Is Your Imitation Learning Policy Better than Mine? Policy Comparison with Near-Optimal Stopping

David Snyder[1,2], Asher James Hancock[2], Apurva Badithela[2], Emma Dixon[1], Patrick Miller[1], Rares Andrei Ambrus[1], Anirudha Majumdar[2], Masha Itkina[1], and Haruki Nishimura[1]
[1]Toyota Research Institute (TRI), [2]Princeton University
dasnyder@princeton.edu

***Abstract*—Imitation learning has enabled robots to perform complex, long-horizon tasks in challenging dexterous manipulation settings. As new policies are developed, they must be rigorously evaluated and compared against corresponding baselines through repeated evaluation trials, which is a costly procedure. This paper proposes a novel statistical framework for rigorously comparing two policies in the small sample size regime. Prior work in statistical policy comparison relies on batch testing, which requires a fixed, pre-determined number of trials and lacks flexibility in adapting the sample size to the observed evaluation data. Furthermore, extending the test with additional trials risks inducing inadvertent p-hacking, undermining statistical assurances. In contrast, our proposed statistical test is *sequential*, allowing researchers to decide whether or not to run more trials based on intermediate results. This adaptively tailors the number of trials to the difficulty of the underlying comparison, saving significant time and effort without sacrificing probabilistic correctness. Extensive numerical simulation and real-world robot manipulation experiments show that our test achieves near-optimal stopping, letting researchers stop evaluation and make a decision in a near-minimal number of trials while preserving the probabilistic correctness and statistical power of the comparison.**

I. Introduction

Reliable robot *policy evaluation* protocols are increasingly important in imitation learning as models and tasks grow in complexity, especially in dexterous manipulation where stochastic and contact-rich interactions introduce inherent randomness in outcomes. A particularly important aspect of evaluation is *policy comparison*, where two policies are repeatedly deployed in an environment to assess relative performance.

To motivate concretely, consider the scenario presented in Fig. 1 comparing a new policy π_1 to a baseline policy π_0 using a binary success / failure metric. While common in robotics research [6, 54, 39], this scenario introduces two challenges. Firstly, real-world evaluations are limited to a small number of trials (10–60) [33, 18, 14, 37, 25, 4]. Secondly, sequential evaluation results can fluctuate depending on when the testing ends. In the Fig. 1 example, the evaluator could observe more successes for π_0 after conducting additional trials, even though π_1 initially appeared superior after the first five.

Recent statistical policy comparison approaches [48, 26] use conventional batch testing, which requires pre-determining a fixed trial count number. Such methods prohibit the addition of new trials when the results of the initial batch are inconclusive, which otherwise would constitute p-hacking and invalidate any statistical guarantees [41].

To address these challenges, we propose a novel sequential testing framework named STEP (**S**equential **T**esting for **E**fficient **P**olicy Comparison) for rigorously comparing performance of imitation learning policies[1]. Unlike batch testing, STEP allows for variable trial numbers given an experimental budget which confers two distinct advantages: (1) early stopping when sufficient evidence exists, without sacrificing the probabilistic correctness of the comparison, and (2) reduced epistemic risk of overconfident (and potentially incorrect) conclusions when policies π_1 and π_0 are closely matched. We extensively demonstrate these advantages through simulation and real-world robot manipulation experiments. In simulation, STEP significantly outperforms state-of-the-art (SOTA) sequential methods, reducing the required number of trials by up to 32% without sacrificing probabilistic correctness.

II. Preliminaries

We assume a robot policy π_1 is trained to complete a task, and that a binary success-failure metric is used to evaluate performance. We assume *regularity*: in evaluation, the initial state s_0 and observation o_0 are drawn i.i.d. from the underlying distribution $\mathcal{D}_{s_0,o_0}$ of environments. The randomness over the draw of environment (and potentially in π_1) induces a Bernoulli distribution with mean p_1 (the true success rate) over observed evaluation rollouts. We denote $z_{1,n} = 1$ as success and $z_{1,n} = 0$ for failure on the n^{th} evaluation trial. Similarly, for a baseline (comparator) policy π_0, outcomes are $z_{0,n} \sim \text{Ber}(p_0)$. We pair the outcomes as $Z_n = (z_{0,n}, z_{1,n})$. The policy comparison problem can be formalized in the sense of Neyman-Pearson statistical testing [34]. The null hypothesis is that the novel policy π_1 is no better than the baseline π_0, and the alternative is that the novel policy is indeed better:

$$\begin{aligned} \text{Null Hypothesis } \mathbb{H}_0 &: \; p_1 \;\leq\; p_0 \equiv (p_0, p_1) \in \mathcal{H}_0 \\ \text{Alt. Hypothesis } \mathbb{H}_1 &: \; p_1 \;>\; p_0 \equiv (p_0, p_1) \in \mathcal{H}_1. \end{aligned} \tag{1}$$

We refer the reader to Supplement VIII-A for a review of the statistical testing nomenclature used in this work.

[1]Although this paper focuses on imitation learning, STEP is naturally applicable to evaluating any types of policies based on binary metrics, including reinforcement learning (RL) policies with sparse 0/1 reward. Please see the code and project website for more details.

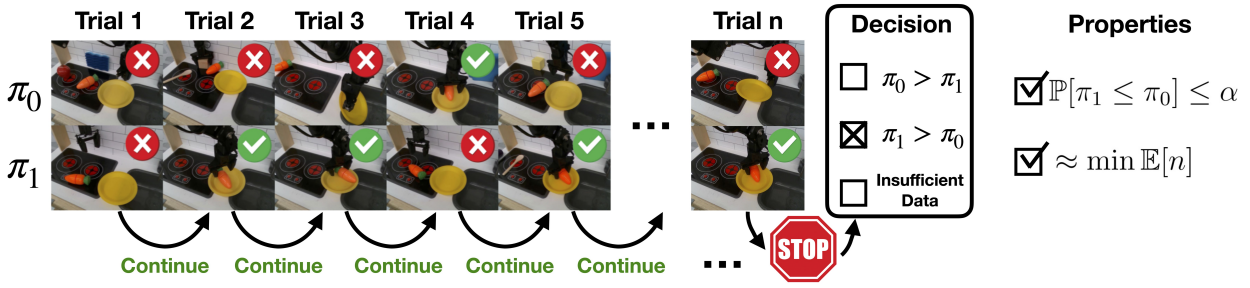


Fig. 1: Robot policy comparison problem under binary success/failure metrics. Novel policy π_1 is compared against baseline π_0 in a sequence of trials. Within a given evaluation budget, the evaluator seeks a statistically significant comparison in as few trials as possible. Allowing the evaluator to adaptively and near-optimally tailor the number of trials based on the data observed so far — *without compromising statistical assurances of the comparison* — is a central contribution of this work.

III. Problem Formulation

We assume that a robot evaluator is tasked with distinguishing two policies via successive evaluations, resulting in the testing paradigm described in Section II. We also assume that the evaluator has *pre-selected* the desired significance level α^* and a maximum number of trials (for each policy) that they are willing or able to run: $N_{\max}$. Note that valid tests must cap Type-I (i.e. false-positive) error at the pre-specified level $\alpha^* \in (0, 1)$; any procedure that fails to do so is inadmissible. Under this constraint, our goal is to synthesize a decision rule that maximizes statistical power (i.e. true positive rate) while minimizing the expected number of evaluation trials.

During the evaluation process the evaluator has access to the "filtration," a (possibly compressed) representation of the results collected *thus far* – represented by the state $x_n = F(Z_1, Z_2, ..., Z_n)$ – which reduces dimension using the sufficient statistics of scalar Bernoulli distributions [3] (here, the state is the empirical success count for each policy, augmented with the time). The evaluator must then sequentially decide whether to **Continue** (gather another trial for each policy) or stop (and either **AcceptNull** or **RejectNull**). Given a decision set $\mathcal{U} = \{\textbf{AcceptNull}, \textbf{Continue}, \textbf{RejectNull}\}$, this amounts to finding a state partition $\zeta = u(x)$ to optimally balance minimizing the expected sample size and maintaining high statistical power at the end of the evaluation process (denoted by $1-\beta_{N_{\max}}$), conditioned on the Type-I Error rate constraint:

$$\begin{aligned} \min_{\zeta:\mathcal{X}\mapsto\mathcal{U}} \quad & \mathbb{E}_{\mu(\mathcal{H}_1)}[n_{\text{stop}} + c\beta_{N_{\max}}] \\ \text{s.t.} \quad & \max_{h_0=(p_0,\, p_1)\in\mathcal{H}_0} \alpha(\zeta, h_0) \;\leq\; \alpha^* \\ & 0 \;\leq\; n_{\text{stop}} \;\leq\; N_{\max} \text{ w.p. } 1. \end{aligned} \tag{2}$$

IV. Methodology

The critical challenge is to find efficient decision regions to solve Eq. (2). For brevity we omit technical details and give an overview, along with pseudocode in Algorithm 1.

