MORE HASTE, MORE SPEED: ADAPTIVE SEQUENTIAL TESTING FOR EFFICIENT DIAGNOSTIC EVALUATION OF TEXT-TO-IMAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diagnostic benchmarks test whether text-to-image (T2I) models satisfy conditions such as object presence, attribute binding, and spatial relations across repeated generations. They typically allocate identical image budgets to every prompt, although obvious successes and failures need less evidence than conditions near the validity threshold. We introduce **HASTE** (**H**ierarchical **A**daptive **S**equential **T**esting for **E**valuation), which varies the sample count without omitting any benchmark condition. Conditions are organized by semantic prerequisites: for example, "a car is present" must hold before "the car is red" can hold. **HASTE** evaluates prerequisite conditions first and uses their results to inform priors for more specific conditions. It then updates each condition's posterior with binary vision-language model judgments and stops when there is sufficient evidence that its success probability lies above or below the benchmark's validity threshold. A minimum sample requirement limits premature stopping, while conditions uncertain at the sample cap use the fixed-budget decision rule. Across six T2I models, **HASTE** achieves average speedups of $2.60\times$ on DSG and $4.93\times$ on FailureAtlas, with failure-set Jaccard agreement above 0.97 relative to full-budget evaluation. At matched average image budgets, **HASTE** attains 0.9806 Jaccard agreement on DSG versus 0.9220 for uniform sampling, and 0.9794 on FailureAtlas versus 0.6558 for uniform sampling.

1 INTRODUCTION

Modern text-to-image (T2I) models generate visually compelling images across diverse domains (Rombach et al., 2022; Saharia et al., 2022; Podell et al., 2024). However, they frequently fail on compositional prompts by omitting requested entities, misbinding attributes, or violating spatial arrangements. To discover these failures, diagnostic benchmarks decompose natural-language prompts into fine-grained *evaluation conditions* (hereafter simply *conditions*), such as object presence, attribute bindings, and relational constraints (Hu et al., 2023; Huang et al., 2023). DSG (Cho et al., 2024), for example, organizes atomic propositions and their semantic dependencies into directed acyclic graphs (DAGs). Recent benchmarks verify these conditions using automated questions answered by a vision-language model (VLM) (Chen et al., 2025; Li et al., 2026), which inspects each generated image and returns a binary decision indicating whether the condition is met.

Because image generation is stochastic, a single image provides only one binary outcome, which cannot establish whether a model consistently satisfies a condition. Reliable assessment therefore requires generating multiple images per prompt: a benchmark typically declares a condition valid only if the fraction of compliant generations reaches a validity threshold (e.g., 50% or 80%), and classifies it as a failure otherwise. Across thousands of conditions, drawing a fixed sample budget creates a heavy computational burden. For example, in FailureAtlas (Chen et al., 2025), evaluating over 440,000 conditions with 25 images each required more than 11 million generated images and 16 H200 GPU-days for image generation and VLM verification. As new T2I models are released at a rapid pace, repeating exhaustive evaluations for every candidate model is highly time-consuming.

Several approaches could accelerate evaluation, but existing methods face trade-offs in reliability or coverage. For example, uniformly reducing the sample budget can misclassify failures under difficult borderline conditions (Chen et al., 2024). Active subset selection reduces the number of inputs tested rather than the number of images sampled per input, limiting diagnostic coverage (Gao et al., 2023; Xu et al., 2024; Qiu et al., 2025). Uncalibrated sequential tests at individual nodes may declare a node safe after a few favorable samples, before later samples reveal failures (Wald, 1945). These limitations raise a **cost–fidelity question**:

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.

How can we reduce the sampling burden of diagnostic T2I evaluation while still assessing every condition and reliably identifying its failures?

In this paper, we observe that after only a few image samples, some conditions are clearly above or below the validity threshold, while those near the threshold remain uncertain. For example, under an illustrative decision threshold of $\tau = 0.5$, after an initial batch of 5 images, a model may consistently succeed on the prompt "a red apple" (yielding 5 images, each with exactly one red apple), while failing consistently on "a red apple with a glossy surface reflecting a mechanical clock" (yielding 0 images with the clock reflection). Both can be decided within these 5 observations under this threshold, yet a fixed budget may draw 25 images for each, adding little information relevant to the final benchmark decision and wasting redundant image generations and VLM queries. In contrast, a subtle condition like "a red apple to the left of a green pear" may lie near the validity threshold and require nearly the full budget to resolve. Because *decisive conditions* (those well separated from the decision boundary) require far fewer observations than borderline ones, stopping early on clear outcomes can therefore substantially accelerate evaluation. In general, we observe that different prompts may require different numbers of generated images for reliable evaluation, calling for a method that automatically determines the appropriate sample size for each prompt.

Based on these observations, we introduce **HASTE** (**H**ierarchical **A**daptive **S**equential **T**esting for **E**valuation) to accelerate diagnostic T2I evaluation. **HASTE** organizes conditions by semantic prerequisites: "a car is present," for example, is a parent of "the car is red" because the latter can be satisfied only if a car is present. This structure lets **HASTE** evaluate parents first and use their results to form a prior for each child's success probability. The child's own binary VLM judgments then update that prior to a posterior. After a minimum number of samples, a condition stops when its posterior provides sufficient evidence that its success probability lies above or below the benchmark's validity threshold. Conditions still unresolved at the sample cap K follow the fixed-budget empirical rule. Prompts whose conditions are resolved early therefore need fewer generated images, while harder prompts can use up to K, reducing image generation and VLM scoring costs without skipping any benchmark condition. Figure 1 summarizes this workflow.

Our contributions are:

1. We formulate diagnostic T2I evaluation as an adaptive sampling problem: every benchmark condition is evaluated, but the number of samples varies with the evidence needed to classify it.
2. We develop **HASTE**, which evaluates prerequisite conditions first and uses their observed results to set priors for more specific conditions. It then updates each condition's posterior with its own VLM judgments and stops when the evidence supports a decision. Prior construction does not use the reference decision obtained by evaluating that condition on all images.
3. Across six T2I models, **HASTE** achieves average speedups of $2.60\times$ on DSG (Cho et al., 2024) and $4.93\times$ on FailureAtlas (Chen et al., 2025), while maintaining failure-set Jaccard agreement above 0.97 with the full-budget references. At matched average image budgets (roughly $K = 20$ on DSG and $K = 10$ on FailureAtlas, compared to the full budget $K = 50$), its agreements are 0.9806 and 0.9794, compared with 0.9220 and 0.6558 for uniform sampling.

2 Related Work

Automatic T2I diagnostic evaluation. TIFA (Hu et al., 2023) uses question answering to evaluate image–prompt faithfulness, while GenEval (Ghosh et al., 2023) and T2I-CompBench (Huang et al., 2023) test object-level and compositional constraints. To preserve semantic relationships across questions, DSG (Cho et al., 2024) organizes atomic propositions into a dependency DAG. FailureAtlas (Chen et al., 2025) structures evaluation conditions into hierarchical entity–attribute graphs to systematically expose failure modes. T2I-CoReBench (Li et al., 2026) extends checklist-based evaluation to composition and complex reasoning. While these evaluators specify *what* visual criteria to inspect, standard protocols uniformly evaluate every condition with a fixed sample budget. **HASTE** complements them by determining *how many* generations each condition requires.

Failure discovery and adaptive test selection. Slice-discovery methods identify error groups in datasets (Eyuboglu et al., 2022; Chen et al., 2023), while active exploration frameworks search for failure conditions or hard prompts, such as FailureAtlas (Chen et al., 2025), AdaVision (Gao et al., 2023), and STEPS (Qiu et al., 2025). These methods decide *which* inputs to evaluate. In contrast, **HASTE** maintains full benchmark coverage and adapts sample sizes per condition.

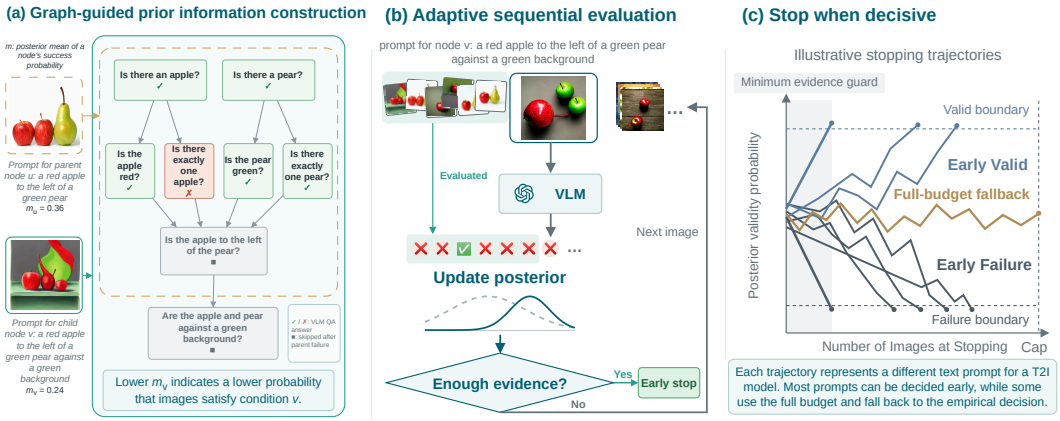


Figure 1: **HASTE** accelerates diagnostic evaluation through hierarchical sequential testing. (a) Completed parent evidence forms a prior for a child condition. (b) Sequential VLM judgments update the posterior. (c) Decisive conditions stop early, while unresolved conditions exhaust the full budget.

Sequential and budgeted evaluation. Sequential analysis replaces predetermined sample sizes with evidence-dependent stopping rules (Wald, 1945). In machine learning, adaptive stopping and confidence sequences terminate sampling once sufficient confidence is reached (Mnih et al., 2008; Howard et al., 2021). While standard stopping rules evaluate conditions in isolation, **HASTE** propagates semantic dependencies across the benchmark DAG through conjugate Beta priors, accelerating termination without label leakage.

3 Background

Modern diagnostic benchmarks for text-to-image (T2I) generation deploy vision-language model (VLM) evaluators to test whether generated images satisfy fine-grained semantic conditions derived from input prompts (Cho et al., 2024; Chen et al., 2025; Li et al., 2026).

