b$ produces a valid total-cost edge. Simulator-wide sensitivity constants require separate calibration; the reported robot experiments use bounded support and the stated statistical intervals. □

The universal support $[-C, C]$ of the certification experiments, $s = C$, gives 185 and 371 pairs per edge at $\alpha = 1/40$, and a calibrated same-seed envelope (Proposition 11) of width $s = 0.05$ gives 8 and 17. Same-seed pairing therefore enters a fresh certificate through the envelope width, and its variance reduction enters through the interval rule.

I EXPERIMENTAL PROTOCOL

I.1 STATES, CANDIDATES AND CONTINUATION OUTCOMES

The Flex-π evaluation uses two outcome-blind, hash-selected tasks from each of LIBERO Spatial, Object, Goal and LIBERO-10, with official initial-state indices 20–29: eight tasks and 80 states.

Table 8: Held-out physical cost of fresh-seed selection rules on the 80 Flex-π diagnostic states. Success and the lower and higher counts use the 96-rollout budget.

Rule	32	64	96	Succ. (%)	Lower	Higher
First candidate	0.0420	0.0420	0.0420	95.9	0	0
Uniform	0.0396	0.0373	0.0362	96.2	51	27
LUCB (Kalyanakrishnan et al., 2012)	0.0394	0.0388	0.0379	95.8	47	32
UGapE (Gabillon et al., 2012)	0.0412	0.0403	0.0409	96.0	2	4
Successive elimination (Even-Dar et al., 2006)	0.0395	0.0392	0.0373	95.9	46	32
Sequential halving (Karnin et al., 2013)	0.0377	0.0375	0.0370	96.1	46	33
GRAD (Ménard, 2019)	0.0372	0.0360	0.0371	96.0	47	32
VarDE (Luong et al., 2026)	0.0388	0.0372	0.0371	96.0	47	31
PAIRGAP certificate	0.0420	0.0420	0.0420	95.9	0	0
Gap-driven, paired	0.0381	0.0375	0.0376	95.9	49	30
Gap-driven, unpaired	0.0394	0.0379	0.0370	96.1	48	29
Closure elimination, paired	0.0393	0.0374	0.0371	96.0	48	31
Closure elimination, unpaired	0.0388	0.0376	0.0371	96.0	49	31

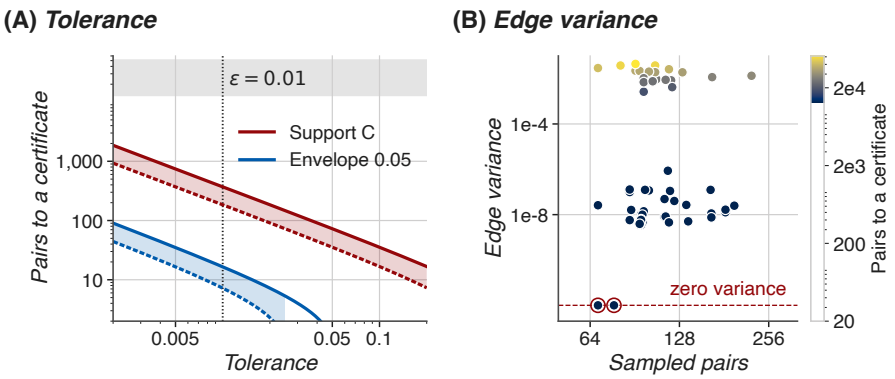

Figure 13: Pairs to a certificate. (A) Upper (solid) and lower (dashed) bounds of Theorem 2 at $\alpha = 1/40$ against tolerance. (B) Pairs against edge variance.

This is a diagnostic panel covering 4% of the official LIBERO protocol, which runs ten tasks per suite at fifty initial states per task. The OpenVLA-OFT closed-loop measurements in Section 5.2 use that official protocol; every Flex-π and LaRA-VLA number comes from the diagnostic panel. Development states are disjoint. Candidate proposal seeds define nested two- and four-candidate banks. Every candidate has a prefix and a full-depth version. Each version receives eight restored-state continuation outcomes for certification. The four-candidate population contains 5,120 outcomes, including the 2,560 outcomes of the nested two-candidate bank.

The continuation cost is

$$Y = 1 - g + 0.01 \min\{h, H\}/H \in [0, 1.01],$$

where g is the satisfied-goal fraction, h is executed steps and H is the task horizon. The finite-population estimand averages the eight fixed continuation outcomes. Each selection rule reveals entries in a fixed permutation. Allocation seed 17 defines the reported operating point; seeds 29 and 43 assess reveal-order sensitivity. The main price is $\lambda_b = 5 \times 10^{-8}$, the tolerance is $\epsilon = 0.01$, and the price sweep uses 14 values from zero to 0.005.

Selections are frozen before independent evaluation. Four new continuation seeds for each candidate and depth produce 2,560 additional Flex-π trajectories. Each selector's reported success rate therefore uses 320 held-out outcomes. These seeds participate exclusively in execution evaluation. Population regret and held-out physical cost are separate estimands.

LaRA-VLA uses the same eight tasks with initial-state indices 21 and 28, giving 16 matched states. Its native action horizon is eight; Flex-π uses ten. Four candidates at two depths and eight acquisi-

tion seeds produce 1,024 outcomes. Four fresh seeds produce 512 independent outcomes, with 64 selected-action outcomes per reported selector. Each model retains its own continuation policy.

I.2 Communication and baseline accounting

All candidate prefixes are prepaid. Remaining suffix bits enter the selection objective and total transmission includes the prepaid prefixes. A simulation rollout is one restored-state continuation; a paired observation uses two rollouts under the same seed. Fixed prefix and fixed full select within their declared depths. The first-candidate control executes a predetermined prefix with zero selection rollouts. Tree refinement and uniform allocation produce certificates; their candidate and depth tolerances sum to ϵ. The oracle reads every population. Each aggregate is a state mean or an outcome proportion with the denominator stated above.

The candidate, depth and seed are matched across selection rules. Full-depth and prefix executions share the restored state and continuation seed. Reused outcomes in the depth sweep additionally require identical decoded actions. The logged-population certificate concerns those fixed outcomes; general deployment requires valid fresh-sampling intervals or calibrated surrogate-error bounds.

The logged replay exposes fixed server-side outcome populations before receiver-side suffix transmission, so $S = \mathcal{A}$ and E grows as tokens arrive. This protocol isolates allocation over known candidate banks. The paired-sampling panel supplies all candidate tokens before acquisition. Native streaming evaluation records the controller's actual simulator and receiver availability.