A. Decision Regions

Decision regions (state partitions) are represented as a sequence of ternary-valued sets:

$$\zeta \equiv \left\{ \mathcal{X}_n^{\text{Reject Null}}, \mathcal{X}_n^{\text{Accept Null}}, \mathcal{X}_n^{\text{Continue}} \right\}_{n=1}^{N_{\max}}. \tag{3}$$

Intuitively, larger rejection regions stop the evaluation sooner, but accumulate greater risk of Type-1 Errors. Controlling this error rate is critical to assure useful implementation.

B. Type-1 Error Control and Power Adaptivity

Suppose that some decision region for steps $n-1$ has been obtained with accumulated risk α_{n-1}. Bounding the Type-I Error for evaluation n by some $\alpha_n > \alpha_{n-1}$ conditioned on the preceding regions amounts to the following:

$$\max_{h\in\mathcal{H}_0} \mathbb{P}_h \left(\mathbf{x}_n \in \mathcal{X}_n^{\text{Reject Null}} \mid \mathcal{X}_{n-1}^{\text{Reject Null}} \right) \;\leq\; \alpha_n. \tag{4}$$

The dependence on the preceding rejection region is made explicit in Eq. (4), reflecting the internal dynamic structure under the null hypotheses; as the state represents the empirical mean, this structure is inherently local. To control Type-1 Error, it suffices to consider discrete "worst-case" nulls $\hat{\mathcal{H}}_0 = \{(p^{(1)}, p^{(1)}), \cdots, (p^{(M)}, p^{(M)})\}$, where $0 < p^{(1)} < \cdots < p^{(M)} < 1$ (see Supplement VIII-E2 for details).

Through discretization, Type-1 Error control becomes a linear inequality $P_n \mathbf{w}_n \;\leq\; \alpha_n \mathbb{1}$, where $\mathbf{w}_n$ represents the probability of rejecting the null in each state $\mathbf{x}_n \in \mathcal{X}_n = \{(0,0,n), (0,1,n), \cdots (N_{\max}, N_{\max}, n)\}$. P_n is a non-negative matrix of size $(M, |\mathcal{X}_n|)$; each row represents the probability of reaching each state under $h^{(i)} = (p^{(i)}, p^{(i)})$:

$$(P_n)_{ij} = \mathbb{P}_{h^{(i)}} \left(\mathbf{x}_n = \mathbf{x}_n^j \mid \mathcal{X}_{n-1}^{\text{Reject Null}} \right). \tag{5}$$

Given the previous rejection region, we can compute this probability by forward-propagating the previous state occupancy distribution $(P_{n-1})_i$ according to the state dynamics model.

Having controlled Type-1 Error at level α_n, all that remains is to choose the sequence $\{\alpha_n\}_{n=1}^{N_{max}}$. We introduce a non-negative scalar "risk budget" $f(n)$ for $n \in \{1, \cdots, N_{\max}\}$, which determines the maximum allowable Type-I Error under any null hypothesis at each step n. Constraining $\sum_{n=0}^{N_{\max}} f(n) = \alpha^*$ globally limits the Type-I error of the procedure by α^*. For all reported results, the budget is uniform: $f(n) = \alpha^*/N_{max}$.

C. Tractable Optimization

We now to solve a series of optimization problems to tractably construct the rejection regions, one for each n:

$$\begin{aligned} \max_{\|\mathbf{w}_n\|_\infty \le 1} \quad & \mathbb{1}^T \mathbf{w}_n \\ \text{s.t.} \quad & P_t \mathbf{w}_n \le \sum_{k=1}^{n} f(k) \mathbb{1} \\ & 0 \le \mathbf{w}_n \le 1, \end{aligned} \tag{6}$$

This objective encourages the rejection from as many states as possible, maximizing the size of $\mathcal{X}_n^{\text{Reject Null}}$. Furthermore, it implicitly rejects from states unlikely to occur under any null hypothesis, which are "cheaper" in terms of accruing risk. The first constraint ensures that the Type-I error is controlled up to time n as discussed in Section IV-B, while the second enforces boundedness of rejection probabilities in $[0, 1]$.

Algorithm 1 STEP Decision-Rule Synthesis

Input: $N_{\max} > 0$, risk budget function $f(n)$, type-I error limit $\alpha^* \in (0, 1)$, number of approximation points M
Initialize: $\zeta_0 = \emptyset$, $(P_0)_{ij} = 1$ if $(i, j) = (0, 0)$ else 0.
for n $\in \{1, ..., N_{\max}\}$ **do**
  $P_n \leftarrow$ **Propagate**$(P_{n-1}, \zeta_{n-1}, M)$ {Eq. (5)}
  $\mathbf{w}_n \leftarrow$ **Opt**(P_n, f) {Eq. (6)}
  $\zeta_n \leftarrow$ **Compress**$(\mathbf{w}_n)$
end for
return $\zeta = \{\zeta_1, \ldots, \zeta_{N_{\max}}\}$ {STEP policy}

V. Experiments

We conduct extensive simulation and real-world experiments to assess the sample efficiency of STEP in practical policy comparison settings. We consider the following baselines: i) SOTA sequential analysis methods [28, 29] (termed "Lai"), and ii) the method of Turner and Grünwald [45], which is specifically tailored to policy comparison problems (termed "SAVI"). Additionally, we include an Oracle Sequential Probability Ratio Test (SPRT) [50], infeasible to the evaluator, but included to give a conservative estimate of the optimality gap of each method. Additional information about each baseline is included in Supplement VIII-A and additional experimental details are presented in Supplement VIII-F.

A. Hardware Evaluation in the Large/Medium-Gap Regime

In this set of experiments, we compare policies with noticeable performance gaps to show our early-stopping capability. We consider the **CleanUpSpill** (Fig. 5b and Fig. 5c) task for a bimanual Franka Emika Panda robot. We trained single-task diffusion policies [14] using 150 human demonstrations. In addition to the RGB images, the policy receives the proprioceptive states as additional observations.

For evaluation, we compare the same imitation learning policy on two different distributions over initial conditions. The setting is similar to the one originally presented by Xu et al. [55], which compares a set of ID initial conditions against the out-of-distribution (OOD) initial conditions. The ID set includes 10 initial conditions with a white towel and a short blue mug whereas the OOD set uses 10 with a checkered towel with a tall cyan mug (each initial condition is repeated five times). As shown in Table I (rows 1–3), the empirical gap of 52 percentage points was detected in 7–14 trials by all methods, though they were tuned (where applicable) to an $N_{\max}$ up to thirty to seventy times larger; this demonstrates the significant reduction in sensitivity (from an evaluator's standpoint) arising from setting $N_{\max}$ versus choosing a batch size N. Furthermore, STEP's efficiency only minimally degrades when $N_{\max}$ is increased from 200 (row 2) to 500 (row 3). In this setting, any of the three sequential methods would have prevented the need for nearly 70 of the 100 total rollouts (35 of the 50 batch trials per policy).

We run a similar hardware procedure for the bimanual manipulation task **FoldRedTowel** (see Fig. 5a), where two policy checkpoints are compared on the same distribution of initial conditions. For brevity, this is deferred to Supplement VIII-F1.

B. Hardware Evaluation in the Small-Gap Regime

In addition, we run a separate hardware evaluation on the task **CarrotOnPlate**. This setting compares two distinct and closely-competing policies to characterize necessary sample sizes for statistical validity. See Supplement VIII-F2 for more details. We find that no statistical tests (including the Oracle SPRT) is able to detect significant difference between the two policies with $N_{\max} = 100$ despite the empirical performance gap of nine percentage points (Table I, row 4). In fact, further analysis in Supplement VIII-F3 suggests that we would have to perform 500 trials (apiece) to reliably reach a decision with statistical confidence. This sample size is an order of magnitude larger than the current norms, reflecting fundamental yet often overlooked challenges in trustworthy policy comparison.

C. Multi-Task Evaluation in SimplerEnv Simulation

Finally, we consider the problem of multi-task and multi-policy extensions to this framework, and illustrate via an example of policy evaluation in simulation (where costs of evaluation can still be significant). Concretely, Octo-Small (π_1) and Octo-Base (π_0) [37] are compared in the SimplerEnv [30] simulation environment on three tasks: **SpoonOnTowel**, **EggplantInBasket**, and **StackCube** (Table I, rows 5–7). The empirical success rates we observed are consistent with the findings of Li et al. [30] (Table V) that Octo-Small is more performant on these tasks. We seek a multitask comparison: for $p_s^{[\tau]}$ denoting the performance of Octo-Small and $p_b^{[\tau]}$ the performance of Octo-Base on task τ, we test:

$$\begin{aligned} \mathbb{H}_0 &: \exists \tau \in \{1, 2, 3\}\ p_s^{[\tau]} \le p_b^{[\tau]} \\ \mathbb{H}_1 &: \forall \tau \in \{1, 2, 3\}\ p_s^{[\tau]} > p_b^{[\tau]}. \end{aligned} \tag{7}$$

Many sophisticated methods exist to efficiently run multi-hypothesis testing (in this case, we are essentially evaluating three separate hypotheses, one for each task[2]). We use the stan-

[2]Note that multi-hypothesis testing can naturally handle the case of **multi-policy** comparison as well, where we would reduce the test to a set of pairwise policy comparisons which are examined simultaneously.