Diagnostic evaluation and fixed-budget reference. Let v denote an evaluation condition (such as object presence, attribute binding, or spatial relations) and x_v its associated prompt. A T2I model $\mathcal{M}$ generates the j-th independent image sample $I_{v,j} \sim \mathcal{M}(\cdot \mid x_v)$, and a VLM evaluator $\mathcal{E}$ returns a binary judgment:

$$y_{v,j} = \mathcal{E}(I_{v,j}, v) \in \{0,1\}, \qquad \theta_v = \Pr(y_{v,j} = 1), \tag{1}$$

where $y_{v,j} = 1$ indicates that $I_{v,j}$ satisfies condition v. Here, θ_v denotes the evaluator-defined success probability under repeated generation. In standard diagnostic protocols, evaluation proceeds under an exhaustive budget of K independent generations per prompt. The empirical success rate is:

$$\widehat{\theta}_v = \frac{1}{K} \sum_{j=1}^{K} y_{v,j}. \tag{2}$$

For a validity threshold $\tau \in (0,1)$, the reference decision d_v^{ref} evaluated at the full budget of K image generations declares v as VALID when $\widehat{\theta}_v \geq \tau$ and as FAILURE otherwise:

$$d_v^{\text{ref}} = \begin{cases} \text{VALID}, & \widehat{\theta}_v \geq \tau, \\ \text{FAILURE}, & \widehat{\theta}_v < \tau. \end{cases} \tag{3}$$

Condition graphs in structured diagnostic benchmarks. Structured diagnostic benchmarks organize evaluation conditions into directed acyclic graphs (DAGs) to assess model behavior systematically (Cho et al., 2024; Chen et al., 2025). For example, DSG (Cho et al., 2024) decomposes complex prompts into atomic visual propositions linked by semantic dependencies to facilitate fine-grained failure discovery in T2I models. Similarly, FailureAtlas (Chen et al., 2025) structures hierarchical entity–attribute taxonomies as structured trees (a DAG as well) to discover failure cases.

Formally, let $G = (V, E)$ denote a directed acyclic graph over the set of evaluation conditions V. A directed edge $(u, v) \in E$ represents a semantic prerequisite: condition v depends on the realization of condition u (for example, "a car is present" must hold before "the car is red" can be evaluated).

For each condition $v \in V$, its prerequisite parent set is defined as:

$$\mathrm{pa}(v) = \{u \in V : (u, v) \in E\}. \tag{4}$$

Root conditions $V_{\mathrm{root}} = \{v \in V : \mathrm{pa}(v) = \varnothing\}$ have no prerequisites.

4 METHOD

Although these condition graphs provide structural clarity for organizing evaluation questions and categorizing failure modes, existing frameworks treat the graph purely as an organizational hierarchy. During evaluation, standard benchmarks continue to sample an exhaustive, uniform budget of K images and scoring queries for every condition $v \in V$ independently, without using the dependency structure to reduce evaluation compute. In contrast, we aim to build on the pre-existing condition graph structure to accelerate diagnostic evaluation. We introduce **HASTE**: Section 4.1 formalizes the adaptive sequential testing problem and complexity metrics over condition graphs, and Section 4.2 presents the hierarchical sequential testing algorithm for accelerating diagnostic evaluation.

4.1 PROBLEM FORMULATION: ADAPTIVE DIAGNOSTIC EVALUATION

Sequential testing over condition graphs. We exploit the semantic prerequisite structure in condition graphs $G = (V, E)$ to accelerate diagnostic evaluation of text-to-image models. An adaptive evaluation procedure dynamically assigns an evidence-dependent sample count $T_v \in \{1, \ldots, K\}$ to each condition $v \in V_{\mathrm{eval}}$ and outputs an estimated decision:

$$d_v \in \{\textsc{Valid}, \textsc{Failure}\}, \qquad 1 \le T_v \le K. \tag{5}$$

The goal is to minimize total evaluation compute while maintaining high decision agreement with the exhaustive reference decisions d_v^{ref}.

Hardware-agnostic sample and query complexity. Total evaluation cost decomposes into two distinct, hardware-agnostic complexity metrics: physical image generation count C_{image} and vision-language model scoring query count C_{vlm}. Diagnostic benchmarks exhibit two operational structures that govern these quantities:

Under *independent image streams*, where each condition generates its own distinct images (e.g., FailureAtlas (Chen et al., 2025)), image generations and VLM queries coincide:

$$C_{\mathrm{image}} = C_{\mathrm{vlm}} = \sum_{v \in V_{\mathrm{eval}}} T_v. \tag{6}$$

Under *shared image streams*, where multiple conditions evaluate the same generated images for each prompt $x \in \mathcal{X}$ (e.g., DSG (Cho et al., 2024), T2I-CoReBench (Li et al., 2026)), let $V_x = \{v \in V_{\mathrm{eval}} : x_v = x\}$ denote the conditions sharing prompt x. Because conditions sequentially reuse cached images, image cost is governed by the longest stopping time across all propositions for prompt x, whereas VLM scoring queries scale additively across individual propositions:

$$C_{\mathrm{image}} = \sum_{x \in \mathcal{X}} \max_{v \in V_x} T_v, \qquad C_{\mathrm{vlm}} = \sum_{v \in V_{\mathrm{eval}}} T_v. \tag{7}$$

4.2 HIERARCHICAL SEQUENTIAL TESTING

In diagnostic benchmarks with thousands of conditions, a fixed sample budget spends unnecessary computation on conditions whose outcomes are already clear. In addition, independent stopping rules overlook relationships between conditions: the result for "a car is present", for example, can inform the evaluation of "the car is red". The key novelty of **HASTE** is to use these relationships to guide sequential sampling while still faithfully evaluating every condition. It constructs each child's prior from completed parent evaluations, updates the posterior with the child's own binary judgments, and stops only after a minimum sample count is met and the posterior crosses a decision threshold. We describe the steps of **HASTE** below.

Step 1: Graph-guided prior construction. For each condition v, we model the evaluator's binary judgments as $y_{v,j} \sim \mathrm{Bernoulli}(\theta_v)$, where θ_v is the unknown probability that a generated image satisfies v. We assign θ_v a Beta prior with parameters (α_v, β_v) and process conditions in topological order, so each parent is evaluated before its children. Root conditions receive the uniform prior $(\alpha_v, \beta_v) = (1, 1)$. For a non-root condition v, each parent $u \in \mathrm{pa}(v)$ has already completed

evaluation, observing evidence $\mathcal{D}_u = (s_u, f_u)$ with s_u successes and f_u failures. We summarize the parent's evidence by its posterior mean:

$$m_u = \mathbb{E}[\theta_u \mid \mathcal{D}_u] = \frac{\alpha_u + s_u}{\alpha_u + \beta_u + s_u + f_u}. \tag{8}$$

Importantly, parent results inform the child's prior because the child can satisfy its condition only when its semantic prerequisites are met. For example, "the car is red" cannot hold if "a car is present" fails. A parent that rarely succeeds therefore suggests a lower success probability for its child, even before the child is sampled. Because failure of any prerequisite can prevent child success, we summarize multiple parents by their lowest posterior mean:

$$g_v = \min_{u \in \mathrm{pa}(v)} m_u. \tag{9}$$

We blend this summary with a global reference mean μ_{g} to obtain the child's prior mean:

$$\mu_v = \rho g_v + (1 - \rho)\mu_{\mathrm{g}}, \tag{10}$$

where $\rho \in [0, 1]$ controls how strongly the parent informs the child's prior, and $\mu_{\mathrm{g}} = 0.5$ is the mean of the $\mathrm{Beta}(1, 1)$ prior used at the root. The prior parameters are then initialized as $(\alpha_v, \beta_v) = \big(\kappa\mu_v,\, \kappa(1 - \mu_v)\big)$, where $\kappa > 0$ represents the prior sample strength. The parent-informed prior gives each child an initial estimate based on its prerequisites, helping identify likely failures with fewer child samples. The child's own judgments then update that estimate and can outweigh the prior as evidence accumulates.

Step 2: Sequential posterior updating. For condition v, each accepted VLM judgment on sample slot $t \in \{1, \ldots, K\}$ increments the cumulative success count s_t or failure count f_t. Combining these observations with the prior from Step 1 gives the Beta posterior:

$$\theta_v \mid \mathcal{D}_t \sim \mathrm{Beta}(\alpha_v + s_t,\, \beta_v + f_t), \tag{11}$$

where $\mathcal{D}_t$ contains the accepted judgments through slot t. We then compute the posterior probability that condition v satisfies validity threshold τ:

$$p_{v,t} = \Pr(\theta_v \geq \tau \mid \mathcal{D}_t) = 1 - I_\tau(\alpha_v + s_t,\, \beta_v + f_t), \tag{12}$$

where $I_\tau(\alpha, \beta)$ is the cumulative distribution function (CDF) of $\mathrm{Beta}(\alpha, \beta)$ evaluated at τ.

Step 3: Early stopping and fixed-budget fallback. To reduce premature decisions from a small number of observations, **HASTE** requires at least $n_{\min}$ accepted judgments before stopping (the *sample gate*). Let $a_t = s_t + f_t$ be the number accepted through sample slot t. The decision rule is:

$$d_{v,t} = \begin{cases} \textsc{Valid}, & a_t \geq n_{\min} \quad \text{and} \quad p_{v,t} \geq 1 - \epsilon, \\ \textsc{Failure}, & a_t \geq n_{\min} \quad \text{and} \quad p_{v,t} \leq \epsilon, \\ \textsc{Continue}, & \text{otherwise.} \end{cases} \tag{13}$$

where ϵ sets the posterior stopping boundaries (by default, $n_{\min} = 5$ and $\epsilon = 0.05$). Thus, **HASTE** stops early only when the posterior strongly favors one side of the validity threshold τ. If a condition remains unresolved after all K sample slots, **HASTE** applies the fixed-budget empirical rule:

$$d_v = d_v^{\mathrm{ref}} = \begin{cases} \textsc{Valid}, & s_K / K \geq \tau, \\ \textsc{Failure}, & \text{otherwise.} \end{cases} \tag{14}$$

When every slot yields an accepted binary judgment, this fallback exactly matches the fixed-budget reference decision. Under an i.i.d. Bernoulli model with an accepted binary answer at every slot, Appendix E.1 bounds the expected stopping time and the resulting image and VLM query costs for both stream types.