I.3 What a seed controls

On the executed path every random draw is derived by hash. A draw seed is the first eight bytes of a SHA-256 digest taken over a purpose label together with the suite, the task index, the official initial-state index and, for candidate proposal, the candidate index. Candidate proposal and held-out continuation carry different purpose labels, so their draws are independent by construction. A single configuration seed also exists, reaching environment construction and the framework's global seeder. It changes no reported outcome. Initial states are restored from the official fixed state array, each restored state is verified against the recorded digest of its decision state before any candidate is proposed, and every policy call receives its derived seed explicitly. Independent repetition therefore lives in the continuation domain, a purpose label that enters the same digest alongside the decision-state identifier and the sample index. The three replicates of Appendix H.4 vary only that label over the same 80 decision states, the same candidate banks and the same frozen selections. Varying it redraws every held-out continuation and leaves all decision inputs fixed, which is the repetition the replicate spread measures. The allocation seeds 17, 29 and 43 control a separate object, the fixed permutation in which each selection rule reveals entries of the already-logged population. Figure 14 shows what that permutation moves.

I.4 Reproduction

The study records model and simulator revisions, state identities, proposal and continuation seeds, and compact outcomes. Analysis generators compute the reported aggregates, tables and figures from the corresponding outcome populations. The protocols above specify task, seed and resource conditions; the submitted PDF provides the supporting aggregate results.

J Supplementary decision and execution results

J.1 Candidate-count and allocation effects

The two-candidate and four-candidate banks share 80 states and their original candidate identities. At $\epsilon = 0.01$ and $\lambda_b = 5 \times 10^{-8}$, PairGap uses 11.8625 and 16.7875 rollouts; tree refinement uses 29.8 and 57.8125; the full-tolerance incumbent–rival rule uses 17.5625 and 21.1875. The rollout reduction from tree refinement holds at all 80 states under reveal order 17, and at four candidates under all three allocation seeds. At two candidates it holds at 236 of 240 state–seed pairs, with two ties under seed 29, one tie under seed 43 and one seed-43 state where tree refinement spends one rollout fewer (Figure 14). Figure 16 gives the rollout cost and the held-out outcomes at four

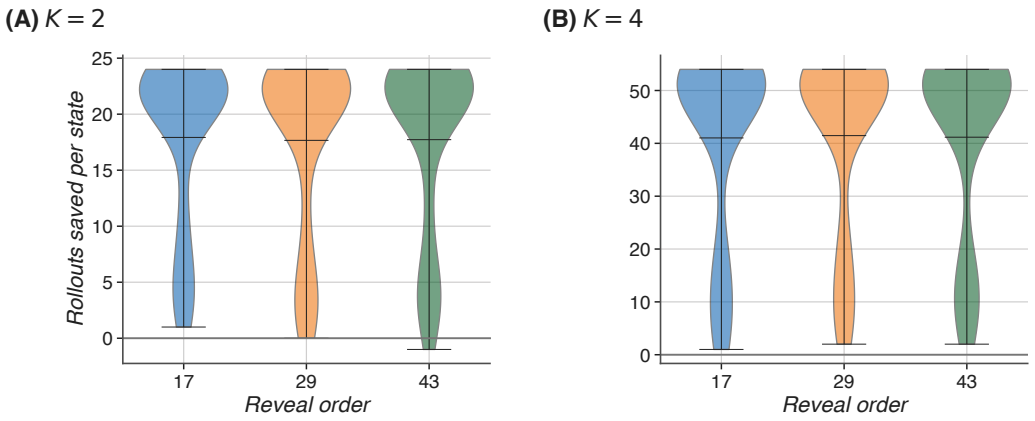

Figure 14: Rollouts tree refinement spends beyond PAIRGAP per state under allocation seeds 17, 29 and 43. (A) Two candidates. (B) Four candidates.

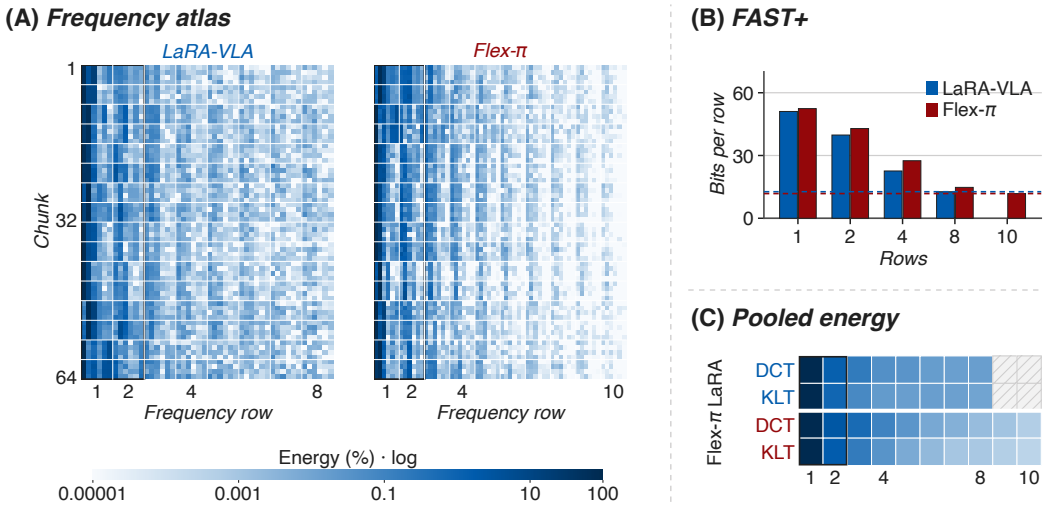

Figure 15: Frequency structure and coding cost. (A) Frequency atlas. (B) FAST+ bits per row. (C) Pooled DCT and KLT energy.

candidates. The full-population oracle reads 32 and 64 outcomes. Figure 17 carries both quantities across the price grid, where PAIRGAP spends the fewest rollouts at all fourteen prices and both bank sizes.

At four candidates, fixed prefix and fixed full satisfy the population tolerance at 73 and 79 of 80 states. PAIRGAP, tree refinement, uniform allocation and the oracle satisfy it at every state (Figure 18).

Tree refinement's mean population regret is 9.6995×10^{-6}; PAIRGAP has 1.0897×10^{-4}. Both lie below the 0.01 tolerance. At tolerance 0.005, PAIRGAP uses 13.625 and 20.8625 rollouts with the same selected bits and held-out success counts as at 0.01. These allocation variants are developmental reanalyses of the fixed populations. The subsequent independent continuations evaluate selections frozen before those outcomes were generated. Figure 19 records where tree refinement stops and where it refines across the same price grid.