Task	Type	α^*	N_{max}	N	$\hat{p}_0$	$\hat{p}_1$	SAVI	Lai	STEP (Ours)	SPRT***
CleanUpSpill	$\mathcal{D}^i_{s_0,o_0}$	0.05	50	50	0.280	0.800	**7**	**8**	**8**	7
CleanUpSpill	$\mathcal{D}^i_{s_0,o_0}$	0.05	200	50	0.280	0.800	**7**	**13**	**9**	7
CleanUpSpill	$\mathcal{D}^i_{s_0,o_0}$	0.05	500	50	0.280	0.800	**7**	**14**	**13**	7
CarrotOnPlate	π_i	0.05	100	100	0.680	0.760	–	–	–	–
SpoonOnTowel	π_i	0.01	500	500	0.084	0.386	**33**	**36**	**36**	26
EggplantInBasket	π_i	0.01	500	500	0.400	0.564	192	**125**	**131**	128
StackCube	π_i	0.01	500	500	0.000	0.030	329	417	225	135
Multitask	π_i	0.03	1500	1500	N/A	N/A	554	578	392	289

TABLE I: Empirical time-to-correct-decision for subset of **hardware (top)** and **simulation (bottom)** policy comparisons (see Table II for counterfactual results for all experimental settings). The comparison type is described first; π_i is comparing two policies, while $\mathcal{D}^i_{s_0,o_0}$ compares one policy under possible distribution shift. The utilized Type-I Error α^* and N_{max} describe the constraints applied *a priori* by the evaluator (we underline to emphasize the change in N_{max} for rows 1-3; observe that the sensitivity of the stopping times is very small). $N \leq N_{\text{max}}$ represents the amount of data available for the statistical analysis (i.e., the data that was actually collected). We report the terminal empirical success rates (after N trials) of each policy in each setting under $\hat{p}_i$ (this information is not available to any feasible algorithm). Because these are real experiments, we do not have truth labels. However, in all cases, every method arrived at the same decision, including the Oracle SPRT which has *a priori* access to $(\hat{p}_0, \hat{p}_1)_N$. This decision was Reject Null for all rows except the CarrotOnPlate task, which returned Fail To Decide. We report the stopping times of all methods on the right of the table for every context; in all cases: lower is better. We put in **bold** any *feasible* method result that is near-optimal within ten trials (absolute) or 25% (relative) of the SPRT Oracle, which is *not implementable* by an evaluator. In the Multitask setting, we test $p_1 > p_0$ *uniformly across the preceding three tasks*. This stopping time is the sum by column of the stopping times for the three tasks. Our method saves the evaluator over 160 trials in uniform certification over these three tasks as compared to either feasible baseline.

dard Bonferroni (union bound) correction [15] to evaluate our test at $\alpha = 0.03$ (each task is at $\alpha = 0.01$), observing the stopping times shown in the table. Notably, each sequential method saves a substantial number of simulation rollouts on the easiest comparison (**SpoonOnTowel**). SAVI begins to struggle when the tests become more challenging (**EggplantInBasket**), and Lai struggles in heavily skewed cases where success rates are close to zero (**StackCube**). To summarize: naive multitask evaluation requires the aggregation of multiple batches of rollouts, here totaling 500 per task per policy. On the easiest task, even when tuned to $N_{\text{max}} = 500$, the comparison was answered in fewer than 40 rollouts by all sequential methods, a savings of over 90%. On the progressively harder cases the number of required samples increased 5-10 times over the easiest, but our method (STEP) improved substantially over each of the other sequential procedures. In total, STEP would have saved the evaluator an additional 160 rollouts for each of Octo-small and Octo-base for the multitask comparison problem as compared to the current SOTA approaches.

VI. Real-World Deployability and Generalization Discussion

STEP can be deployed as a wrapper for essentially any evaluation pipeline, conditioned on the use of a binary success metric (extensions to partial credit will be considered in future work). With relatively light offline pre-computation to synthesize (reusable) STEP decision rules (see Footnote 1), the decision process is nearly instantaneous to evaluate. Further, it can be used for both hardware and simulation evaluation, even beyond imitation learning or nominally robotic contexts, as it builds on general statistical methods which apply to testing for medical, polling, and quality assurance applications [38, 40]. A lingering practical consideration is the specification of the risk budget; we show in this work that a simple choice (uniform) works quite well, but in general the optimal budget shape design depends on the belief of the evaluator as to the measure over plausible null and alternative hypotheses.

VII. Conclusion

We present STEP, a novel sequential statistical method to rigorously compare performance of imitation learning policies through a series of evaluation trials. STEP provides flexibility in adapting the number of necessary trials to the underlying difficulty of the comparison problem. This leads to low sample complexity in cases where one policy clearly outperforms the other while avoiding overconfident and potentially incorrect evaluation decisions when the policies are closely competing. We show that STEP near-optimally minimizes the expected number of required trials. Furthermore, STEP matches or exceeds the performance of state-of-the-art baselines across a wide swath of practical evaluation scenarios in numerical and robotic simulation and on numerous physical hardware demonstrations. These results highlight the practical utility of STEP as a versatile statistical analysis tool for policy comparison, contributing to the foundation of robot learning as an empirical science.

References

[1] Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 29304–29320, 2021. doi: 10.48550/arXiv.2108.13264.

[2] G. A. Barnard. Significance Tests for 2×2 Tables. *Biometrika*, 34(1-2):123–138, January 1947. ISSN 0006-3444. doi: 10.1093/biomet/34.1-2.123.

[3] Peter J. Bickel and Kjell A. Doksum. *Mathematical Statistics: Basic Ideas and Selected Topics, Volumes I-II Package*. Chapman and Hall/CRC, New York, December 2015. ISBN 978-1-315-36926-6. doi: 10.1201/9781315369266.

[4] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint arXiv:2410.24164*, 2024. doi: 10.48550/arXiv.2410.24164.

[5] R. D. Boschloo. Raised conditional level of significance for the 2 × 2-table when testing the equality of two probabilities. *Statistica Neerlandica*, 24(1):1–9, 1970. doi: 10.1111/j.1467-9574.1970.tb00104.x.

[6] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023. doi: 10.48550/arXiv.2307.15818.

[7] Luis A. Caffarelli and S. Salsa. *A Geometric Approach to Free Boundary Problems*. American Mathematical Soc., 2005. ISBN 978-0-8218-3784-9. Google-Books-ID: YOzpBwAAQBAJ.

[8] Hock Peng Chan and Tze Leung Lai. Asymptotic Approximations for Error Probabilities of Sequential or Fixed Sample Size Tests in Exponential Families. *The Annals of Statistics*, 28(6):1638–1669, 2000. ISSN 0090-5364. Publisher: Institute of Mathematical Statistics.

[9] Herman Chernoff. Sequential Tests for the Mean of a Normal Distribution. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, volume 4.1, pages 79–92. University of California Press, January 1961.

[10] Herman Chernoff. Sequential Test for the Mean of a Normal Distribution III (Small t). *The Annals of Mathematical Statistics*, 36(1):28–54, 1965. ISSN 0003-4851. Publisher: Institute of Mathematical Statistics.

[11] Herman Chernoff. Sequential Tests for the Mean of a Normal Distribution IV (Discrete Case). *The Annals of Mathematical Statistics*, 36(1):55–68, 1965. ISSN 0003-4851. Publisher: Institute of Mathematical Statistics.

[12] Herman Chernoff. Optimal Stochastic Control. *Sankhyā: The Indian Journal of Statistics, Series A (1961-2002)*, 30(3):221–252, 1968. ISSN 0581-572X. URL https://www.jstor.org/stable/25041355. Publisher: Springer.

[13] Herman Chernoff and A. John Petkau. Numerical Solutions for Bayes Sequential Decision Problems. *SIAM Journal on Scientific and Statistical Computing*, 7(1):46–59, 1986. doi: 10.1137/0907003.

[14] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 0, 2024. doi: 10.48550/arXiv.2303.04137.

[15] Olive Jean Dunn. Multiple Comparisons among Means. *Journal of the American Statistical Association*, 56(293):52–64, 1961. doi: 10.1080/01621459.1961.10482090.

[16] Michael Fauss, Abdelhak M. Zoubir, and H. Vincent Poor. Minimax Optimal Sequential Hypothesis Tests for Markov Processes. *The Annals of Statistics*, 48(5):2599–2621, 2020. ISSN 0090-5364. Publisher: Institute of Mathematical Statistics.