5 Experiments

5.1 Experimental Setup and Diagnostic Testbeds

We evaluate **HASTE** by replaying full-budget reference traces ($K = 50$) across two representative diagnostic DAG testbeds: FailureAtlas (Chen et al., 2025) and DSG (Cho et al., 2024) (with supplementary experiments on other testbeds in Appendix C). Both testbeds use Qwen3-VL-8B (Bai et al., 2025) as the VLM evaluator (with other VLMs evaluated in Appendix K) and a validity threshold of $\tau = 0.5$ following Cho et al. (2024), unless stated otherwise. Our evaluation addresses four primary questions: (1) How much image generation and VLM query cost does **HASTE** save across

diverse T2I generators while preserving full-budget failure classification fidelity (§5.2, §5.4)? (2) Does **HASTE** maintain decision fidelity through adaptive, graph-guided sample allocation rather than naive budget reduction (§5.3)? (3) How do proposition stopping times distribute across models of varying generative capability under sample-size gating (§5.5)? (4) How robust is evaluation efficiency across validity thresholds τ, and what do hierarchical prior propagation and sample gating individually contribute (§5.6, §5.7)?

FailureAtlas search DAG. Following the construction in FailureAtlas (Chen et al., 2025), we build a layered DAG of entity–attribute combinations for evaluation by **HASTE**. A root condition tests a base entity (e.g., "tiger"); each child adds a visual attribute (e.g., "wearing a hat"). Edges represent semantic specialization and search dependencies: a child condition is generated only after its parent is verified to be reliable. FailureAtlas assigns an independent image stream to each condition, so total image cost is additive across evaluated nodes, $C_{\text{image}} = \sum_{v \in V} T_v$. The testbed evaluates 12 entities, 25,943 conditions, and 1,297,150 reference images per model.

DSG proposition DAG. Following the construction in DSG (Cho et al., 2024), we organize each prompt's fine-grained questions and their semantic dependencies into a proposition DAG. A root proposition might ask if a cat is present; dependent questions ask about its pose, color, or relation to other objects. Unlike FailureAtlas, all propositions for a prompt inspect the same set of generated images. Total image cost sums the longest stopping time within each prompt, $C_{\text{image}} = \sum_{x \in \mathcal{X}} \max_{v \in V_x} T_v$, while VLM scoring queries sum across propositions, $C_{\text{vlm}} = \sum_{v \in V_{\text{eval}}} T_v$. We evaluate 1,060 items containing 8,182 propositions on 53,000 images. The benchmark sets $\tau = 0.5$ as its validity threshold. Results on T2I-CoReBench (Li et al., 2026), which shares the same prompt-level evaluation format, are reported in Appendix C.

Together, these formulations ensure that edges encode condition dependencies in FailureAtlas and semantic prerequisites among conditions within each prompt in DSG. This pairing tests **HASTE** under distinct graph semantics while preserving the same condition-level evaluation contract, evaluated across multiple T2I models to span both failure-heavy and success-heavy outcome distributions.

Metrics. Let F and F_{ref} denote the sets of identified failure conditions under an evaluated method and the fixed-budget reference, respectively. The primary fidelity metric is *failure-set Jaccard*:

$$J(F, F_{\text{ref}}) = \frac{|F \cap F_{\text{ref}}|}{|F \cup F_{\text{ref}}|}. \tag{15}$$

For DSG, a proposition failure is defined at the prompt level: any prompt–proposition pair classified as failure, and we pool these pairs across the benchmark before computing Eq. (15). For FailureAtlas, F contains the *minimal failed nodes* retained after graph transitions and descendant closure. Some other metrics and matched-cost evaluation protocols are detailed in Appendix G.3.

Baselines. The full-budget evaluation ($K = 50$) serves as the reference baseline, with image generation speedup $1\times$ and failure-set Jaccard 1 by definition. We compare against seven sequential methods: (1) Exact Sequential Elimination (**Exact-SE**) terminates sampling early if and only if remaining samples cannot alter whether the empirical mean reaches τ, guaranteeing lossless early stopping without approximation error; (2) Hoeffding Confidence Sequence (**Hoeffding-CS**) (Howard et al., 2021) stops when a distribution-free, sub-Gaussian time-uniform confidence sequence around the empirical mean excludes τ; (3) Empirical Bernstein Confidence Sequence (**EB-CS**) (Mnih et al., 2008; Howard et al., 2021) tightens confidence radii adaptively via empirical sample variance; (4) Kullback–Leibler Confidence Sequence (**KL-CS**) (Howard et al., 2021) inverts time-uniform Bernoulli relative entropy bounds to obtain tighter boundary coverage; (5) Wald's Sequential Probability Ratio Test (**SPRT**) (Wald, 1945) monitors the log-likelihood ratio between two Bernoulli rates on either side of τ; (6) Any-Time Parameter-Free Thresholding (**APT**) (Locatelli et al., 2016) terminates when the empirical distance from the threshold exceeds a confidence radius; and (7) Graph-regularized Anytime Path Learning (**GrAPL**) (LeJeune et al., 2020) diffuses empirical observations across proposition DAGs via graph Laplacian smoothing. Thus, Exact-SE and the confidence-sequence methods (Hoeffding-CS, EB-CS, and KL-CS) emphasize conservative decision certification, SPRT and APT trade some reference agreement for earlier stopping, and GrAPL additionally exploits dependencies among conditions. Finally, we evaluate a **Budget-Matched Fixed-K** baseline (abbreviated **Fixed-K_{match}**) that uniformly assigns each condition the average sample budget consumed by **HASTE** ($K_{\text{match}} = \lfloor K/\text{Speedup} \rceil$). This comparison tests decision fidelity at matched image cost, separating the effect of adaptive allocation from simple budget reduction.

Table 1: Sequential baseline **image generation speedup** across six T2I models on FailureAtlas (**failure-set Jaccard** in parentheses). Bold indicates the highest speedup in each column.

Method	SD 1.5	SD 3.5-Turbo	SDXL-Turbo	SANA-1.5	FLUX.2-dev	Qwen-Image	Avg.
Exact-SE	1.54 (1.0000)	1.67 (1.0000)	1.75 (1.0000)	1.64 (1.0000)	1.94 (1.0000)	1.66 (1.0000)	1.70 (1.0000)
Hoeffding-CS	1.35 (1.0000)	1.60 (1.0000)	1.80 (1.0000)	1.54 (1.0000)	1.30 (1.0000)	1.60 (1.0000)	1.53 (1.0000)
EB-CS	1.02 (1.0000)	1.04 (1.0000)	1.05 (1.0000)	1.03 (1.0000)	1.09 (1.0000)	1.04 (1.0000)	1.05 (1.0000)
KL-CS	1.55 (0.9999)	1.96 (0.9979)	2.33 (1.0000)	1.85 (0.9999)	2.05 (1.0000)	1.97 (1.0000)	1.95 (0.9996)
SPRT	3.00 (0.9629)	3.73 (0.9774)	4.19 (0.9715)	3.43 (0.9633)	3.10 (0.9539)	3.52 (0.9642)	3.50 (0.9655)
APT	3.12 (0.9584)	4.22 (0.9739)	5.07 (0.9672)	3.81 (0.9611)	4.09 (0.9574)	4.00 (0.9387)	4.05 (0.9595)
GrAPL	3.10 (0.9588)	4.20 (0.9743)	5.05 (0.9676)	3.79 (0.9670)	4.15 (0.9637)	3.99 (0.9392)	4.05 (0.9618)
Fixed-K_{match}	3.78 (0.6321)	4.92 (0.6476)	5.76 (0.8165)	4.56 (0.6444)	5.96 (0.5820)	4.57 (0.6120)	4.93 (0.6558)
HASTE (Ours)	**3.78 (0.9812)**	**4.92 (0.9823)**	**5.76 (0.9862)**	**4.56 (0.9727)**	**5.96 (0.9804)**	**4.57 (0.9738)**	**4.93 (0.9794)**

Table 2: Sequential baseline **image generation speedup** across six T2I models on DSG (**failure-set Jaccard** in parentheses). Bold indicates the highest speedup in each column.

Method	SD 1.5	SD 3.5-Turbo	SDXL-Turbo	SANA-1.5	FLUX.2-dev	Qwen-Image	Avg.
Exact-SE	1.25 (1.0000)	1.42 (1.0000)	1.39 (1.0000)	1.42 (1.0000)	1.56 (1.0000)	1.55 (1.0000)	1.43 (1.0000)
Hoeffding-CS	1.08 (1.0000)	1.24 (1.0000)	1.21 (1.0000)	1.26 (1.0000)	1.45 (1.0000)	1.44 (1.0000)	1.28 (1.0000)
EB-CS	1.01 (1.0000)	1.02 (1.0000)	1.01 (1.0000)	1.02 (1.0000)	1.03 (1.0000)	1.00 (1.0000)	1.02 (1.0000)
KL-CS	1.12 (1.0000)	1.36 (0.9999)	1.31 (1.0000)	1.38 (1.0000)	1.69 (1.0000)	1.66 (1.0000)	1.42 (1.0000)
SPRT	1.65 (0.9732)	2.29 (0.9744)	2.17 (0.9754)	2.29 (0.9749)	2.91 (0.9755)	2.85 (0.9645)	2.36 (0.9730)
APT	1.54 (0.9688)	2.24 (0.9718)	2.09 (0.9728)	2.24 (0.9704)	3.04 (0.9724)	2.92 (0.9621)	2.35 (0.9697)
GrAPL	1.53 (0.9638)	2.23 (0.9694)	2.08 (0.9696)	2.23 (0.9679)	3.03 (0.9674)	2.91 (0.9621)	2.34 (0.9667)
Fixed-K_{match}	1.70 (0.9297)	2.48 (0.9256)	2.30 (0.9357)	2.51 (0.9222)	3.35 (0.9027)	3.23 (0.9161)	2.60 (0.9220)
HASTE (Ours)	**1.70 (0.9736)**	**2.48 (0.9819)**	**2.30 (0.9837)**	**2.51 (0.9801)**	**3.35 (0.9816)**	**3.23 (0.9825)**	**2.60 (0.9806)**

5.2 Primary Cross-Model Results against Sequential Baselines

Tables 1 and 2 report image generation speedups with failure-set Jaccard in parentheses, comparing **HASTE** against competitive sequential testing baselines on FailureAtlas and DSG at $K = 50$ and $\tau = 0.5$. All evaluations use identical fixed outcome pools and average results over five deterministic observation-order seeds (seeds $\{0, 1, 2, 3, 4\}$; other seed settings are detailed in Appendix G.1). Detailed breakdowns of absolute image generation counts, agreement metrics, and hardware wall-clock benchmarks for **HASTE** are provided in Appendix E.4.