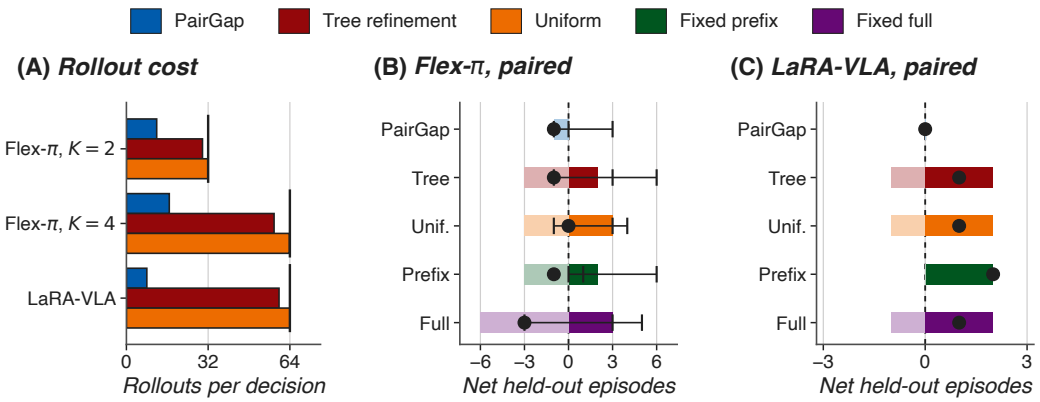

Figure 16: Certification cost and held-out outcomes at four candidates. (A) Rollout cost per decision. (B) Flex-π and (C) LaRA-VLA held-out episodes, paired against the first-candidate control.

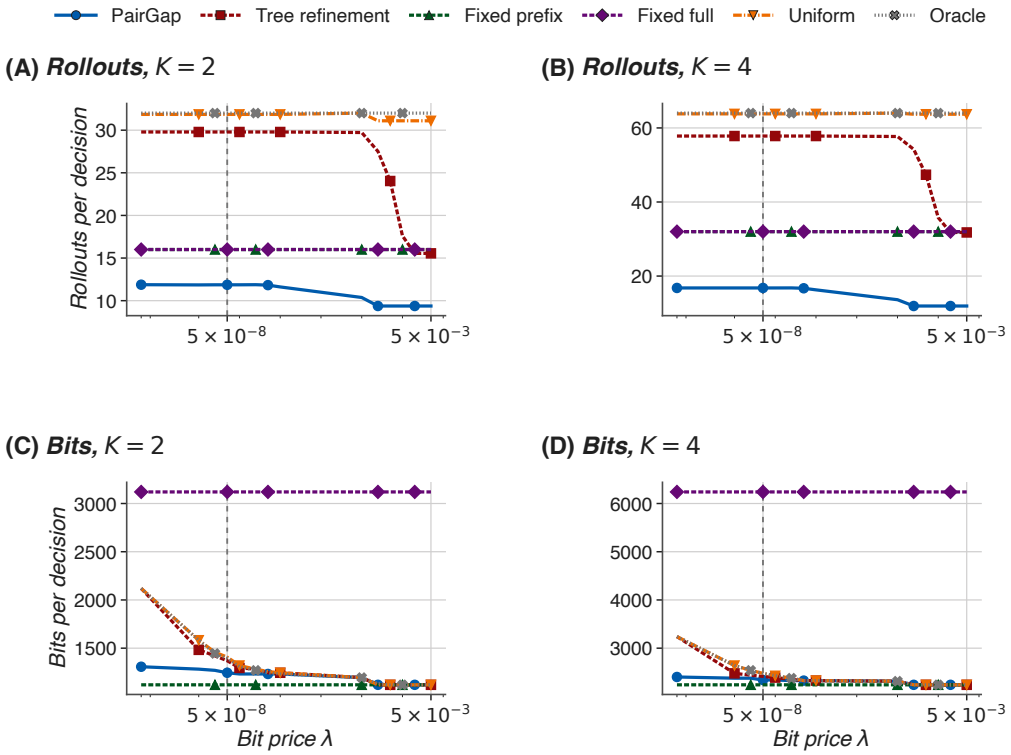

Figure 17: Price sweep over the 80-state panel. (A, B) Rollouts per decision at two and four candidates. (C, D) Bits per decision.

J.2 Independent task costs

The acquisition population and held-out execution have distinct evaluation roles. Table 9 reports physical costs alongside the main table's success and resource measurements. On Flex-π, PairGap and tree refinement both achieve 309 successes among 320 continuations. Their mean physical costs differ by 0.002134437, with 39 positive state differences, 14 negative differences and 27 ties. Each state effect averages four paired continuations. The retrospective first-candidate prefix control

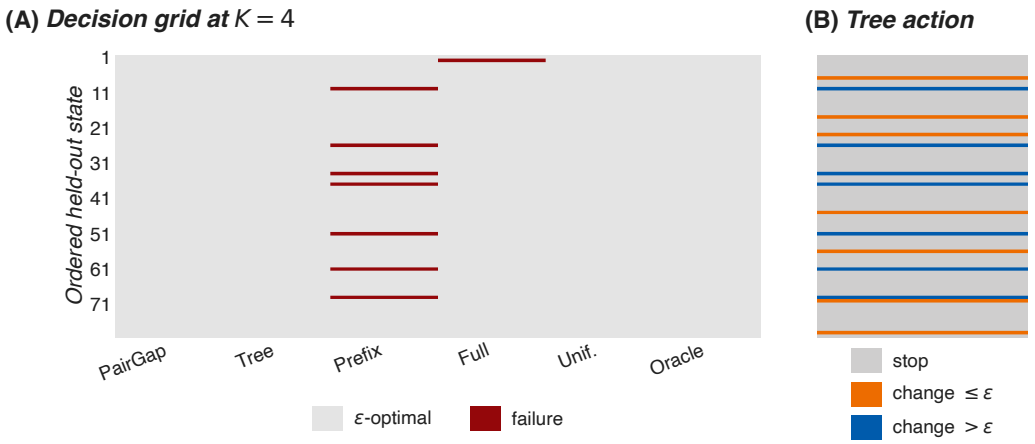

Figure 18: Per-state outcomes at four candidates. (A) ϵ-optimality per selector. (B) Executed choice of tree refinement.