[17] R. A. Fisher. On the interpretation of χ^2 from contingency tables, and the calculation of p. *Journal of the Royal Statistical Society*, 85(1):87–94, 1922. ISSN 09528385. URL http://www.jstor.org/stable/2340521.

[18] Pete Florence, Corey Lynch, Andy Zeng, Oscar A Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit behavioral cloning. In *Conference on Robot Learning (CoRL)*, pages 158–168. PMLR, 2022. doi: 10.48550/arXiv.2109.00137.

[19] Robert Fortus. Approximations to Bayesian Sequential Tests of Composite Hypotheses. *The Annals of Statistics*, 7(3):579 – 591, 1979. doi: 10.1214/aos/1176344679.

[20] Sander Greenland, Stephen J Senn, Kenneth J Rothman, John B Carlin, Charles Poole, Steven N Goodman, and Douglas G Altman. Statistical tests, P values, confidence intervals, and power: a guide to misinterpretations. *European Journal of Epidemiology*, 31(4):337–350, 2016.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. doi: 10.48550/arXiv.1512.03385.

[22] Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, April 2021. ISSN 0090-5364, 2168-8966. doi: 10.1214/20-AOS1991. Publisher: Institute of Mathematical Statistics.

[23] Samuel Karlin and Herman Rubin. The Theory of Decision Procedures for Distributions with Monotone Likelihood Ratio. *The Annals of Mathematical Statistics*, 27(2):272–299, June 1956. ISSN 0003-4851, 2168-8990. doi: 10.1214/aoms/1177728259. Publisher: Institute of Mathematical Statistics.

[24] J. Kiefer and Lionel Weiss. Some Properties of Gener-

alized Sequential Probability Ratio Tests. *The Annals of Mathematical Statistics*, 28(1):57–74, 1957. ISSN 0003-4851. Publisher: Institute of Mathematical Statistics.

[25] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. OpenVLA: An Open-Source Vision-Language-Action Model. *arXiv preprint arXiv:2406.09246*, 2024. doi: 10.48550/arXiv.2406.09246.

[26] Hadas Kress-Gazit, Kunimatsu Hashimoto, Naveen Kuppuswamy, Paarth Shah, Phoebe Horgan, Gordon Richardson, Siyuan Feng, and Benjamin Burchfiel. Robot Learning as an Empirical Science: Best Practices for Policy Evaluation. In *Robotics: Science and Systems*, 2024. doi: 10.48550/arXiv.2409.09491.

[27] Tze Leung Lai. Power-One Tests Based on Sample Sums. *The Annals of Statistics*, 5(5):866 – 880, 1977. doi: 10.1214/aos/1176343943.

[28] Tze Leung Lai. Nearly Optimal Sequential Tests of Composite Hypotheses. *The Annals of Statistics*, 16(2):856–886, 1988. ISSN 0090-5364. Publisher: Institute of Mathematical Statistics.

[29] Tze Leung Lai and Li Min Zhang. Nearly Optimal Generalized Sequential Likelihood Ratio Tests in Multivariate Exponential Families. *Lecture Notes-Monograph Series*, 24:331–346, 1994. ISSN 0749-2170. Publisher: Institute of Mathematical Statistics.

[30] Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, et al. Evaluating Real-World Robot Manipulation Policies in Simulation. *arXiv preprint arXiv:2405.05941*, 2024. doi: 10.48550/arXiv.2405.05941.

[31] Gary Lorden. Nearly-Optimal Sequential Tests for Finitely Many Parameter Values. *The Annals of Statistics*, 5(1):1–21, 1977. ISSN 0090-5364. Publisher: Institute of Mathematical Statistics.

[32] I Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. doi: 10.48550/arXiv.1711.05101.

[33] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning (CoRL)*, volume 164, pages 1678–1690. PMLR, 2021. doi: 10.48550/arXiv.2108.03298.

[34] Jerzy Neyman, Egon Sharpe Pearson, and Karl Pearson. IX. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231(694-706):289–337, January 1997. doi: 10.1098/rsta.1933.0009. Publisher: Royal Society.

[35] Andrei Novikov and Fahil Farkhshatov. A computational approach to the Kiefer-Weiss problem for sampling from a Bernoulli population. *Sequential Analysis*, 41(2):198–219, 2022. doi: 10.1080/07474946.2022.2070212.

[36] Andrey Novikov. A numerical approach to sequential multi-hypothesis testing for Bernoulli model. *Sequential Analysis*, 42(3):303–322, 2023. doi: 10.1080/07474946.2023.2215825.

[37] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Charles Xu, Jianlan Luo, Tobias Kreiman, You Liang Tan, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An Open-Source Generalist Robot Policy. In *Robotics: Science and Systems (RSS)*, 2024. doi: 10.48550/arXiv.2405.12213.

[38] Fredrik Ohrn and Christopher Jennison. Optimal group-sequential designs for simultaneous testing of superiority and non-inferiority. *Statistics in Medicine*, 29(7-8):743–759, March 2010. ISSN 1097-0258. doi: 10.1002/sim.3790.

[39] Abby O'Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, et al. Open x-embodiment: Robotic learning datasets and RT-X models. *arXiv preprint arXiv:2310.08864*, 2023. doi: 10.48550/arXiv.2310.08864.

[40] E. S. PAGE. Continuous inspection schemes. *Biometrika*, 41(1-2):100–115, 06 1954. ISSN 0006-3444. doi: 10.1093/biomet/41.1-2.100. URL https://doi.org/10.1093/biomet/41.1-2.100.

[41] Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-Theoretic Statistics and Safe Anytime-Valid Inference. *Statistical Science*, 38(4):576–601, November 2023. ISSN 0883-4237, 2168-8745. doi: 10.1214/23-STS894. Publisher: Institute of Mathematical Statistics.

[42] H. Robbins and D. Siegmund. The Expected Sample Size of Some Tests of Power One. *The Annals of Statistics*, 2(3):415 – 436, 1974. doi: 10.1214/aos/1176342704.

[43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[44] Gideon Schwarz. Asymptotic Shapes of Bayes Sequential Testing Regions. *The Annals of Mathematical Statistics*, 33(1):224 – 236, 1962. doi: 10.1214/aoms/1177704726.

[45] Rosanne J. Turner and Peter D. Grünwald. Exact anytime-valid confidence intervals for contingency tables and beyond. *Statistics & Probability Letters*, 198:109835, July 2023. ISSN 0167-7152. doi: 10.1016/j.spl.2023.109835. URL https://www.sciencedirect.com/science/article/pii/S0167715223000597.

[46] Pierre Van Moerbeke. Optimal Stopping and Free Boundary Problems. *The Rocky Mountain Journal of*

Mathematics, 4(3):539–578, 1974. ISSN 0035-7596. Publisher: Rocky Mountain Mathematics Consortium.

[47] Jean Ville. *Etude Critique de la Notion de Collectif*. PhD thesis, Universite de Paris, 1939. Publisher:Gauthier-Villars, Paris.

[48] Joseph A. Vincent, Haruki Nishimura, Masha Itkina, Paarth Shah, Mac Schwager, and Thomas Kollar. How Generalizable is My Behavior Cloning Policy? A Statistical Approach to Trustworthy Performance Evaluation. *IEEE Robotics and Automation Letters*, 9(10):8619–8626, 2024. doi: 10.1109/LRA.2024.3445635.

[49] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[50] A. Wald. Sequential Tests of Statistical Hypotheses. *The Annals of Mathematical Statistics*, 16(2):117–186, 1945. ISSN 0003-4851. Publisher: Institute of Mathematical Statistics.

[51] A. Wald and J. Wolfowitz. Optimum Character of the Sequential Probability Ratio Test. *The Annals of Mathematical Statistics*, 19(3):326–339, 1948. ISSN 0003-4851. Publisher: Institute of Mathematical Statistics.

[52] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning (CoRL)*, 2023. doi: 10.48550/arXiv.2308.12952.

[53] David Williams. *Probability with Martingales*. Cambridge University Press, 1991. doi: 10.1017/CBO9780511813658.

[54] Xuan Xiao, Jiahang Liu, Zhipeng Wang, Yanmin Zhou, Yong Qi, Qian Cheng, Bin He, and Shuo Jiang. Robot learning in the era of foundation models: A survey. *arXiv preprint arXiv:2311.14379*, 2023. doi: 10.48550/arXiv.2311.14379.

[55] Chen Xu, Tony Khuong Nguyen, Emma Dixon, Christopher Rodriguez, Patrick Miller, Robert Lee, Paarth Shah, Rares Ambrus, Haruki Nishimura, and Masha Itkina. Can We Detect Failures Without Failure Data? Uncertainty-Aware Runtime Failure Detection for Imitation Learning Policies. *arXiv preprint arXiv:2503.08558*, 2025.

VIII. SUPPLEMENT

A. Related Work

This section provides an extensive review of the statistics literature to highlight the significance of our approach.