Across the six generators, **HASTE** achieves average image generation speedups of $4.93\times$ on FailureAtlas and $2.60\times$ on DSG, with average failure-set Jaccard of 0.9794 and 0.9806, respectively. At the reported operating points, it has both higher speedup and higher Jaccard than SPRT, APT, and GrAPL for every generator in both testbeds. The two benchmarks have different image-cost structures: FailureAtlas uses an independent image stream for each condition, whereas DSG shares generated images across propositions within a prompt.

Exact-SE reproduces the fixed-budget decisions and yields average speedups of $1.70\times$ on FailureAtlas and $1.43\times$ on DSG. The confidence-sequence baselines also retain near-perfect Jaccard at their reported settings, generally with smaller speedups than the approximate stopping methods. Thus, the tables show distinct cost–fidelity operating points: **HASTE** saves more images while allowing some disagreement with the fixed-budget failure sets. Appendix H examines these regimes in more detail, with comprehensive cross-model breakdowns reported in Appendix E.4.

5.3 Decision Fidelity at Matched Image Budgets

A central question in adaptive sequential evaluation is whether **HASTE**'s efficiency gains arise merely from reducing sample counts, or whether its graph structure actively preserves decision fidelity. If an evaluation procedure merely stops early uniformly or lowers the per-condition budget naively, aggregate compute decreases at the cost of severe classification degradation on conditions near the validity threshold.

We compare **HASTE** with the Budget-Matched Fixed-K (Fixed-K_{match}) baseline, which assigns each condition a uniform budget matched to **HASTE**'s average image cost after rounding. On FailureAtlas, both methods therefore achieve approximately $4.93\times$ speedup, but **HASTE** retains a cross-model mean failure-set Jaccard of 0.9794, compared with 0.6558 for uniform allocation (Figure 2). This 32.36-percentage-point gap shows that **HASTE** better preserves fidelity in identifying failures at approximately matched cost.

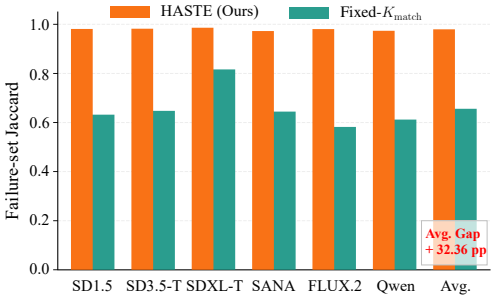


Figure 2: The Fixed-K_{match} baseline uniformly assigns each condition the average sample budget by **HASTE**. At the same speedup, **HASTE** better preserves fidelity to identify failures, achieving 0.9794 average Jaccard compared with 0.6558 for Fixed-K_{match}.

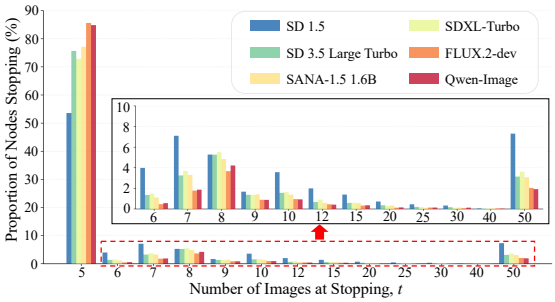


Figure 3: Distribution of the number of images t at stopping, across six T2I models on DSG. With the sample gate $n_{\min} = 5$, higher-capability models resolve up to 85.6% of propositions at the first admissible step, while only 1.9%–7.3% of borderline propositions require the full $K = 50$ budget.

Table 3: **VLM evaluation speedup** across six T2I models on DSG (**failure-set Jaccard** in parentheses). Bold indicates the highest speedup in each column.

Method	SD 1.5	SD 3.5-Turbo	SDXL-Turbo	SANA-1.5	FLUX.2-dev	Qwen-Image	Avg.
Exact-SE	1.62 (1.0000)	1.80 (1.0000)	1.77 (1.0000)	1.80 (1.0000)	1.88 (1.0000)	1.87 (1.0000)	1.79 (1.0000)
Hoeffding-CS	1.51 (1.0000)	1.98 (1.0000)	1.92 (1.0000)	2.00 (1.0000)	2.24 (1.0000)	2.23 (1.0000)	1.98 (1.0000)
EB-CS	1.00 (1.0000)	1.00 (1.0000)	1.00 (1.0000)	1.00 (1.0000)	1.00 (1.0000)	1.00 (1.0000)	1.00 (1.0000)
KL-CS	1.81 (1.0000)	2.64 (0.9999)	2.52 (1.0000)	2.68 (1.0000)	3.19 (1.0000)	3.17 (1.0000)	2.67 (1.0000)
SPRT	3.42 (0.9732)	4.54 (0.9744)	4.36 (0.9754)	4.56 (0.9749)	5.09 (0.9755)	5.09 (0.9645)	4.51 (0.9730)
APT	3.77 (0.9688)	5.70 (0.9718)	5.34 (0.9728)	5.76 (0.9704)	6.90 (0.9724)	6.87 (0.9621)	5.72 (0.9697)
GrAPL	3.74 (0.9638)	5.67 (0.9694)	5.31 (0.9696)	5.74 (0.9679)	6.88 (0.9674)	6.85 (0.9621)	5.70 (0.9667)
HASTE (Ours)	**4.27 (0.9736)**	**6.06 (0.9819)**	**5.71 (0.9837)**	**6.17 (0.9801)**	**7.18 (0.9816)**	**7.18 (0.9825)**	**6.10 (0.9806)**

5.4 Decoupled Vision-Language Evaluator Query Savings

Image generation speedup alone does not capture the full evaluation cost on DSG, where multiple propositions share each generated image but require separate VLM judgments. We therefore measure whether stopping a decided proposition also reduces VLM queries while other propositions for the same prompt continue to use the shared image stream. The last proposition to stop determines image generation cost, whereas each proposition's own stopping time determines its VLM query cost.

Table 3 shows a mean VLM query speedup of 6.10× across six T2I models, compared with a mean image generation speedup of 2.60×. These figures count images and queries; runtime also depends on their respective costs. Because image generation and VLM evaluation latencies vary across GPU architectures, on our NVIDIA H200 GPUs, total GPU time for the six-model suite falls from about 513 to 157 hours (3.26×). Per-model breakdowns of wall-clock time appear in Appendix E.4.

5.5 Proposition Stopping-Time Distributions Across Generators

To examine how individual conditions resolve across sequential sampling steps, we analyze the proposition-level stopping-time distribution under **HASTE** across all 8,182 proposition nodes on DSG ($K = 50$, $\tau = 0.5$) for the six evaluated generators (Figure 3). Each proposition node v terminates at step $T_v \in \{n_{\min}, \ldots, K\}$, defined as the earliest observation index where the posterior probability crosses the stopping boundaries in Eq. (13). Across all generators, the empirical distribution exhibits a pronounced peak at the sample gate $t = n_{\min} = 5$: conditions with decisive visual evidence (such as unanimous successes or failures) satisfy posterior confidence immediately at the safety threshold. High-fidelity generators resolve a large majority of propositions at this first admissible step (85.6% for FLUX.2-dev and 77.2% for SANA-1.5), whereas SD 1.5 terminates 53.6% at $t = 5$ due to higher semantic ambiguity.

Stopping times vary across T2I models because generation quality determines which conditions are clear or ambiguous. **HASTE** stops clear propositions early and reserves the full $K = 50$ budget for 1.9%–7.3% of borderline cases, yielding a mean stopping count of $\bar{K} = 8.33$ and a 6.00× node-level query speedup. These results show that **HASTE** generalizes across T2I architectures without model-specific tuning.

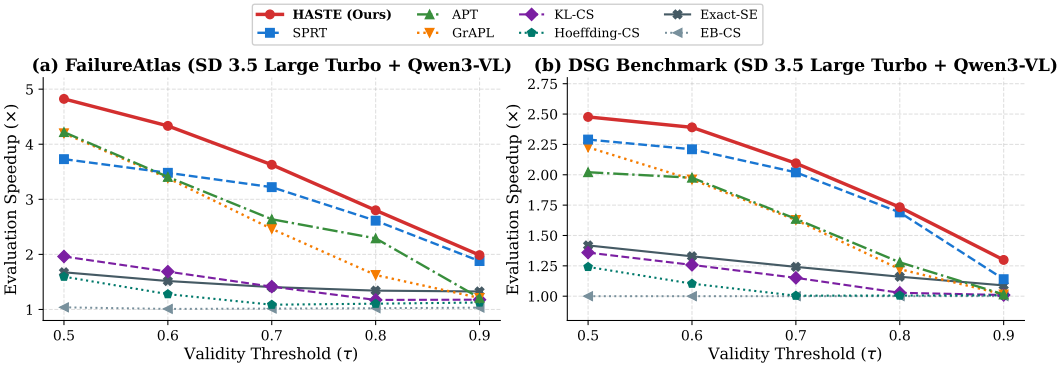


Figure 4: Evaluation speedup versus validity threshold $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ across eight sequential evaluation methods on Stable Diffusion 3.5 Large Turbo with Qwen3-VL-8B. Across both deep condition DAGs and shared-image question trees, **HASTE** consistently achieves the highest speedup across all tested validity thresholds.

Table 4: Component ablation on FailureAtlas using SD 3.5 Large Turbo. Bold and underline indicate the best and second-best results; red marks severe fidelity loss.

	Algorithmic Elements		Evaluation Performance		
Configuration	Hier. Prior	Sample Gate	Speedup ↑	Jaccard ↑	Agreement ↑
(a) Naive Flat ($\mathrm{Beta}(1,1), n_{\min} = 1$)			4.03×	0.8365	0.9408
(b) Unconstrained Hierarchical	✓		**6.15**×	0.8482	0.9689
(c) Flat Sequential ($\mathrm{Beta}(1,1)$)		✓	4.89×	<u>0.9537</u>	<u>0.9912</u>
(d) **HASTE**	✓	✓	<u>4.92×</u>	**0.9823**	**0.9967**

5.6 Evaluation Efficiency Across Validity Thresholds

We evaluate how the validity threshold affects image generation speedup by varying $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ on FailureAtlas and DSG. We compare **HASTE** with seven sequential evaluation baselines using Stable Diffusion 3.5 Large Turbo, Qwen3-VL-8B, and a fixed reference budget of $K = 50$. **HASTE** has the highest image generation speedup among the methods in Figure 4 at each tested threshold on both benchmarks. Its speedup ranges from $1.99\times$ to $4.82\times$ on FailureAtlas and from $1.30\times$ to $2.48\times$ on DSG.