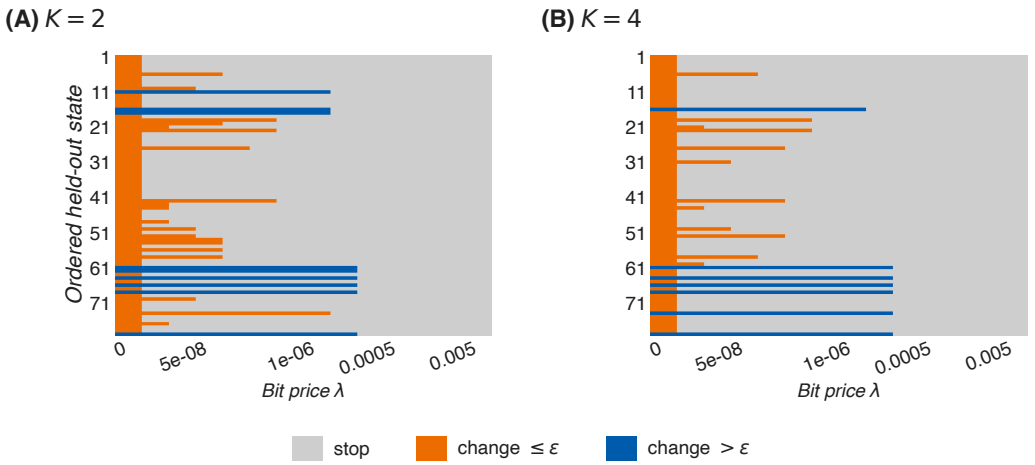

Figure 19: Stop-and-refine decisions of tree refinement across the price grid. (A) Two candidates. (B) Four candidates.

achieves 310 successes and uses zero selection rollouts. Figure 20 places each certifying selector against that control at all three continuation-seed replicates.

The certified options of the certificate-producing selectors can differ. On the initial held-out draw of four fresh seeds per state, PAIRGAP, tree refinement and the first candidate reach mean costs 0.0337, 0.0316 and 0.0327. Frozen selections place every selector of Table 6 inside three Flex-π episodes, from 307 of 320 outcomes for fixed full to 310 for the zero-rollout first-candidate control. Percentages in Table 6 round half up, and its first-candidate rows coincide across the two panels at 96.875% and four 560-bit prefixes. Paired on the shared state and continuation seed, PAIRGAP loses one episode to that control and wins none, at exact binomial $P = 1$; tree refinement and fixed prefix each win two and lose three, also at $P = 1$. On LaRA-VLA, where one episode is 1.56 points, PAIRGAP and the control succeed on the same 62 of 64 outcomes, and fixed prefix reaches 64 with two paired gains, zero losses and $P = 0.50$. Three continuation-seed replicates over the same 80 states give PAIRGAP 304, 310 and 303 successes of 320, a replicate spread of 2.19 points, and move its paired difference against the control from -1 episode to $+3$ and back to zero. Those three differences span 1.25 points with a standard deviation of 0.65, which sets the scale a held-out

claim would need: that spread reaches 0.3 points at 376 states and 1,504 held-out outcomes, nearly five times this panel, the order of the 500 episodes per suite of Section 5.2.

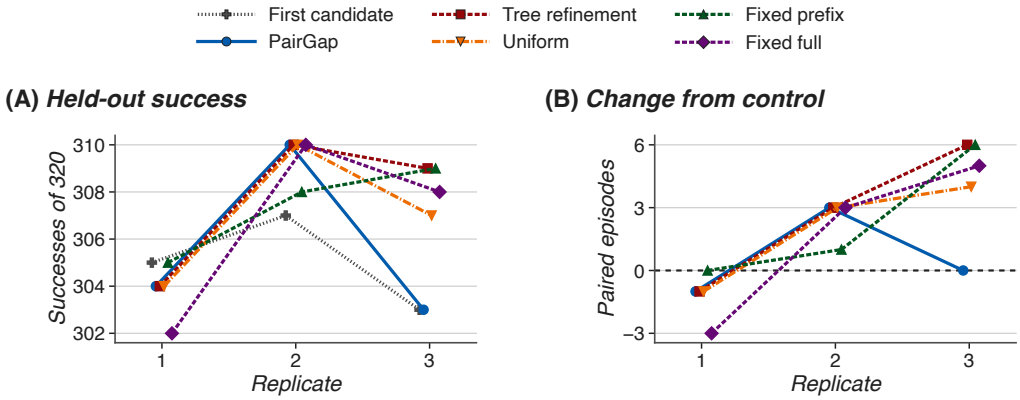

Figure 20: Replicate variation. (A) Held-out success per replicate. (B) Change from the first-candidate control.

Table 9: Mean physical continuation cost on independently seeded execution.

Selector	Flex-π	LaRA-VLA
First candidate	0.032655	0.025688
Fixed prefix	0.031572	0.004707
Fixed full	0.032649	0.020372
Uniform	0.028430	0.020406
Tree refinement	0.031568	0.020406
PairGap	0.033702	0.025688

J.3 LaRA-VLA and candidate-order sensitivity

The largest first-prefix acquisition mean in the original 16-state order is 0.00734375. It satisfies Proposition 10, and PairGap uses eight rollouts and selects that prefix at every state. Its 62 successes among 64 independent outcomes coincide with the predeclared zero-rollout first-candidate control. Population-selected fixed prefix changes the candidate at 12 states, gains two independent successes and loses none. It lowers mean cost by 0.020980407.

A retrospective replay evaluates all 24 candidate permutations at four tolerances, preserving actions and each option's outcome order. The 384 state–order combinations reuse the original independent outcomes. Mean rollout counts at tolerances 0, 0.001, 0.005 and 0.01 are 62.3125, 52.8594, 14.3438 and 9.5156. Exactly eight rollouts suffice in 0, 0, 192 and 372 combinations, respectively. At 0.01, 12 combinations change the first candidate's executable action, preserving success and changing mean physical cost by $+0.000004808$. Candidate access order and tolerance therefore affect the acquisition schedule. The zero-tolerance population optimum over all eight options, full actions and bit prices included, has independent cost 0.02040556, and the population-selected best of the four prefixes has 0.004707; the order-averaged first prefix has cost 0.01780639. These quantities condition on the same observed candidate banks and separately seeded execution.