1) Statistical Testing and Policy Evaluation: The Neyman-Pearson statistical testing paradigm [34] forms the foundation of frequentist statistical decision theory. These methods have been applied in robotics for predictable policy *characterization*[3] [48, 1] in the batch regime. The Neyman-Pearson Lemma [34] and the Karlin-Rubin Theorem [23] give sufficient conditions for maximal power batch tests. Specific methods have been developed for two-factor comparison-type problems in the context of 2x2 contingency tables. Of the tests by Fisher [17], Boschloo [5], and Barnard [2], the last is most relevant here; however, while it has strong power in the batch setting, it does give insight into selecting the size of the batch *a priori*.

2) Sequential Statistical Evaluation Methods: The difficulty in choosing the appropriate batch size motivates the sequential testing framework set out in Wald [50], which is adopted in this paper (see Section III). Wald and Wolfowitz [51] showed that in the simple-vs-simple setting, the sequential probability ratio test (SPRT)[4] minimizes the expected number of samples among all tests that control Type-I and Type-II error, extending the Neyman result. The composite regime is more complicated; minimax results limit the worst-case expected sample size [24, 31, 16], more generally, the expected sample size must be minimized under a mixture over the alternatives [44, 19]. Lai [28] reconciled this Bayesian interpretation with the frequentist developments of Chernoff [9, 10, 11].

Optimal Stopping-Based Methods. The direct approach to synthesizing near-optimal decision regions in the composite-vs-composite regime relies on developments in the theory of martingales and optional stopping [53]. Van Moerbeke [46] reduces the statistical testing problem to optimal stopping and demonstrates the equivalence of the solution with the solution to a related Stefan-type free-boundary partial differential equation (PDE) [7], building on results in Chernoff [12]. Unfortunately, the mapping to the PDE parameterization is implicit and difficult to specify under composite null hypotheses, rendering this method less practical. Asymptotic approximations of the free-boundary problem can be more profitably used to construct near-optimal tests. Lai [28] solves for a near-optimal procedure in the univariate composite-vs-composite setting and [29, 8] extends this to the multivariate setting. However, while useful for proving optimal rates, the latter methods suffer in the finite-$N_{\max}$ regime.

Safe, Anytime-Valid Inference (SAVI) Methods. Utilizing Ville's Inequality (a sequential generalization of Markov's Inequality) [47], SAVI methods construct a probability ratio test that enforces Type-I error control uniformly in time [22, 41]; this is tailored to contingency tables in [45]. These methods also suffer in the small-sample regime due to the power-1 nature of the resulting tests [42, 27, 41].

3) Numerical Implementations: The SciPy [49] package contains many batch procedures; recently, optimal binomial confidence intervals were constructed [48]. However, numerical methods for sequential analysis are quite limited and focus on the simple-null or univariate settings [13, 35, 36, 16]. To our knowledge, this is the first composite-vs-composite implementation for policy comparison-type problems.

B. Full Details of Real-World Robot Experiments

All of our real-world hardware tasks are visualized in Fig. 5. In **FoldRedTowel**, the robot first observes an unfolded red towel placed in random poses. The task is considered a success if the robot folds the towel twice and then moves the folded towel to a corner of the table. In **CleanUpSpill**, a mug is initially lying sideways on the table and a coffee spill exists near the mug. The task is successful if one arm puts the mug upright while the other arm picks up a white towel and wipes the spill. In both tasks, a total of four RGB cameras observe the Franka robot and the objects, where two monocular cameras are mounted on the table top and a stereo wrist camera on each of the arms. We trained single-task diffusion policies [14] on each task, with 300 human demonstrations for **FoldRedTowel** and 150 for **CleanUpSpill**, respectively. In addition to the RGB images, the policy receives the proprioceptive states as additional observations. Following [14], the image observations are passed to the ResNet-18 [21] encoder before fed into the U-Net [43] diffusion policy architecture. $T_o = 2$ observations are stacked and fed into the policy network to predict $T_p = 16$ steps of actions. The actions are re-planned after $T_a = 8$ actions are executed.

For the **CarrotOnPlate** task, an experiment is recorded as a success if the robot policy succeeds in placing the carrot on the plate within the max episode count without: i) pushing the carrot off the counter, ii) colliding with the back wall, iii) pushing the plate into the sink, and iv) accumulating a total of 3 cm of negative z commands when the end-effector is in contact with the table surface. For Octo evaluations, we use an action-chunking horizon of 2.

In all the experiments, we take the effort to mitigate distribution shift during trials, such as a change in lighting conditions. We also randomize the order of trials so that any distribution shift due to other factors (e.g., hardware degradation over the course of trials) is equally reflected in all the settings. Where applicable, we also separate the role of the evaluator from the demonstrator of the tasks for training. These practices are adopted from [26] to reduce unintended variability in environmental conditions during policy evaluation.

C. Additional Numerical Simulation Results

We plot numerical Monto-Carlo validation results for key properties of sequential procedures – type-1 error control, statistical power, and stoppping time – for $N_{\max} = 100$ and $\alpha^* = 0.05$ in Fig. 2, Fig. 3, and Fig. 4. We include for this case the power profile for a Barnard Test that is validly

[3] As a simple example, one can accurately predict *a priori* that for estimating Ber(p) with $\hat{p} \in [0.25, 0.75]$ and $N \geq 36$, a 95% confidence interval for p will be approximately $\hat{p} \pm \frac{1}{\sqrt{N}}$.

[4] This is the form of the SPRT Oracle method.

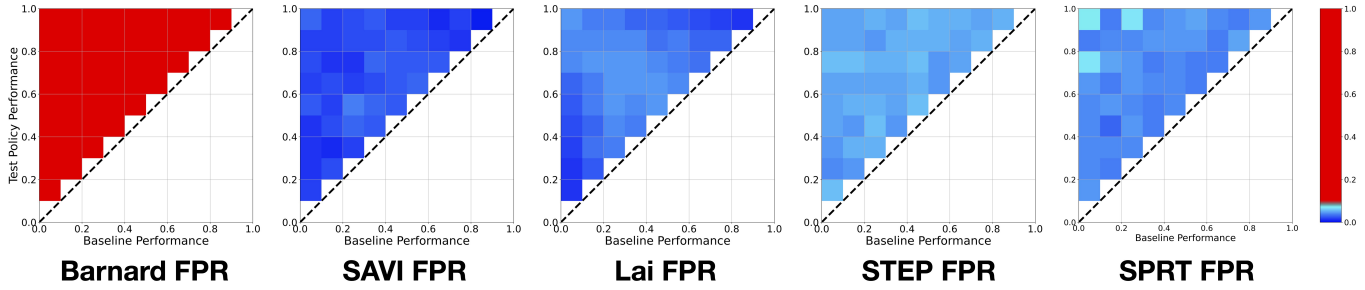


Fig. 2: False positive rate of four feasible methods (Barnard, SAVI, Lai, and Ours (STEP)) and the SPRT (Oracle method) for 1000 simulated trajectories on each of 45 alternatives (squares in color); $N_{\max} = 100$ and $\alpha^* = 0.05$. Note that naively utilizing a batch method in sequence leads to violation of Type-1 Error control (Barnard). Additionally, note that SAVI and Lai struggle to utilize the full risk budget in finite $N_{\max}$ (darker blue regions).

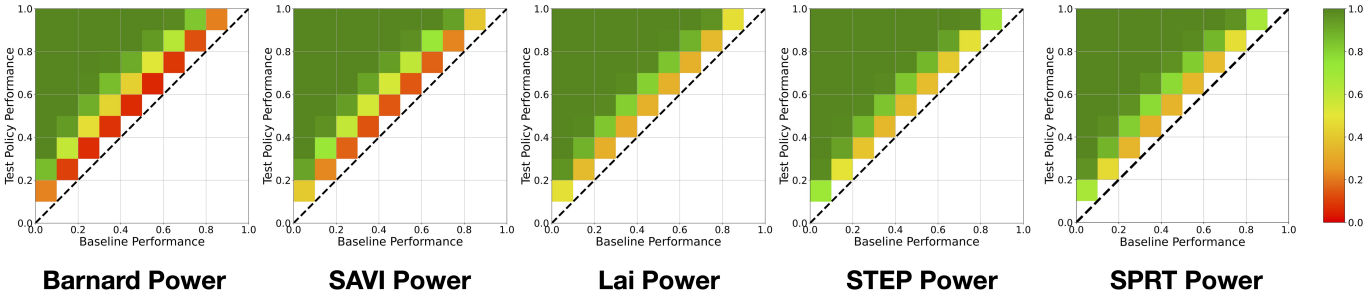


Fig. 3: Terminal power of four feasible methods (Barnard, SAVI, Lai, and Ours (STEP)) and the SPRT (Oracle method) for 5000 simulated trajectories on each of 45 alternatives (squares in color); $N_{\max} = 100$ and $\alpha^* = 0.05$. Because $N_{\max}$ is small, the terminal power is generally low for gaps less than 20 percentage points. Moving from left to right: sequentializing Barnard's Test is inefficient due to a loss of structure; SAVI methods also suffer when p_0 and p_1 are closely competing, due to the method inherently generalizing to arbitrary $N_{\max}$. The Lai procedure and our STEP are similar to the SPRT oracle; however, note that Lai struggles more at the extremes (bottom left and top right). This inefficiency in the skewed regime becomes more pronounced as N grows and the gaps shrink.