5.7 Ablation Study

Table 4 evaluates the algorithmic components of **HASTE** (hierarchical prior propagation and minimum-sample gating $n_{\min}$) on FailureAtlas with Stable Diffusion 3.5 Large Turbo.

Without the minimum-sample gate, the flat and hierarchical variants achieve failure-set Jaccard scores of only 0.8365 and 0.8482, even though the hierarchical variant reaches a $6.15\times$ speedup. Adding the gate raises the flat variant's Jaccard to 0.9537. Combining the gate with hierarchical conditioning yields a $4.92\times$ speedup, 0.9823 Jaccard, and 0.9967 agreement. With parent weighting removed ($\rho = 0$), speedup falls to $4.45\times$ and Jaccard to 0.9712. These results show that the gate reduces premature stopping, while parent-informed conditioning improves both speed and fidelity.

6 Conclusion

We presented **HASTE**, an adaptive sequential evaluator for diagnostic T2I benchmarks. **HASTE** first organizes evaluation conditions into a graph representation, then allocates repeated observations according to condition-specific evidence. It uses completed parent summaries without treating them as child observations, and retains the benchmark's empirical decision at the fixed cap. This formulation exposes the trade-off between image cost and failure-set fidelity across both independently generated and shared-image conditions. Optional frontier scheduling can further prioritize FailureAtlas nodes, but the central result is that fixed-budget diagnostic evaluation need not spend the same number of generations on every decision.

AI USE STATEMENT

In this work, large language models (LLMs) and vision–language models (VLMs) were utilized for automated benchmark evaluation, theoretical proof assistance, and manuscript preparation:

- **Automated condition evaluation:** Vision–language models (including Qwen3-VL (Bai et al., 2025), BLIP-1 and BLIP-2 (Li et al., 2022; 2023), and InternVL3.5 (Wang et al., 2025)) served as automated evaluators providing per-image boolean verifications for diagnostic text-to-image benchmarks (Section 5 and Appendix K). The fidelity and reliability of these VLM-assigned judgments were verified through a blinded human audit conducted with six graduate students (Appendix J), evaluated under standard inter-annotator agreement metrics (Fleiss, 1971).
- **Theoretical proof assistance:** Large language models were consulted to check intermediate algebra, formatting, and boundary conditions during the derivation of Proposition 1 in Appendix E. The authors verified all mathematical statements and proofs independently.
- **Writing assistance:** LLMs assisted with phrasing refinement, grammatical polishing, and LaTeX formatting suggestions.

The authors have reviewed all AI-assisted outputs and take full responsibility for the final text, code, proofs, and experimental claims presented in this work.

ETHICS STATEMENT

This research focuses on accelerating evaluation and improving diagnostic assessment for text-to-image generative models. All empirical evaluations utilize standard, publicly available research benchmarks (FailureAtlas (Chen et al., 2025), DSG (Cho et al., 2024), and T2I-CoReBench (Li et al., 2026)) and open-weight generative models (including Stable Diffusion 1.5 (Rombach et al., 2022), SDXL and SDXL-Turbo (Podell et al., 2024; Sauer et al., 2024), Stable Diffusion 3.5 (Esser et al., 2024), SANA-1.5 (Xie et al., 2025), FLUX.2-dev (Black Forest Labs, 2025), and Qwen-Image (Wu et al., 2025)). Evaluation conditions examine benign visual attributes (such as spatial relations, entity presence, count verification, and attribute binding) without generating, assessing, or disseminating sensitive, private, or defamatory content. The human audit (Appendix J) involved voluntary graduate students evaluating non-sensitive visual benchmark images, with no collection of personally identifiable information.

REPRODUCIBILITY STATEMENT

The complete algorithmic procedure for **HASTE** is presented in Algorithm 1 (Appendix B). Theoretical proofs for the finite-budget stopping bound and expected cost formulations are provided in Appendix E and Appendix E.1. Comprehensive benchmark setups, hyperparameter specifications (κ, $n_{\min}$, ϵ, and validity threshold $\tau = 0.5$ following Cho et al. (2024)), and random seed controls are documented in Appendix G and Appendix G.1. Full evaluation traces across all testbeds (Chen et al., 2025; Cho et al., 2024; Li et al., 2026), baseline replay logs, and source code will be made publicly available upon publication.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, et al. Qwen3-VL technical report, 2025. URL https://arxiv.org/abs/2511.21631.

Black Forest Labs. FLUX.2: Next-generation latent flow matching models for text-to-image generation. https://bfl.ai/news/flux-2, 2025. Accessed: 2026-09-01.

Muxi Chen, Yu Li, and Qiang Xu. HiBug: On human-interpretable model debug. In *NeurIPS*, volume 36, pp. 4753–4766, 2023. doi: 10.52202/075280-0212.

Muxi Chen, Yi Liu, Jian Yi, Changran Xu, Qiuxia Lai, Hongliang Wang, Tsung-Yi Ho, and Qiang Xu. Evaluating text-to-image generative models: An empirical study on human image synthesis. *arXiv preprint arXiv:2403.05125*, 2024.

Muxi Chen, Zhaohua Zhang, Chenchen Zhao, Mingyang Chen, Wenyu Jiang, Tianwen Jiang, Jianhuan Zhuo, Yu Tang, Qiuyong Xiao, Jihong Zhang, and Qiang Xu. FailureAtlas: Mapping the failure

landscape of T2I models via active exploration. *arXiv preprint arXiv:2509.21995*, 2025. URL https://arxiv.org/abs/2509.21995.

Jaemin Cho, Yushi Hu, Roopal Garg, Peter Anderson, Ranjay Krishna, Jason Baldridge, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian Scene Graph: Improving reliability in fine-grained evaluation for Text-to-Image generation. In *ICLR*, 2024.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024.

Sabri Eyuboglu, Maya Varma, Khaled Saab, Jean-Benoit Delbrouck, Christopher Lee-Messer, Jared Dunnmon, James Zou, and Christopher Ré. Domino: Discovering systematic errors with cross-modal embeddings. In *ICLR*, 2022.

Alvan R. Feinstein and Domenic V. Cicchetti. High agreement but low kappa: I. The problems of two paradoxes. *Journal of Clinical Epidemiology*, 43(6):543–549, 1990. doi: 10.1016/0895-4356(90)90077-C.

Joseph L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76 (5):378–382, 1971. doi: 10.1037/h0031619.

Irena Gao, Gabriel Ilharco, Scott Lundberg, and Marco Tulio Ribeiro. Adaptive testing of computer vision models. In *ICCV*, pp. 4003–4014, 2023.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *NeurIPS*, volume 36, pp. 52132–52152, 2023.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021. doi: 10.1214/20-AOS1991.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. TIFA: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *ICCV*, pp. 20406–20417, 2023.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench: A comprehensive benchmark for open-world compositional text-to-image generation. In *NeurIPS*, volume 36, pp. 78723–78747, 2023.

J. Richard Landis and Gary G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977. doi: 10.2307/2529310.

Daniel LeJeune, Gautam Dasarathy, and Richard G. Baraniuk. Thresholding graph bandits with GrAPL. In *AISTATS*, pp. 2476–2485, 2020.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, volume 162, pp. 12888–12900. PMLR, 2022.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, pp. 19730–19742. PMLR, 2023.

Ouxiang Li, Yuan Wang, Xinting Hu, Huijuan Huang, Rui Chen, Jiarong Ou, Xin Tao, Pengfei Wan, Xiaojuan Qi, and Fuli Feng. Easier painting than thinking: Can text-to-image models set the stage, but not direct the play? In *ICLR*, 2026. URL https://openreview.net/forum?id=iqAFhWistW.

Andrea Locatelli, Maurilio Gutzeit, and Alexandra Carpentier. An optimal algorithm for the thresholding bandit problem. In *ICML*, pp. 1690–1698, 2016.

Volodymyr Mnih, Csaba Szepesvári, and Jean-Yves Audibert. Empirical Bernstein stopping. In *ICML*, pp. 672–679. ACM, 2008. doi: 10.1145/1390156.1390241.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

Yuning Qiu, Andong Wang, Chao Li, Haonan Huang, Guoxu Zhou, and Qibin Zhao. STEPS: Sequential probability tensor estimation for Text-to-Image hard prompt search. In *CVPR*, pp. 28640–28650, 2025. doi: 10.1109/CVPR52734.2025.02667.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pp. 10684–10695, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, volume 35, pp. 36479–36494, 2022.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *ECCV*, pp. 87–103, 2024. doi: 10.1007/978-3-031-73016-0_6.

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

Weiyun Wang, Zhe Chen, Wenhai Wang, Yue Cao, Yang Liu, Zhangwei Gao, Shenglong Ye, Xizhou Zhu, Lewei Lu, Tong Lu, Dahua Lin, Jifeng Dai, and Yu Qiao. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-Image technical report. *arXiv preprint arXiv:2508.02324*, 2025.

Enze Xie, Junsong Chen, Yuyang Zhao, Jincheng YU, Ligeng Zhu, Yujun Lin, Zhekai Zhang, Muyang Li, Junyu Chen, Han Cai, Bingchen Liu, Daquan Zhou, and Song Han. SANA 1.5: Efficient scaling of training-time and inference-time compute in linear diffusion transformer. In *ICML*, 2025.

Cong Xu, Gayathri Saranathan, Mahammad Parwez Alam, Arpit Shah, James Lim, Soon Yee Wong, Martin Foltin, and Suparna Bhattacharya. Data efficient evaluation of large language models and Text-to-Image models via adaptive sampling. *arXiv preprint arXiv:2406.15527*, 2024. URL https://arxiv.org/abs/2406.15527.

Linqi Zhao, Shengyu Zhang, Jiawei Dong, Haozhe Yang, Lele Chen, Zhaowei Zheng, Lingxuan Zhu, Meng Shen, Yujia Wang, Zhichao Wang, et al. FlashEval: Towards fast and accurate evaluation of text-to-image diffusion generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9456–9465, 2024.