J.4 Selection at the arrival state

The delayed-arrival study uses eight states, four matched continuation samples, four arrival ticks, eight incoming candidate-depth choices and an outgoing-action retention choice: 1,056 assignments. A selector uses two samples and evaluates on the other two, then exchanges the folds. Each arrival and price has 16 state–fold decisions and 32 held-out outcomes. Selection minimizes physical cost plus transmitted-bit price. Table 10 reports the operating price.

Table 10: Arrival-state and request-state decisions on eight panel states. Cost change: arrival minus request.

Arrival	Request bits	Arrival bits	Request successes	Arrival successes	Cost change
0	510.0	510.0	29	29	0
2	510.0	307.5	28	27	+0.031354
5	510.0	307.5	28	28	−0.000100
8	510.0	245.0	30	29	+0.031378

Across the three delayed ticks, request-state selection succeeds on 86 of 96 outcomes and arrival-state selection on 84. Communication falls by 39.7% at ticks 2 and 5 and by 52.0% at tick 8. The state-conditioned result includes gains and losses: Spatial task 4 loses one quarter of its matched successes, and Spatial task 5 gains one twelfth. This offline study isolates the effect of evaluating choices at the state reached during transmission. Native closed-loop latency is a separate measurement. Figure 21 resolves the aggregate of Table 10 by task.

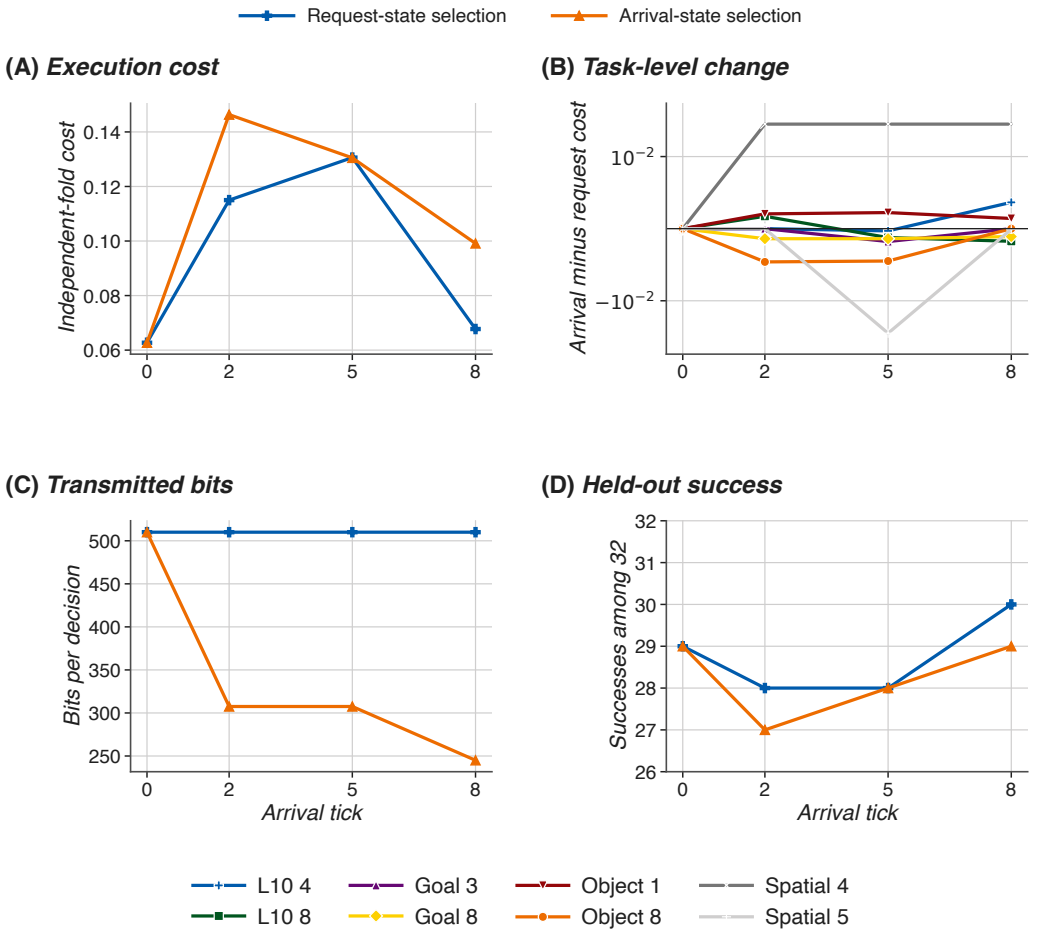

Figure 21: Arrival-state selection at ticks 2, 5 and 8. (A) Execution cost. (B) Task-level change. (C) Transmitted bits. (D) Held-out success.

K PAIRED-ROLLOUT ALLOCATION AND STREAMING EVALUATION

K.1 MATCHED SAMPLING PROTOCOL

Eight LaRA-VLA tasks at initial-state index 28 each provide four candidates at prefix and full depth. The logged execution-option identifiers are zero-based: $2i$ is candidate i's prefix and $2i + 1$ its full action. The pair set $(0, 1), (2, 3), (4, 5), (6, 7), (0, 2), (0, 4), (0, 6)$ compares each candidate's depths and each prefix with the first prefix. The partner map $(1, 0, 0, 2, 0, 4, 0, 6)$ is shared by all seven sampling rules. Each rule receives at most 224 trajectories per state. The error budget $1/20$ is divided equally across states and then across the seven bidirectional edges. Recommendations freeze at budget exhaustion and are evaluated on 16 fresh seeds per state, and Table 11 lists each rule's outcome.

LUCB completes 222 trajectories per state and the other rules complete 224, giving 12,528 acquisition trajectories. Each rule receives 664 token bytes. All runs exhaust their schedule with zero certificates. Independent execution uses 896 trajectories and gives 891 successes. The five failures occur at one task: PAIRGAP, sequential halving, UGapE and VarDE share one unsuccessful seed, and GRAD fails on a second seed.

Table 11: Sampling rules under a matched budget of 224 trajectories per state on eight LaRA-VLA states.