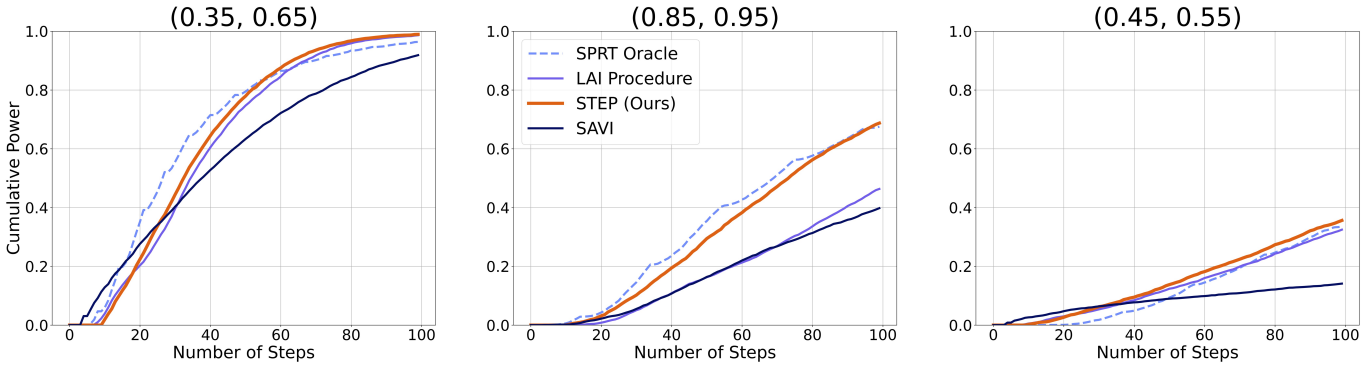


Fig. 4: Cumulative power of all feasible methods (Lai, SAVI, STEP (Ours)) and SPRT Oracle over 5000 trajectories in three evaluation settings of increasing difficulty; (p_0, p_1) for each setting title the respective figures. $N_{\max} = 100$ and $\alpha^* = 0.05$. The expected time-to-decision is the integral of the area *above* the cumulative power curve; therefore, curves higher and to the left are better. (Left) For a gap of 30 percentage points, all methods demonstrate similar stopping times. (Center) For a gap of 10 percentage points in the low-variance regime (i.e., farther from 0.5), STEP significantly outperforms the Lai and SAVI procedures. (Right) For a gap of 10 percentage points in the high-variance regime, STEP and Lai are similar but again SAVI struggles and underperforms the other methods.

sequentialized using Bonferroni's correction; *this rectifies the Type-1 Error violation in Fig. 2*. In so doing, it loses significant power and fails to meaningfully compete with the SOTA sequential procedures. In addition to inefficient computational properties, the Bonferroni-correct Barnard procedure becomes even weaker for larger $N_{\max}$.

A key point of emphasis in the $N_{\max} = 100$ regime is the low power of all tests for gaps of approximately 10 percentage points and smaller. Notably, no procedure has power over 50% in the hardest regimes (see the orange regions of every method in Fig. 3). A small amount of this is due to the sequential procedure; however, a significant amount reflects fundamental uncertainty (variance in outcomes) present for small sample sizes in evaluation. The implication of this is the need for significant increases in evaluation trials in order to effect meaningful comparisons when the underlying gap is small. This will be considered further in the context of the **CarrotOnPlate** hardware experiments (Section V) in Supplement VIII-F3 below.

Finally, we note the presence of a small hint to the weakness of the Lai procedure in skewed settings. Note that in the bottem left and top right of the Lai panel of Fig. 3, the power significantly lags STEP and SPRT; in a similar vein, note the regions of darker blue in the Lai procedure panel of Fig. 2. These reflect an inherent inefficiency undergirding Lai method, which directly explain the significant gap on the highly-skewed **StackCube** task in Section V.

D. Empirical Results with Regenerated Sequences

To (approximately) evaluate the counterfactual noise in the data generation process for robotic evaluation, we randomly generate Bernoulli sequences using (as the true data-generating parameters) the empirical success rates of each hardware and simulation task, expanding on the results shown in Table I. This provides an estimate of the average sample complexity for each method *were the empirical success rates equal to the true rates 'in the world'*. This "Bernoulli counterfactual" data is presented in Table II; in that table, all entries present the empirical mean complexities (with standard deviation) over 400 regenerated sequences per task.

E. Mathematical and Numerical Notes

1) Worst-Case Null Hypotheses: The worst-case null hypotheses are computed in this framework as the real number $p \in (0, 1)$ that maximizes the expected log-likelihood ratio. First, noting the monotonicity properties of the joint distribution, we claim that the worst-case null hypothesis must lie on the line $p_0 = p_1 = p \in (0, 1)$. Second, noting the optimal power properties of the SPRT for simple-vs-simple problems, we construct the log probability-ratio test maximization as:

$$\arg\max_p \mathbb{E}_{x\sim(p,p)} \left[(\frac{p_1}{p})^x (\frac{1-p_1}{1-p})^{1-x} (\frac{p_0}{p})^x (\frac{1-p_0}{p})^{1-x} \right]$$
$$\equiv \arg\max_p \mathbb{E} \left[x \log \frac{p_0 p_1}{p^2} + (1-x) \log \frac{(1-p_0)(1-p_1)}{(1-p)^2} \right]$$

Differentiating, the solution is the interpolation in the natural parameter space of the Bernoulli distribution:

$$\log \frac{p^*}{1-p^*} = \frac{\log \frac{p_0}{1-p_0} + \log \frac{p_1}{1-p_1}}{2}$$
$$\implies \eta^* = \frac{\eta_0 + \eta_1}{2};$$

the reconstruction of p^* follows directly as

$$p^* = (1 + \exp \eta^*)^{-1}.$$

That is, the worst-case null in the sense of 'falsely' maximizing the probability ratio test under the null is precisely the interpolation in natural parameter space of (p_0, p_1). In fact, the 'true' worst-case null is difficult to compute exactly; as we verify the Type-1 Error control against the additional methods of linear projection in the nominal parameter space

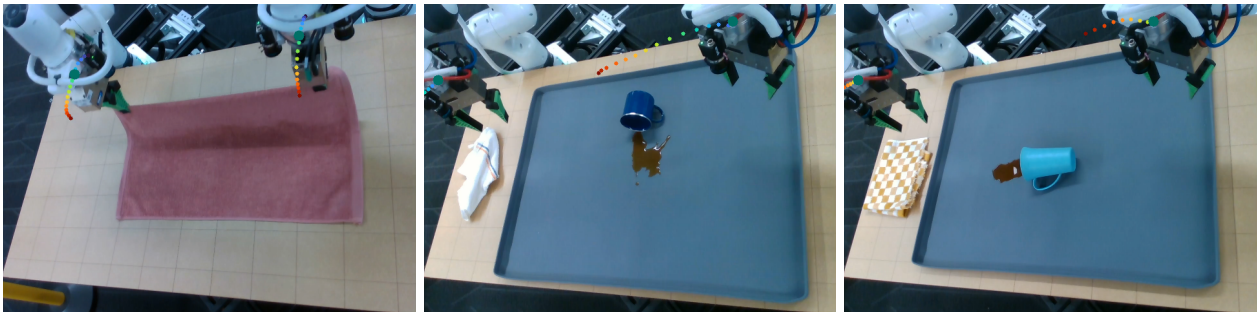
$$p^* = \frac{p_0 + p_1}{2}$$

and as the interpolation under the KL-divergence 'pseudo-distance:'

$$p^* = \{p' \in (0, 1) \mid \mathrm{KL}(p_0, p') = \mathrm{KL}(p_1, p')\}.$$

In practice, assuming continuity corrections are applied to any case in which p_0 or p_1 belong to $\{0, 1\}$, these methods generally result in similar estimates of the worst-case null hypothesis, and form a small region in which the error control can be verified to greater numerical accuracy.