Appendix

Appendix outline

A Glossary **14**

B HASTE Sequential Procedure **14**

C T2I-CoReBench: Generalization to Flat Checklists without Prerequisite Graphs **16**
C.1 Benchmark Specifications and Dual Cost Dynamics 16
C.2 Sequential Baseline Comparison on Empty-Edge Checklists 17
C.3 Analysis: Method Decomposition across Structural Layers 18
C.4 Cross-Evaluator Comparison on T2I-CoReBench: InternVL3.5-14B vs. Qwen3-VL-8B 18

D Generalization Across Proprietary Generators and Commercial Evaluators **19**
D.1 Sequential Evaluation on Proprietary T2I: GPT-Image-1.5 19
D.2 Sequential Evaluation with Commercial Closed-Source VLM: GPT-5.4-mini 20

E Theoretical Analysis and Baseline Comparison **21**
E.1 Finite-Budget Stopping and Evaluation Cost 21
E.2 Comparison with Multi-Armed Bandit and Graph Learning Baselines 23
E.3 Theoretical Role of Minimum Sample Requirements and Directional Likelihood Safeguards 23
E.4 Detailed Image Cost, VLM Scoring Compute, and Hardware Timings 24

F Training-data Attribution **26**

G Implementation Details **26**
G.1 Random Seeds and Deterministic Evaluation Protocol 27
G.2 Benchmark Testbed Specifications and DAG Construction 27
G.3 Evaluation Metrics and Comparison Protocol 28
G.4 Text-to-Image Generation Specifications 28
G.5 Vision-Language Model (VLM) Evaluation Protocol 28
G.6 HASTE Hyperparameters and Configurations 29
G.7 Prior Hyperparameter Exploration and Grid Search 29
G.8 Sequential Baselines and Implementation Details 29
G.9 Compute Hardware 31

H Comparative Analysis of Sequential Baselines and Cost–Fidelity Trade-offs **31**
H.1 Lossless Certificates versus Calibrated Approximate Evaluation 31
H.2 Exact-SE versus HASTE: Distinct Objectives and Cost Regimes 32
H.3 Pareto Optimality Among Approximate Sequential Rules 32
H.4 Statistical Context: Reference Variance, Evaluator Noise, and Threshold Concentration 32
H.5 Online End-to-End Early Stopping versus Offline Replay on FailureAtlas 33
H.6 Prompt-Level Subset Selection (FlashEval) versus Condition-Adaptive Stopping . . . 34

I Reference-budget Sensitivity **34**

J Human Audit of Vision-Language Model Judgments **34**

K Effect of the Vision-Language Model Evaluator **36**

L FailureAtlas Frontier Scheduling **39**

M Evaluation-state transitions **41**

N Limitations **41**

A GLOSSARY

Table 5 summarizes the variables, symbols, and notation used throughout this paper.

Table 5: Glossary of variables and symbols used in this paper.

Symbol / Notation	Description
Condition Graphs and Diagnostic Formulation	
$G = (V, E)$	DAG of evaluation conditions and prerequisite dependencies.
$v \in V$	Condition node (e.g., entity presence, attribute binding, relation).
$\mathrm{pa}(v)$	Prerequisite parent condition set, $\{u \in V : (u, v) \in E\}$.
$V_{\mathrm{root}}, V_{\mathrm{eval}}$	Root condition set ($\mathrm{pa}(v) = \varnothing$) and evaluated condition set.
$x, x_v \in \mathcal{X}$	Input text prompt for condition v, and prompt universe $\mathcal{X}$.
V_x	Conditions sharing prompt x under shared image streams.
Generative Modeling and Verification Contract	
$\mathcal{M}, \mathcal{E}$	T2I generator under test and automated VLM evaluator.
$I_{v,j} \sim \mathcal{M}(\cdot \mid x_v)$	The j-th independently generated image sample for node v.
$y_{v,j} \in \{0, 1\}$	Binary VLM judgment on whether image $I_{v,j}$ satisfies condition v.
θ_v	Evaluator success probability $\Pr(y_{v,j} = 1)$ of condition v.
K, τ	Sample budget cap ($K = 50$) and validity threshold ($\tau = 0.5$).
$\widehat{\theta}_v$	Full-budget empirical success rate, $(1/K) \sum_{j=1}^{K} y_{v,j}$.
d_v^{ref}	Reference decision: VALID if $\widehat{\theta}_v \geq \tau$, else FAILURE.
Adaptive Sequential Testing and Prior Propagation (HASTE)	
t, T_v	Observation slot index and stopping time $T_v \leq K$ for node v.
$d_v, d_{v,t}$	Final sequential decision and intermediate state at slot t.
s_t, f_t, a_t	Cumulative successes, failures, and total observations $a_t = s_t + f_t$.
$\mathcal{D}_t$	Cumulative observation evidence through slot t, (s_t, f_t).
m_u, g_v	Parent posterior mean and bottleneck summary $\min_{u \in \mathrm{pa}(v)} m_u$.
μ_v, μ_{g}	Graph prior mean $\rho g_v + (1 - \rho)\mu_{\mathrm{g}}$ and neutral mean $\mu_{\mathrm{g}} = 0.5$.
ρ, κ	Parent prior weight $\rho \in [0, 1]$ and prior sample strength $\kappa > 0$.
α_v, β_v	Beta prior parameters $(\kappa\mu_v, \kappa(1 - \mu_v))$, set to $(1, 1)$ at roots.
$p_{v,t}$	Posterior validity probability, $1 - I_\tau(\alpha_v + s_t, \beta_v + f_t)$.
$I_\tau(\alpha, \beta)$	Regularized incomplete beta function (Beta CDF at τ).
$n_{\min}, \epsilon$	Minimum sample gate ($n_{\min} = 5$) and error tolerance ($\epsilon = 0.05$).
Theoretical Bounds and Analysis	
$\mu_{L,v}, \mu_{U,v}$	Deterministic lower and upper bounds on prior mean μ_v.
$b_{v,t}^-, b_{v,t}^+$	Deterministic lower (failure) and upper (valid) stopping thresholds.
$B_{v,t}, q_v(t)$	Binomial variable $\mathrm{Binomial}(t, \theta_v)$ and continuation probability.
Evaluation Complexity and Fidelity Metrics	
$C_{\mathrm{image}}, C_{\mathrm{vlm}}$	Physical image generation count and VLM scoring query count.
$S_{\mathrm{image}}, S_{\mathrm{vlm}}$	Image generation speedup and VLM scoring query speedup.
F, F_{ref}	Discovered sequential failure set and reference failure set.
$J(F, F_{\mathrm{ref}})$	Failure-set Jaccard agreement, $\lvert F \cap F_{\mathrm{ref}} \rvert / \lvert F \cup F_{\mathrm{ref}} \rvert$.
A_{node}	Overall binary node decision agreement across all conditions.

B HASTE SEQUENTIAL PROCEDURE

Algorithm 1 gives the node-level procedure from Sec. 4.2. **HASTE** waits for all concrete parents before constructing the prior, then processes the selected condition one logical slot at a time. Every attempted slot counts toward cost, although missing and parse-failed judgments do not update the Beta–Bernoulli posterior. For intermediate steps $t < K$, a valid or failure boundary triggers early stopping once its minimum sample requirement is satisfied and posterior confidence is established. Otherwise, **HASTE** applies the empirical rule at cap K over the identical sample sequence, guaranteeing exact agreement with the full-budget reference estimator. Figure 5 illustrates these three operational outcomes on representative benchmark conditions. The decision, stopping slot, and trace record the evidence used at termination.