Rule	Succ. (%)	Rollouts	Bits
GRAD (Ménard, 2019)	99.2	224.0	5,312
LUCB (Kalyanakrishnan et al., 2012)	100.0	222.0	5,312
Sequential halving (Karnin et al., 2013)	99.2	224.0	5,312
Successive elimination (Even-Dar et al., 2006)	100.0	224.0	5,312
VarDE (Luong et al., 2026)	99.2	224.0	5,312
UGapE (Gabillon et al., 2012)	99.2	224.0	5,312
PAIRGAP	99.2	224.0	5,312

K.2 FRESH-SAMPLING CERTIFICATE UNDER THE UNIVERSAL SUPPORT

The matched panel reruns eight LaRA-VLA states, four candidates at two depths each, on up to 224 shared seeds per state. Recomputing the fresh-sampling certificate with time-uniform empirical-Bernstein paired bounds on all 12,528 of its trajectories, pooled over the seven sampling rules to 68 to 224 pairs per edge, certifies 0 of 8 states at $\epsilon = 0.01$. This matches Theorem 2: under $[-C, C]$ and 112 directional edges a zero-variance edge needs 12,885 empirical-Bernstein pairs. An envelope fitted to the observed differences is data-dependent and serves as a counterfactual. At that width 2 of 8 states would certify. The held-out replay of Appendix H.4 likewise reaches no fresh certificate at its budgets, and its certificate rule executes the first candidate.

K.3 REFERENCE CANDIDATE AND INTERVAL RESOLUTION

PAIRGAP and UGapE recommend the first prefix at all eight states. PAIRGAP spends 64–66 of 224 trajectories on this prefix; UGapE spends 109. Each edge receives 15–18 pairs under PAIRGAP. The first prefix has final bound 1.0099616, other prefixes have bound 1.01, and full actions have bound 1.0100384. Its roles as a frequent comparison endpoint, a received action and the lowest-index candidate are coupled in this design.

Table 12: Analytic pairs per edge for an all-zero difference sequence at $\epsilon = 0.01$ and $C = 1.01$.

Edges	Directional α	Planned-window	Empirical Bernstein	Zero-record
1	$1/40$	529	10,824	371
28	$1/1120$	878	12,532	706

The sampling panel uses the 65-point uniform grid with spacing $1/64$. Its zero-difference precision floor is 0.0315625 per direct edge, above the 0.01 decision tolerance. Table 12 uses the tolerance-

augmented 67-point grid; its 529 and 878 pairs describe that analytic configuration. The panel's resolution floor and the analytic sample counts quantify separate constraints on certification.

The ordered-grid scan behind Table 12 controls its failure probability as follows.

Proposition 12 (Betting failure probability)**.** *A two-ended ordered-grid scan with two unit-capital pivots, predictable solvent bets and budgets $\alpha_1, \alpha_2 \geq 0$ has failure probability at most $\alpha_1 + \alpha_2$ on a finite rational history tree, under the declared conditional-mean model.*

Proof. For a positive budget α, let M_t be a nonnegative supermartingale with $M_0 = 1$ on a finite history tree, and let $\tau = \inf\{t : M_t \geq 1/\alpha\}$. For any finite horizon T, iterating conditional expectations for the stopped process gives $\mathbb{E}[M_{\tau \wedge T}] \leq 1$. Nonnegativity implies $\Pr(\tau \leq T) \leq \alpha$. Applying this bound to the two true-mean bracketing pivots and taking a union bound gives failure probability at most $\alpha_1 + \alpha_2$. The ordered scan retains the true mean whenever both bracketing pivots remain unrejected. Integer wealth rounded downward preserves the supermartingale inequality. This argument assumes bounded observations, predictable bets, and the declared conditional mean at each sampled edge. A zero-budget side retains every pivot and has zero rejection probability. □

K.4 Observed pairing effects

A task-stratified panel uses 1,792 trajectories at eight states and has 18 terminal task failures. Its 896 pairs contain four shared-failure pairs and ten discordant pairs. Of 56 edges, 54 compare distinct actions and two compare exactly identical decoded actions. At a success-dominated state, the six distinct-action variance ratios range from 0.2537313 to 0.5983237. Two shared-failure edges have ratios 7.1×10^{-6} and 4.3×10^{-6}, reflecting nearly constant paired differences. All edges remain uncertified at 16 pairs. Exact-action controls have zero paired variance by construction.

L Depth and transmission mechanisms

L.1 Matched action-depth execution

The LaRA-VLA depth intervention fixes 16 states, four candidates per state and four continuation seeds. One, two, four and eight token rows, together with the projected raw action, give 1,280 logical outcomes. The prefix and full controls contribute 512 existing outcomes. The remaining versions add 484 physical trajectories and 284 exact-action reuses. Each reuse matches candidate, state, action and seed.

One, two, four and eight rows require 560, 720, 952 and 1,328 bits per candidate. Ten of 64 candidates equal their complete quantized action at two rows; 61 do so at four rows. Four-row and full execution preserve all 256 success labels. Executed steps agree in 246 outcomes, decrease in five and increase in five; the largest absolute cost change is 0.000142857, with identical mean cost. Two-row and full execution have 252 joint successes, two gains and two losses. The raw-action control has 252 joint successes with full depth, one gain, two losses and one joint failure, so the unquantized control succeeds in 253 of 256 continuations. These matched outcomes separate geometric reconstruction, task success and execution duration.

L.2 The constant row and a discrete gripper

The mechanism comparison holds the payload apart from the row count on the official protocol, ten tasks per suite and fifty initial states per task. One DC row, two rows and the quantized full code cost 560, 720 and 1,328 bits per call. The comparison carries its own quantized-full anchor from the same execution batch, so its contrasts stay inside one execution identity. Table 13 lists both comparisons.

On Goal the DC row wins 13 paired episodes against two rows and loses 9 ($P = 0.5235$), and against the quantized full code it wins 11 and loses 7 ($P = 0.4807$). On Spatial the DC row wins 3 and loses 8 against two rows ($P = 0.2266$). The comparison reports paired outcomes alone and issues no non-inferiority decision.

The discrete-gripper arm encodes the six continuous axes with two DCT rows and packs eight ternary gripper commands into sixteen bits, so its payload equals the two-row continuous stream. Each com-

Table 13: Row count and gripper representation at the official protocol.

Suite	Arm	Episodes	Success (%) ↑	Calls	Bits per call
Row count, ten tasks per suite					
Spatial	One row	500	97.8	6,730	560
Spatial	Two rows	500	98.8	6,669	720
Spatial	Quantized full	500	99.0	6,632	1,328
Goal	One row	500	97.8	7,221	560
Goal	Two rows	500	97.0	7,320	720
Goal	Quantized full	500	97.0	7,327	1,328
Discrete gripper at the two-row payload					
Spatial	Two rows, continuous gripper	400	99.0	5,323	720
Spatial	Two rows, discrete gripper	400	98.0	5,358	720
Goal	Two rows, continuous gripper	500	97.0	7,320	720
Goal	Two rows, discrete gripper	500	97.2	7,308	720

mand carries the sign of the quantized-full anchor's gripper value. Over the confirmation cohorts, which use 40 initial states per Spatial task and 50 per Goal task, the discretized arm reaches 97.2% on Goal against 97.0% for the continuous two-row stream, and 98.0% on Spatial against 99.0%. Changing the gripper representation leaves success and payload at their two-row levels.