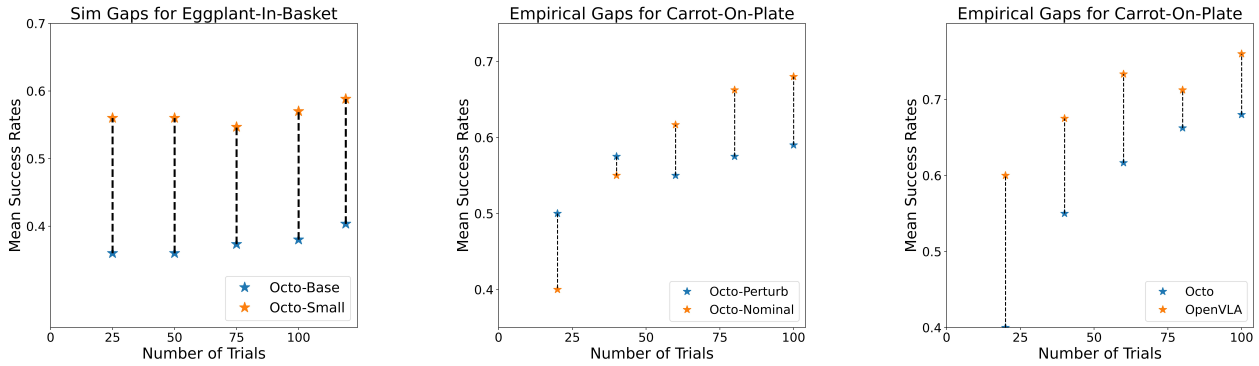
2) Discretizing the Null Hypotheses: In order to discretize the null hypotheses safely, it is necessary to ensure coverage over the set of possible worst-case nulls: $\{(p, p) : p \in [0, 1]\}$. First, we establish an interior bound $(\epsilon, 1-\epsilon)$ to the necessary values $p \in (0, 1)$. Specifically, for a fixed $N_{\max}$ one can derive a value of ϵ such that if $p \geq 1-\epsilon$ (or $p \leq \epsilon$), it holds w.p. $\geq 1 - \alpha^*$ that $\hat{p}_{1,n} = 1$ (resp. 0) for all $n \in \{1, ..., N_{\max}\}$. These extremal nulls pose no risk to the algorithm (because they cannot violate α^* Type-I error if we never **RejectNull** when $\hat{p}_1 \leq \hat{p}_0$). With this limitation in place we avoid problems arising from the rapid decay of the variance near 0 and 1 in the distribution set. Now, discretization in the range $(\epsilon, 1-\epsilon)$ can be undertaken to approximate all possible worst-case null hypotheses. In practice, we used approximately 100 points for $N_{\max}$ up to 500; this is significantly (3x) more than the default in the Scipy implementation of Barnard's Test [49]. Note that formally, this discretization can be ensured to be safe numerically by using Pinsker's inequality to relate the total variation distance (which upper bounds, for example, the event of a false rejection from a null hypothesis) to the KL-divergence; the implication of the inequality is that the false rejection rate error due to discretization is upper bounded by a monotonic function of the maximal KL divergence between any adjacent points in the discretization; for a sufficiently dense discretization, the error can be made arbitrarily small.



(a) **BimanualFoldRedTowel** (b) **BimanualCleanUpSpill** (ID) (c) **BimanualCleanUpSpill** (OOD)

(d) **PutCarrotOnPlate** (no distractors) (e) **PutCarrotOnPlate** (with distractors)

Fig. 5: Snapshots of robot policy evaluation tasks. (Top) Bimanual manipulation tasks with diffusion policy. Colored dots represent the camera projection of planned future end-effector positions. In **BimanualFoldRedTowel**, all the evaluations are done with in-distribution (ID) initial conditions and we compare two policy checkpoints from a single training run. In **BimanualCleanUpSpill**, we evaluate a single policy checkpoint in ID initial conditions with a white towel and out-of-distribution (OOD) initial conditions with a green towel to measure generalization performance. (Bottom) **PutCarrotOnPlate** task on the WidowX platform in a toy kitchen environment. The carrot is initially placed in one of three possible locations on the stove. The environment can either have no distractors or two distractors. We compare Octo and OpenVLA under the nominal environment distribution, and compare Octo performance in nominal environment distribution and under distribution shift. Detailed policy comparison metrics are given in Table I.



(a) Octo-Base and Octo-Small in simulation **EggplantInBasket** task (b) Octo-Base (**Env1**) and Octo-Base (**Env2**) in real-world **CarrotOnPlate** task (c) Octo-Base (**Env1**) and OpenVLA (**Env1**) in real-world **CarrotOnPlate** task

Fig. 6: Running empirical success rates of two policies as the number of trials increases. (a) In the **EggplantInBasket** task, there is a consistent gap in performance due to lower statistical uncertainty. This is reflected in Table I (row 8) where STEP terminates at N = 119. (b and c) On the other hand, in the **CarrotOnPlate** task, the relative performance consistently fluctuates and even sometimes flips due to high statistical uncertainty arising from the close competition between two policies. This leads to even SPRT oracle requiring more than 500 trials to confidently determine the relative performance (Table III).

Task	Type	α^*	$N_{\max}$	N	$\hat{p}_0$	$\hat{p}_1$	SAVI	Lai	STEP (Ours)	SPRT***
FoldRedTowel	π_i	0.05	50	50	0.560	0.920	21.1 (0.63)	23.7 (0.56)	21.8 (0.47)	14.5 (0.53)
FoldRedTowel	π_i	0.05	200	50	0.560	0.920	21.1 (0.63)	27.0 (0.57)	24.4 (0.53)	14.5 (0.53)
FoldRedTowel	π_i	0.05	500	50	0.560	0.920	21.1 (0.63)	32.0 (0.56)	28.0 (0.56)	14.5 (0.53)
CleanUpSpill	$\mathcal{D}^i_{s_0,o_0}$	0.05	50	50	0.280	0.800	13.8 (0.44)	16.9 (0.39)	16.6 (0.31)	11.4 (0.41)
CleanUpSpill	$\mathcal{D}^i_{s_0,o_0}$	0.05	200	50	0.280	0.800	13.8 (0.44)	19.6 (0.43)	18.4 (0.36)	11.4 (0.41)
CleanUpSpill	$\mathcal{D}^i_{s_0,o_0}$	0.05	500	50	0.280	0.800	13.8 (0.44)	23.4 (0.45)	21.0 (0.43)	11.4 (0.41)
CarrotOnPlate	$\mathcal{D}^i_{s_0,o_0}$	0.05	100	100	0.590	0.680	–	–	–	–
CarrotOnPlate	π_i	0.05	100	100	0.680	0.760	–	–	–	–
SpoonOnTowel	π_i	0.01	500	500	0.084	0.386	43.3 (1.20)	55.7 (1.13)	48.2 (1.06)	34.5 (1.08)
EggplantInBasket	π_i	0.01	500	500	0.400	0.564	235.8 (6.5)	200.0 (4.6)	183.3 (4.6)	193.4 (5.3)
StackCube	π_i	0.01	500	500	0.000	0.030	330.3 (4.7)	386.2 (4.5)	267.7 (4.2)	70.5 (3.0)
Multitask	π_i	0.03	1500	1500	N/A	N/A	609.4	641.9	499.2	298.4

TABLE II: Empirical **expected** time-to-correct-decision for all **hardware (top)** and **simulation (bottom)** policy comparisons. The contexts, parameters, and annotation are identical to those in Table I. Summary statistics are taken over 400 random trajectories generated according to a Bernoulli distribution with data-generating (i.e., 'true') parameters corresponding to the observed empirical success rates $(\hat{p}_0, \hat{p}_1)$. We report the average stopping times of all methods on the right of the table for every context (standard deviation of the *empirical mean* in parentheses). In all cases: lower is better. In the Multitask setting, we test $p_1 > p_0$ *uniformly across the preceding three tasks*. This stopping time is the sum by column of the average stopping times for the three tasks. Our method saves the evaluator approximately 110 to 140 trials (in expectation) in uniform certification over these three tasks as compared to either feasible baseline. Note that for any single sequence of evaluations, the standard deviation for the stopping time on a given task can be approximately computed by multiplying the parenthetical standard deviation by 20 (the square root of the number of trials (400)). This correction confirms that the observed improvement of 200 evaluation trials saved in multitask simulation in Table I is likely more than is to be expected given the respective task success rates, but is not at all implausible given the degree of inherent randomness in the data generation process.

3) Technical Setting: The problem formulation in Section III was presented informally to avoid needless over-technical confusion. Slightly more precisely, we assume necessary measurability conditions on the random variables representing the success or failure of the policy in the environment. Given the probability space implicit in this assumption, the concatenation of observations constitutes the natural filtration on this space; that is, $\mathcal{F}_n = \{n, Z_1, Z_2, ..., Z_n\}$. In practice, knowledge of the sufficient statistic for exponential families induces us to use the compressed filtration $\mathcal{F}_n^{[comp]} = \{n, \sum_{i=1}^n z_{0,i}, \sum_{i=1}^n z_{1,i}\}$. Interestingly, using the Neyman-Pearson lemma, one can show that the two-dimensional state represents a lossy compression as a three-dimensional state is needed to construct the optimal exact SPRT.