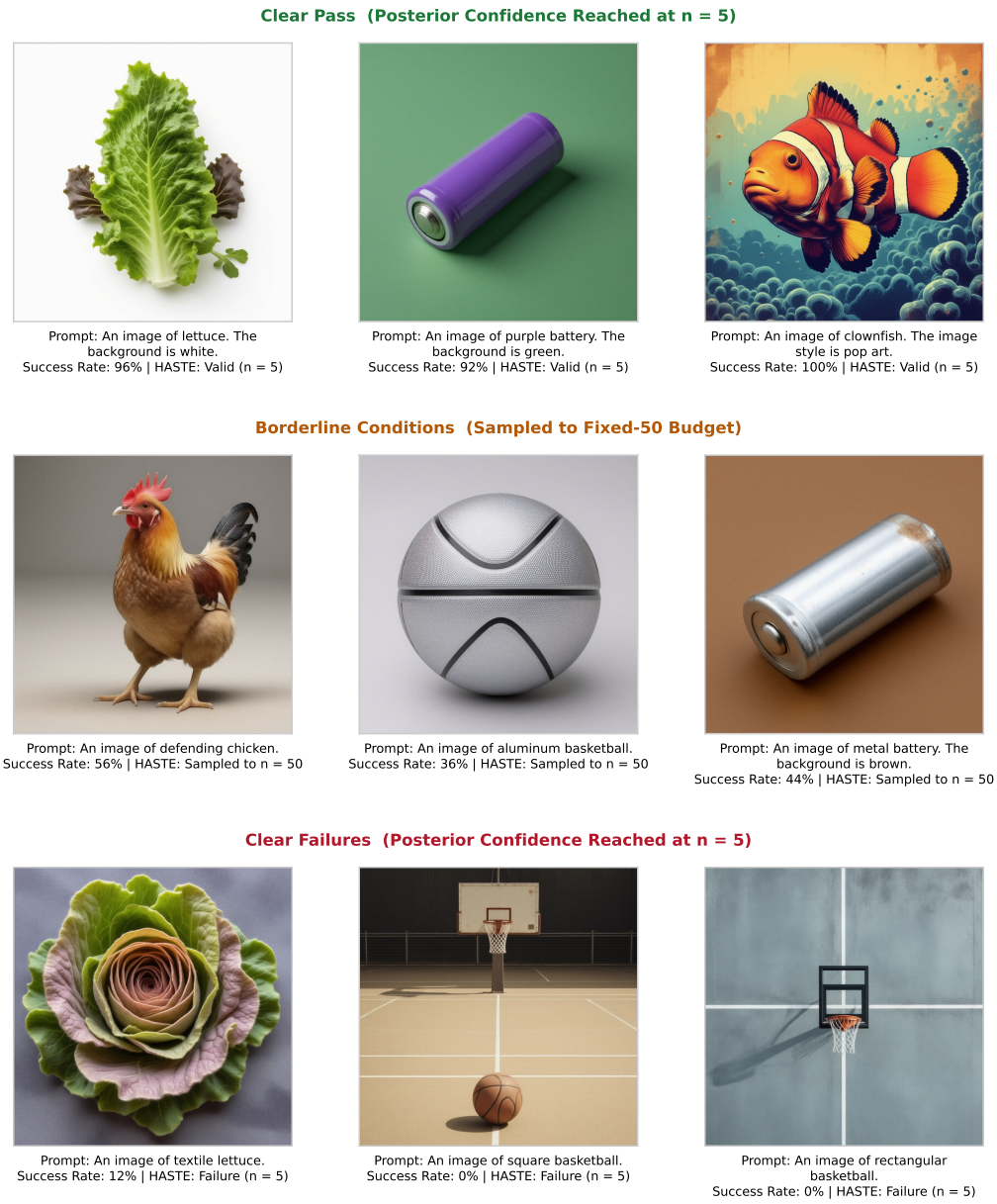


Figure 5: Qualitative examples from the FailureAtlas SD 3.5 Large Turbo evaluation suite across stopping strata. *Top row (Clear Pass, pass rate* $\geq 88\%$*):* Decisive conditions with compliant attributes achieve posterior confidence at the minimum sample gate $n_{\min} = 5$, concluding as VALID while saving 90% of generation compute. *Middle row (Borderline, pass rate* 36%–64%*):* Ambiguous conditions near the decision boundary ($\tau = 0.5$) remain within the indifference region and are safely allocated the full budget ($K = 50$) under HASTE, preventing classification flips. *Bottom row (Clear Fail, pass rate* $\leq 12\%$*):* Geometric violations (such as square or rectangular basketballs) and incompatible compositions terminate as FAILURE at $n_{\min} = 5$.

Algorithm 1: HASTE Sequential Testing of One Condition

Require: Condition v, completed parent summaries, threshold τ, cap K
Ensure: Decision d_v, stopping slot T_v, and an auditable trace

```
Construct (α_v, β_v) from the root prior or Eq. (10)
Initialize s ← 0, f ← 0
for t = 1, ..., K do
    Consume the next logical slot
    if the outcome is an accepted Boolean then
        Update s or f and compute p_{v,t} by Eq. (12)
        if t < K and s + f ≥ n_min and p_{v,t} ≥ 1 − ε then
            return (VALID, t)
        else if t < K and s + f ≥ n_min and p_{v,t} ≤ ε then
            return (FAILURE, t)
        end if
    end if
    Append the slot and posterior state to the trace
end for
if s/K ≥ τ then
    return (VALID,K)
else
    return (FAILURE,K)
end if
```

C T2I-CoReBench: Generalization to Flat Checklists without Prerequisite Graphs

In structured diagnostic benchmarks such as DSG and FailureAtlas, evaluation conditions are organized into directed acyclic graphs encoding prerequisite semantic dependencies. However, in practical evaluation scenarios, constructing parent–child prerequisite graphs can be labor-intensive, computationally demanding, and inherently subjective across different human annotators or automated prompt-parsing pipelines. When users choose not to construct a dependency graph, or when a diagnostic benchmark does not provide explicit prerequisite dependencies, each evaluation condition can be formulated as an independent node in an empty-edge graph ($E = \varnothing$).

Even in this empty-edge setting, **HASTE** remains directly applicable: every node is evaluated sequentially using Algorithm 1 under the neutral, uninformative root prior $\mathrm{Beta}(1, 1)$, without requiring parent-conditioned prior propagation. In this configuration, the hierarchical prior layer naturally operates without parent–child information sharing. Demonstrating that **HASTE** preserves substantial computational speedups and high decision fidelity on such flat benchmarks confirms the universal applicability of **HASTE** across arbitrary text-to-image diagnostic benchmarks, completely independent of whether a prerequisite graph structure is constructed.

C.1 Benchmark Specifications and Dual Cost Dynamics

T2I-CoReBench (Li et al., 2026) provides a flat checklist benchmark of this form, containing no released dependencies among questions. We evaluate on the complete released benchmark: 1,080 prompts spanning four composition dimensions (e.g., spatial relations, attribute binding, action binding, complex multi-entity composition) and eight reasoning dimensions (e.g., counting, negation, comparison, physical commonsense, visual reasoning), with exactly 90 prompts per dimension. In total, the benchmark comprises 13,536 binary checklist questions.

For each prompt, image index j identifies one shared generation that is scored against every checklist question associated with that prompt. Because multiple questions share the same generated image, stopping question v after slot T_v does not remove an image until every still-active question for that prompt has reached a stopping decision. Consequently, two distinct cost metrics emerge:

$$C_{\text{image}} = \max_{v \in V} T_v, \qquad C_{\text{eval}} = \sum_{v \in V} T_v. \tag{16}$$

Table 6: Full T2I-CoReBench $K = 50$ root-only replay across six text-to-image generators (1,080 items, 54,000 reference images, 676,800 question–image queries per model). All conditions operate with the neutral root prior $\mathrm{Beta}(1, 1)$. Estimated end-to-end GPU hours reflect measured benchmarks on NVIDIA H200 SXM GPUs running Qwen3-VL-8B evaluation (0.056s per query).

Generator Model	Image Cost (Speedup)	VLM Queries (Speedup)	Fixed GPU-h	**HASTE** GPU-h (E2E Speedup)	Fail. Jaccard	Prop. Agree.
SD 1.5	30,957 (1.74×)	128,617 (5.26×)	18.78h	6.73h (**2.79**×)	0.9832	0.9944
SD 3.5 Large Turbo	27,281 (1.98×)	110,951 (6.10×)	19.83h	6.42h (**3.09**×)	0.9848	0.9967
SDXL-Turbo	24,200 (2.23×)	99,096 (6.83×)	13.83h	3.02h (**4.58**×)	0.9842	0.9963
SANA-1.5 1.6B	22,986 (2.35×)	100,433 (6.74×)	23.28h	6.99h (**3.33**×)	0.9842	0.9974
FLUX.2-dev	16,984 (3.18×)	84,543 (8.01×)	88.53h	25.85h (**3.43**×)	0.9912	0.9990
Qwen-Image	17,713 (3.05×)	86,301 (7.84×)	382.53h	123.37h (**3.10**×)	0.9892	0.9987
Cross-Model Mean	23,354 (2.42×)	101,657 (6.80×)	546.77h	172.38h (**3.17**×)	0.9862	0.9971

Table 7: Comparison of all eight sequential evaluation baselines on full T2I-CoReBench $K = 50$ across six text-to-image generators (1,080 items, 54,000 reference images, 676,800 question–image queries per model, $\tau = 0.5$). Bold indicates the highest speedup in each column.

	Image Generation Speedup						**Cross-Model Summary**		
Method	SD 1.5	SD 3.5	SDXL	SANA	FLUX.2	Qwen-Img	Avg. Img.	Avg. Query	Fail. Jacc.
Exact-SE	1.25×	1.32×	1.36×	1.37×	1.53×	1.50×	1.39×	1.82×	1.0000
Hoeffding-CS	1.08×	1.14×	1.18×	1.20×	1.40×	1.36×	1.23×	2.08×	1.0000
EB-CS	1.00×	1.00×	1.00×	1.00×	1.00×	1.00×	1.00×	1.00×	1.0000
KL-CS	1.11×	1.21×	1.26×	1.29×	1.61×	1.53×	1.34×	2.88×	1.0000
SPRT	1.53×	1.74×	1.86×	1.95×	2.52×	2.40×	2.00×	4.26×	0.9977
APT	1.54×	1.77×	1.91×	2.05×	2.78×	2.60×	2.11×	6.24×	0.9924
GrAPL	1.54×	1.77×	1.91×	2.05×	2.78×	2.60×	2.11×	6.24×	0.9924
HASTE (Ours)	**1.74**×	**1.98**×	**2.23**×	**2.35**×	**3.18**×	**3.05**×	**2.42**×	**6.80**×	0.9862

The prompt-level image generation cost C_{image} is dictated by the single slowest, most ambiguous question in the prompt. Conversely, vision-language model scoring cost C_{eval} scales with the actual number of evaluated question–image pairs. Evaluating the fixed-50 reference requires 54,000 images and $13{,}536 \times 50 = 676{,}800$ evaluator queries per model.

Replay results across T2I generators. Table 6 presents the full T2I-CoReBench $K = 50$ root-only **HASTE** replay across six text-to-image models, evaluating all 1,080 items, 54,000 generated images, and 676,800 question–image queries per model. Across all architectures, root-only **HASTE** maintains high decision fidelity (failure-set Jaccard > 0.983 and proposition agreement > 0.994) while achieving image generation speedups between $1.74\times$ and $3.18\times$, VLM query reductions up to $8.01\times$, and measured end-to-end evaluation speedups between $2.79\times$ and $4.58\times$ on NVIDIA H200 SXM GPUs.

C.2 Sequential Baseline Comparison on Empty-Edge Checklists

To systematically assess alternative sequential algorithms on flat checklist benchmarks, we evaluate all eight sequential baselines across the complete T2I-CoReBench benchmark for all six text-to-image models. Table 7 summarizes the results, highlighting several notable theoretical and empirical properties:

1. **GrAPL and APT Equivalence under Empty Graphs**: In the absence of prerequisite dependencies ($E = \varnothing$), the graph Laplacian matrix vanishes identically ($L = 0$). As shown in Table 7, the empirical performance of graph-regularized active sequential testing (GrAPL) matches standard Adaptive Posterior Thresholding (APT) exactly across all six models and all metrics down to machine precision. This confirms the theoretical expectation that GrAPL reduces to APT when no graph topology is available.
2. **Shared-Image Bottleneck on Conservative Baselines**: Because each prompt contains up to twenty questions evaluated on the same image stream, the prompt-level image generation cost is governed by the slowest condition ($C_{\text{image}} = \max_{v \in V} T_v$). Highly conservative confidence-sequence baselines (Empirical Bernstein CS and Hoeffding CS) require large sample sizes before confidence intervals exclude $\tau = 0.5$. In particular, Empirical Bernstein CS fails to certify early

stopping on any question before slot 50, resulting in exactly $1.00\times$ image and query speedup. KL-CS and Exact-SE achieve modest image speedups ($1.34\times$ and $1.39\times$ average), but remain constrained by the $\max_v$ bottleneck.