L.3 Long-horizon every-chunk execution

The long-horizon suite repeats the every-chunk measurement over ten multi-stage tasks, fifty official initial states per task and 500 episodes per arm, under the same policy, seed and stopping rule as the other suites. The suite carries its own native, quantized-full, two-row and four-row arms, and its aggregator reports the registered two-point tolerance. Table 14 reports the four arms.

Table 14: Long-horizon every-chunk execution, 500 episodes per arm.

Arm	Success (%) ↑	Calls	Bits per call	Non-inferior to quantized full
Native full	93.8	16,840	1,328	No
Quantized full	92.2	17,148	1,328	Reference
Two rows	90.0	17,515	720	No ($P = 0.6641$)
Four rows	92.6	17,128	952	Yes ($P = 0.0020$)

Two rows win 14 paired episodes against the quantized full code and lose 25. Four rows win 3 and lose 1. The quantized full code falls below native execution by -1.6 points over 19 wins and 27 losses. This suite therefore separates quantization from truncation, and it moves the operating point from two rows to four rows as the task grows longer.

L.4 Physical value of an additional suffix

Across the same 256 candidate–seed pairs, full depth changes physical cost by a mean of -0.005261. State means range from -0.188625 to $+0.125674$. Cost decreases in 102 pairs, remains equal in 74 and increases in 80. Full execution adds four successes and loses two. This heterogeneity motivates state-conditioned depth selection, and Figure 22 resolves it by state.

L.5 Closed-loop depth accounting on OpenVLA-OFT

The closed-loop measurement charges every arrival at the price table above: 560, 720, 952 and 1,328 bits at one, two, four and eight received rows. OpenVLA-OFT generates an eight-step chunk per call; depth-eight execution consumes the whole chunk, and prefix-depth four executes the leading four steps and returns to the policy. Four matched task pairs at fifty official initial states give the totals in Table 15, and Figure 23 shows the arrival and episode costs. Three pairs have equal success counts at both depths. Goal task 0 succeeds on 40 of 50 prefix-depth-four episodes against 50 of 50

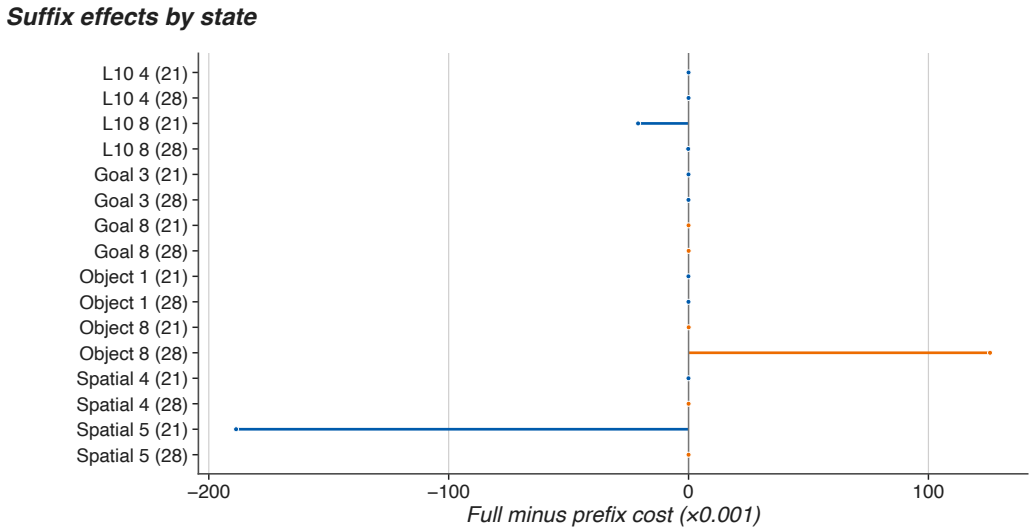

Figure 22: Suffix effects by state, as full-depth minus prefix cost over 256 candidate-seed pairs.

at depth eight, and its failed episodes run to the step limit, which raises its call ratio to 2.46 and its bit ratio to 1.76. Table 16 reports both suites at the official denominator.

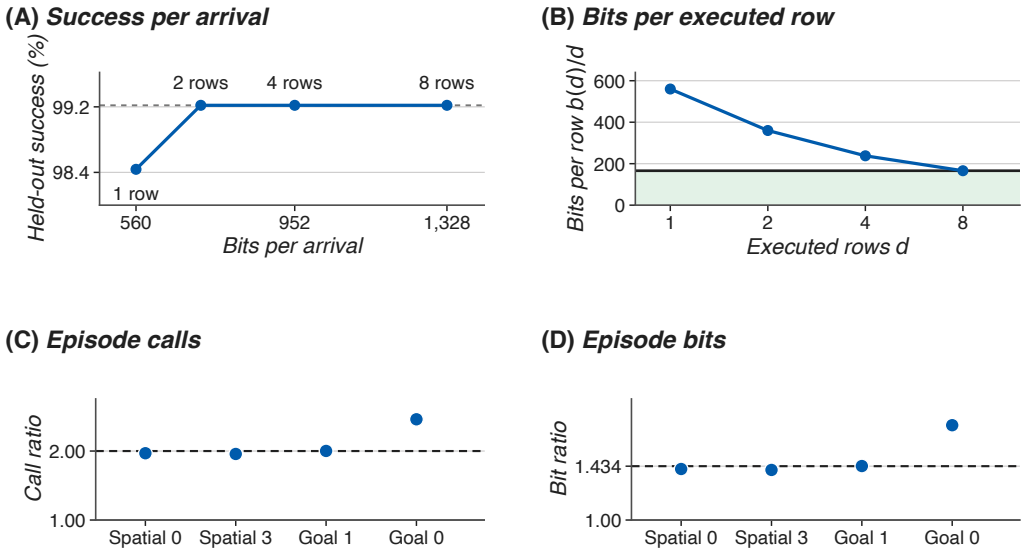

Figure 23: Arrival cost and episode cost. (A) Success per arrival on sixteen LaRA-VLA states. (B) Bits per executed row. (C) Episode calls and (D) episode bits of OpenVLA-OFT at four steps relative to eight.