4) Intuition for Tests: We quickly summarize a few examples of extremal test procedures that can help provide scaffolding for the reader in terms of understanding the tradeoffs inherent between Type-I Error, Type-II Error, and expected sample size. First and foremost, safety is always possible in the Type-I sense: simply never reject the null (i.e., without looking at any data). Slightly more subtly, safety and small sample size is always feasible, as described in the footnote in Section III: decide without looking at any data, but first generate an independent random number uniformly on $[0, 1]$ and reject if the number is less than α^*, otherwise fail to reject. Power and small sample sizes can be obtained accordingly at the cost of violating Type-I Error (just reject instead of failing to reject). Power-1 tests finish out the last leg of the triangle – waiting an arbitrarily long time can allow for simultaneous control of Type-I and Type-II error (the N-P Lemma only concludes that in the batch setting – where N is fixed and finite – there exist instances for which Type-I and Type-II Error cannot be simultaneously controlled).

F. Additional Discussion for Table I

*1) Results for **FoldRedTowel**:* In **FoldRedTowel**, we compare two policy checkpoints from a single training run. The baseline policy π_0 was trained for 10000 gradient steps with an AdamW [32] optimizer, and the other policy π_1 continued training for an additional 70000 steps.

For evaluation, five in-distribution (ID) initial conditions were chosen and repeated 10 times each, constituting 50 total trials. As shown in Table I (rows 1-3), the empirical gap in success rates was 36 percentage points (56% to 92% success), suggesting that π_0 was under-trained. Each sequential method detected a significant difference at level $\alpha^* = 0.05$ in 19 – 23 trials. That is, the last 27 – 31 rollouts per policy are unnecessary for confirming the improvement of π_1 over π_0. Additionally, both Lai method and STEP sequential procedures were each tuned for an $N_{\max}$ of 50 (row 1), 200 (row 2), and 500 (row 3) rollouts. In the latter two cases, additional rollouts could have been run up to 200 or 500 per policy if the gap was smaller *without compromising the validity of the decision*.

*2) Results for **CarrotOnPlate**:* **Task Details** For this task, the initial placement of the carrot is uniformly sampled from three possible locations (left, center, or right) on the counter, and the plate is placed next to the sink (see Figure 5d), and the robot gripper is aligned with the carrot at the start of each trial. In addition, object distractors are sampled uniformly (without replacement) from the following object categories: orange, apple, green and blue sponges, brown and yellow cubes, eggplant, spoon, and towel. The initial locations of distractors is also sampled uniformly without replacement from four possibilities: on the stove, left of the stove, above the stove, or next to the faucet. The distractors are physically placed according to a uniform (continuous) distribution within the selected region.

We evaluate STEP on two open-source vision-language-action (VLA) models: Octo-Base [37], an action-chunking transformer-based diffusion policy, and OpenVLA [25], an autoregressive policy leveraging a pretrained large language model backbone. All experiments were conducted in a toy kitchen environment from the Bridge Data V2 dataset [52], which is included in both policies' training data. We considered the task of placing a carrot on a plate (see Fig. 5d and Fig. 5e), which is representative of evaluations investigated in [37, 25]. All policies were run on the Widow X 250S following the setup in [52]. See Supplement VIII-F for further task implementation details.

The environment uncertainty for **CarrotOnPlate** follows a categorical distribution with two outcomes: no object distractors or two object distractors. We utilize the environment distribution **Env1**, in which there are no distractors with probability 0.8, and two distractors otherwise. We sample 100 environment configurations for each of: i) Octo under distribution **Env1**, ii) OpenVLA under distribution **Env1**. We then compare the performance of each VLA in this distribution over task realizations. In addition, we gather trials for Octo under **Env2**, which has no distractors with probability 0.6, to sequentially test the effect of the distribution shift **Env1** $\rightarrow$ **Env2**.

In the **CarrotOnPlate** policy comparison (Table I, row 4), $\hat{p}_0$ corresponds to Octo (**Env1**) and $\hat{p}_1$ corresponds to OpenVLA (**Env1**). We observe no significant result at $\alpha = 0.05$ despite the empirical gap of 8 percentage points in favor of OpenVLA. Similarly, Octo has a 59% empirical success rate in the latter environment distribution, reflecting a gap of 9 percentage points to its performance under **Env1** (68%). Again, no method returns a significant result. Importantly, insignificance of these tests does not mean that the null hypotheses should be accepted [20]; it is entirely possible that OpenVLA indeed outperforms Octo, or that the added distractors do affect Octo's performance. Instead, the key takeaway is that 100 trials (apiece) is not enough to reliably distinguish gaps of 10 percentage points, reflecting a fundamental limit from statistical testing theory. In Section VIII-F3, we investigate this data insufficiency to show that, if the ground truth values were equal to the empirical success rates (68% vs. 76%), then we would require $N_{\max} = 500$ trials to confidently determine $p_1 > p_0$. This number is an order of magnitude larger than the current norms, reflecting fundamental yet often overlooked challenges in trustworthy policy comparison.

*3) Further Analysis of **CarrotOnPlate** Experiments:* We explore the results of the **CarrotOnPlate** hardware results in more detail. Fig. 6b and Fig. 6c illustrate how the running empirical success rates change as N grows. Note that the relative performance consistently fluctuates and even sometimes flips, which indicates the inherent difficulty of comparison when the two policies are closely competing. In order to estimate the minimum number of necessary trials for these challenging comparisons, we run the SPRT Oracle on multiple instances of $N_{\max}$. Namely, we assume that the true underlying distribution matches the terminal empirical success rates ($\mathbb{H}_1 : (p_0, p_1) = (0.59, 0.68)$ for **Octo** (**Env2**) vs **Octo** (**Env1**) and $\mathbb{H}_1 : (p_0, p_1) = (0.68, 0.76)$ for **Octo** (**Env1**) vs **OpenVLA** (**Env1**)). We determine the worst-case point null (corresponding $\mathbb{H}_0 : (p_0, p_1) = h_0^* \in \mathcal{H}_0$) for each case and run the SPRT Oracle on the associated simple-vs-simple test, where it is essentially optimal. We observe the following empirical power results (Table III), which can be understood as approximating the probability of rejecting the null (under the draw of the sequence of i.i.d. data) at each level of $N_{\max}$ when the true gap matches the empirical gap observed on 100 trials in hardware.

Case ($\downarrow$)	$N_{\max}$ ($\rightarrow$)	100	200	300	400	500
$(0.59, 0.68)$	SPRT Power ($\rightarrow$)	0.324	0.513	0.676	0.762	0.823
$(0.68, 0.76)$	SPRT Power ($\rightarrow$)	0.337	0.491	0.643	0.724	0.804

TABLE III: Empirical power of SPRT Oracle on distributions matching the empirical gaps observed in hardware trials of **CarrotOnPlate**. This suggests that at 200 trials per policy, there is only about a 50% chance of observing a sequence leading to rejection of the null; even for the oracle, 500 trials are required before this approximate probability reaches 80%.

As shown Table III, nearly 400 trials are required before reaching an approximately 75% chance of rejection over the draw of observed sequences. We emphasize that this is computed via a method that is optimal with respect to the expected sample size; as such, the evaluation requirements are primarily fundamental to the variance of Bernoulli random variables, and thus represent fundamental uncertainty and sample complexity for the policy comparison problem.

*4) Additional **CarrotOnPlate** Experiments:* A prior iteration of the **CarrotOnPlate** experiments (not reported in Section V) involved a hardware implementation error in which the end-effector rotation commands output by the policies were not correctly published. As a result, for the purpose of policy comparison, we label these policies **PolicyA** (in place of Octo) and **PolicyB** (in place of OpenVLA). Note that the hardware implementation does not invalidate the policy comparison procedure itself. The environment distributions **Env1**

N	$\hat{p}_0$	$\hat{p}_1$	SAVI	Lai	STEP	SPRT***
200	0.530	0.560	–	–	–	–
150	0.613	0.827	132	64	61	26

TABLE IV: Additional **CarrotOnPlate** evaluations with $N_{\max} = 200$. *Row 1:* Comparing **PolicyA** under **Env1** (π_1) with **PolicyA** under **Env2** (π_0). *Row 2:* Comparing **PolicyA** under **Env1** (π_0) with **PolicyB** under **Env1** (π_1).

and **Env2** are as described in Section V. We set $N_{\max} = 200$ as the evaluation budget for the following three settings: i) **PolicyA** (under **Env1**), ii) **PolicyB** (under **Env1**), and iii) **PolicyA** (under **Env2**). We compare **PolicyA** (**Env1**) with **PolicyB** (**Env1**), and **PolicyA** (**Env1**) with **PolicyA** (**Env2**). These results are listed in Table IV. Observe that despite utilizing the full budget, no procedure yields conclusive results in the first comparison. However, in the second comparison, we began running the evaluation procedure on the collected data after collecting 150 evaluations per policy. Because every method had decided by that point, these evaluations were able to terminate early, saving us 50 hardware evaluations per setting. Had we been running the evaluation the entire time, we could have saved ourselves an additional 20 evaluations per setting, as SAVI (the slowest) terminated in just over 130 evaluations.