Prefix-depth four changes success by −0.2 points on Spatial and −1.4 points on Goal over these 10 tasks per suite. Paired on the initial state, the prefix wins 7 and loses 8 Spatial episodes and wins 13 and loses 20 Goal episodes. The normal-approximation interval for a paired difference of proportions at level 0.95 runs from −1.7 to 1.3 points on Spatial and from −3.6 to 0.8 points on Goal, and both intervals contain zero.

Table 15: Matched OpenVLA-OFT episodes, 50 official initial states per task.

Suite and task	Depth	Successes	Episodes	Policy calls	Bits
Spatial 0	Eight rows	50	50	505	670,640
Spatial 0	Four rows	50	50	994	946,288
Spatial 3	Eight rows	50	50	544	722,432
Spatial 3	Four rows	50	50	1,065	1,013,880
Goal 1	Eight rows	47	50	651	864,528
Goal 1	Four rows	47	50	1,303	1,240,456
Goal 0	Eight rows	50	50	871	1,156,688
Goal 0	Four rows	40	50	2,143	2,040,136

Table 16: Official-protocol closed-loop depth on OpenVLA-OFT, 500 episodes per suite and mode.

Suite	Depth	Succ. (%)	Policy calls	Bits	Call ratio	Bit ratio
Spatial	Eight rows	98.4	6,652	8,833,856	1.00	1.00
Spatial	Four rows	98.2	13,164	12,532,128	1.98	1.42
Goal	Eight rows	97.0	7,295	9,687,760	1.00	1.00
Goal	Four rows	95.6	14,924	14,207,648	2.05	1.47

A control condition executes each chunk's trailing four steps in place of its leading four, holding the policy, the initial states and the arrival schedule fixed. It succeeds on 0 of 50 Spatial task 0 episodes at 2,750 policy calls and on 24 of 50 Goal task 1 episodes at 2,325 calls, against 505 and 651 calls at depth eight. Executing a suffix leaves the controller without the motion that the leading rows carry, so this condition is a scope boundary on the depth comparison and enters no prefix accounting.

L.6 Spectral ordering

In the LIBERO-10 panel, all 160 frozen ten-step candidates have two zero rows after quantization. Their eight-row references therefore reconstruct the full quantized action. Four candidates retain frequencies 0 and 2 at the two-row depth; the other candidates retain frequencies 0 and 1. Across the forty states, two rows produce fewer failures than the eight-row reference in eight states, equal counts in 27 and more in five, and the largest increase is two failures among 256 continuations. Reference failures concentrate in task 6, initial state 49, with 214 failures at eight rows and 216 at two.

Flex-π row energies are monotone; LaRA-VLA's eighth row exceeds its seventh by less than 0.02% of total energy. Prefixes of up to six rows retain the best cosine subset on these logged data.

A separate representation intervention fixes full candidates, quantization, projection and paid-row count. Across 18,432 legal streams and 448,512 decoded chunks, descending row energy reduces residual norm by 18.5%, 45.5% and 72.4% at 4, 8 and 16 rows. Mean serialized length differs by at most 0.021 bit between orders, and all orders recover the same complete 20-row action.

L.7 Additional per-task and codec measurements

The normalized squared subspace overlaps $\|C_{1:2}V_{1:2}\|_F^2/2$ are 0.992 for LaRA-VLA and 0.990 for Flex-π, where V contains the temporal second-moment eigenvectors. Across action dimensions, pooled lag-one autocorrelations range from 0.82 to 0.89 and from 0.88 to 0.91. Removing each chunk's mean gives ranges 0.43 to 0.56 and 0.61 to 0.71, respectively.

Under FAST+, a full LaRA-VLA chunk costs 101.4 bits on average and a two-row prefix costs 79.6, a reduction of 22%; the Flex-π reduction is 27%. LaRA-VLA's price per retained row decreases from 51.0 bits at one row to 22.6 at four rows, against the full-code value of 12.7. Two cosine rows contain 99.69% of LaRA-VLA energy and 99.31% of Flex-π energy, and the two leading Karhunen–Loève components contain 99.86% and 99.78%. The quantized full code has mean relative reconstruction error 3.7% on LaRA-VLA and 3.3% on Flex-π, and two rows raise these values to 4.4% and 5.6%. The largest two-row residual reaches 6.1% and 10.0% of the chunk norm, against quantized full-code maxima of 4.7% and 4.1%.

Figure 24 resolves the four-step execution cost by task. Table 18 records serialized prefix and zero-tail costs, and Table 17 gives the success and payload of every retained row count on both single-decision policies and on the LIBERO-10 panel.

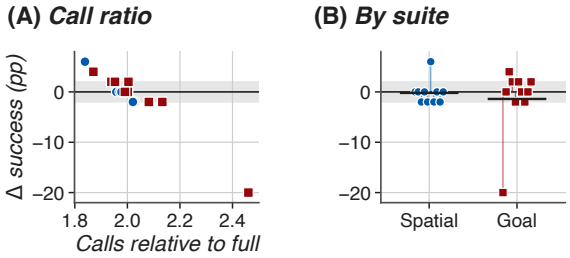

Figure 24: Four-step execution per task. (A) Call ratio and success change. (B) Success change by suite.

Table 17: Retained rows against success and payload.

Rows	Successes	Continuations	Success (%) ↑	Bits ↓
LaRA-VLA, 16 states				
1	252	256	98.44	**560**
2	254	256	**99.22**	720
4	254	256	**99.22**	952
8	254	256	**99.22**	1,328
Flex-π, 16 states				
1	255	256	**99.61**	**560**
10	253	256	98.83	1,560
Flex-π, LIBERO-10, 40 states				
2	9,854	10,240	**96.23**	**720**
8	9,824	10,240	95.94	1,328

Table 18: FAST+ prefix and zero-tail costs, and reconstruction error on LaRA-VLA chunks.

Rows	Prefix bits	Zero-tail bits	Error (%)
1	51.0	73.0	7.9
2	79.6	89.7	4.4
4	90.4	101.6	3.8
8	101.4	101.4	3.7

Table 19: Official initial-state IDs of the forty-state LIBERO-10 panel in Figure 5.

Task	S1	S2	S3	S4
0	10	15	30	44
1	1	17	18	37
2	23	25	29	42
3	13	21	23	38
4	4	22	24	36
5	14	15	25	45
6	8	17	20	49
7	0	3	14	49
8	13	21	43	49
9	0	5	11	35