3. **Consistent Dominance of HASTE**: **HASTE** achieves the highest image generation speedup ($2.42\times$ average) and the highest query speedup ($6.80\times$ average) across all six text-to-image architectures, while maintaining high failure-set fidelity (0.9862 average failure Jaccard). By combining sequential posterior thresholding with a finite-horizon fallback, **HASTE** stops decisive conditions early while avoiding over-sampling on borderline questions.

C.3 Analysis: Method Decomposition across Structural Layers

The T2I-CoReBench findings provide experimental evidence for three structural characteristics of **HASTE**:

1. **Bounded Sequential Evaluator**: Determines the sample requirement for an individual condition using sequential posterior updates, minimum sample requirements, posterior confidence boundaries, and the empirical finite-cap fallback. This layer requires only stochastic binary observations and a validity threshold; it does not depend on graphical relationships.
2. **Graph-Conditioned Prior Propagation**: Collects posterior summaries from completed parents to construct an informative prior for child conditions. This layer operates when explicit dependency edges exist.
3. **Active Frontier Policy**: Orders independently schedulable condition candidates under a cumulative budget to discover failure states efficiently.

For T2I-CoReBench, our graph construction produces $E = \varnothing$, so evaluation uses Layer 1 with a root prior at every node. Layers 2 and 3 are inactive. This setting demonstrates that the sequential decision mechanism provides substantial acceleration even within an unlinked checklist representation without parent–child information sharing.

C.4 Cross-Evaluator Comparison on T2I-CoReBench: InternVL3.5-14B vs. Qwen3-VL-8B

To evaluate whether the adaptive stopping efficiency and fidelity of **HASTE** on flat checklists transfer across distinct vision-language evaluators, we conducted full fixed-50 T2I-CoReBench evaluations (1,080 prompts, 54,000 images, 676,800 question–image queries per model) using `OpenGVLab/InternVL3_5-14B-Instruct` (Wang et al., 2025) served via vLLM on NVIDIA H200 SXM GPUs. Table 8 matches the completed evaluations for FLUX.2-dev, Stable Diffusion 3.5 Large Turbo, and SDXL-Turbo directly against the primary `Qwen3-VL-8B-Instruct` evaluations on identical generated images.

Table 8: Cross-evaluator comparison on full T2I-CoReBench ($K = 50$, $\tau = 0.5$, 1,080 items, 54,000 reference images) between `Qwen3-VL-8B` and `InternVL3.5-14B`.

T2I Generator	VLM Evaluator	Fixed-K Score	Image Cost	Speedup ↑	Prop. Agreement ↑	Fail. Jaccard ↑	Item Agreement ↑
FLUX.2-dev	Qwen3-VL-8B	0.8019	16,984	3.18×	0.9990	0.9912	0.9870
	InternVL3.5-14B	0.7861	15,577	**3.47**×	0.9992	**0.9935**	0.9907
SD 3.5-Turbo	Qwen3-VL-8B	0.7200	27,281	1.98×	0.9967	0.9848	0.9611
	InternVL3.5-14B	0.6643	26,867	**2.01**×	0.9967	**0.9872**	0.9620
SDXL-Turbo	Qwen3-VL-8B	0.6994	24,200	**2.23**×	0.9963	0.9842	0.9565
	InternVL3.5-14B	0.6001	24,973	2.16×	0.9967	**0.9889**	0.9583

This cross-evaluator comparison demonstrates three key findings:

1. **Invariant Model Quality Ranking**: Both evaluators produce an identical generator ranking (FLUX.2-dev > SD 3.5 Large Turbo > SDXL-Turbo).
2. **Evaluator Strictness**: InternVL3.5-14B is systematically more conservative than Qwen3-VL-8B on fine-grained checklist questions (mean scores are 0.016 to 0.099 lower), reflecting stricter count and spatial bounding verification.
3. **Robust Adaptive Fidelity**: Despite differences in underlying pass rates, the adaptive stopping performance of **HASTE** remains consistent: across both evaluators, proposition agreement

Table 9: Sequential baseline evaluation on GPT-Image-1.5 ($K = 50$ on DSG) at validity threshold $\tau = 0.5$. **HASTE** achieves $3.49\times$ image speedup with 0.9981 failure-set Jaccard agreement relative to the full-budget reference.

Method	Speedup ($\times$)	Failure Jaccard
Exact-SE	1.57	1.0000
Hoeffding-CS	1.45	1.0000
EB-CS	1.00	1.0000
KL-CS	1.71	1.0000
GrAPL	2.31	0.8324
SPRT	3.01	0.9905
APT	3.27	0.9915
HASTE (Ours)	**3.49**	0.9981

exceeds 0.996, failure-set Jaccard exceeds 0.984, and image speedup rankings are preserved (FLUX.2-dev at $3.2\times$–$3.5\times$, SDXL-Turbo at $2.2\times$, SD 3.5 Large Turbo at $2.0\times$).

Benchmark scope. T2I-CompBench evaluates compositional T2I alignment, but its categories overlap with the scope already covered by DSG (Huang et al., 2023; Cho et al., 2024). We omit separate T2I-CompBench experiments to avoid duplicating this evaluation axis; this choice does not imply numerical transfer between benchmarks.

D Generalization Across Proprietary Generators and Commercial Evaluators

The core experiments in Section 5 benchmark sequential testing using open-weights generative backbones (e.g., SD 3.5 Large Turbo, SDXL, and FLUX.2-dev) paired with open-source vision-language model evaluators (e.g., Qwen2.5-VL-7B). Practical diagnostic and compliance auditing pipelines frequently interface with closed-source commercial systems, either generating samples via black-box APIs or scoring outputs through commercial foundation models. To verify that the sample efficiency and decision fidelity of **HASTE** generalize across these operational boundaries, we evaluate two complementary closed-source settings: (1) a proprietary text-to-image generator (GPT-Image-1.5) evaluated with an open-source visual judge, and (2) an open-source generator (SD 3.5 Large Turbo) evaluated by a frontier commercial vision-language model (Azure GPT-5.4-mini).

D.1 Sequential Evaluation on Proprietary T2I: GPT-Image-1.5

We first examine whether the computational efficiency and fidelity guarantees of **HASTE** transfer to proprietary generation architectures where internal latent representations and sampling dynamics are inaccessible. We benchmark eight sequential testing methods on GPT-Image-1.5 using the compositional DSG diagnostic suite at full budget cap $K = 50$, spanning 511 prompt items and 25,550 generated images. We evaluate the fixed-K complete benchmark decisions against eight sequential algorithms: Exact-SE, Hoeffding-CS, EB-CS, KL-CS, GrAPL, SPRT, APT, and **HASTE** across five deterministic observation seeds.

Table 9 reports the evaluation speedup and failure-set Jaccard index at standard threshold $\tau = 0.5$. Consistent with findings on open-weights generators in Table 2, conservative methods such as Exact-SE ($1.57\times$) and martingale confidence sequences (Hoeffding-CS at $1.45\times$, KL-CS at $1.71\times$, EB-CS at $1.00\times$) guarantee exact reference agreement ($J = 1.0000$) by construction, but their acceleration remains bounded. Heuristic approximate methods deliver higher speedups (SPRT at $3.01\times$, APT at $3.27\times$) with small fidelity compromises ($J \approx 0.991$), while GrAPL achieves $2.31\times$ but experiences substantial fidelity decay ($J = 0.8324$). In comparison, **HASTE** attains the highest speedup of $3.49\times$ while preserving near-complete failure-set fidelity (0.9981 Jaccard agreement).

Figure 6 plots the speedup trajectory as validity threshold τ varies from 0.5 to 0.9. As the stringency of the validity criterion increases, conditions require substantially more affirmative generations before sequential acceptance or rejection occurs, dampening early termination across all rules. Confidence

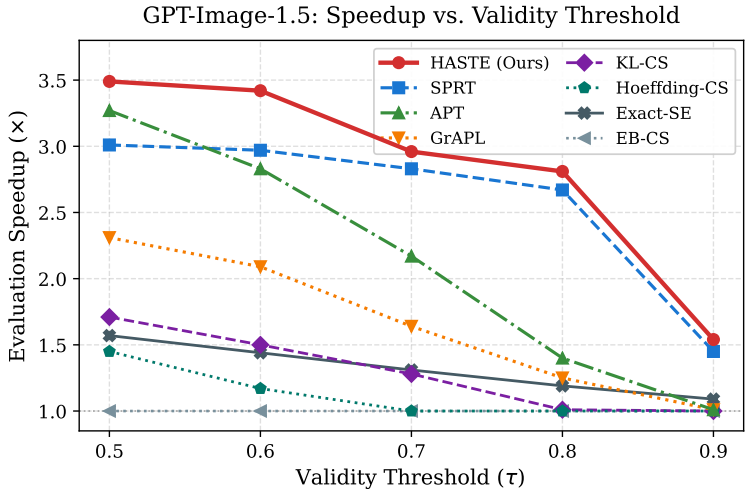


Figure 6: Evaluation speedup across validity thresholds $\tau \in [0.5, 0.9]$ on GPT-Image-1.5 ($K = 50$ on DSG). **HASTE** consistently maintains high acceleration across varying operating points, achieving $3.49\times$ at $\tau = 0.5$, $2.81\times$ at $\tau = 0.8$, and $1.54\times$ at $\tau = 0.9$.

Table 10: Sequential baseline evaluation using commercial VLM judge GPT-5.4-mini on SD 3.5 Large Turbo ($K = 50$ on DSG) at validity threshold $\tau = 0.5$. **HASTE** achieves a $2.14\times$ image speedup with 0.9977 failure-set Jaccard agreement relative to the full-budget reference.

Method	Speedup (×)	Failure Jaccard
Exact-SE	1.13	1.0000
Hoeffding-CS	1.02	1.0000
EB-CS	1.01	1.0000
KL-CS	1.09	1.0000
SPRT	1.18	0.9891
APT	1.35	0.9912
GrAPL	1.37	0.9623
HASTE (Ours)	**2.14**	**0.9977**

sequence methods become strictly inactive ($1.00\times$) for $\tau \geq 0.7$ because finite-K empirical sample paths cannot separate narrow confidence intervals from the threshold. APT decelerates sharply from $3.27\times$ at $\tau = 0.5$ to $1.40\times$ at $\tau = 0.8$ and $1.01\times$ at $\tau = 0.9$. GrAPL also collapses toward $1.01\times$ while its Jaccard agreement deteriorates severely to 0.6070 at $\tau = 0.9$. In contrast, **HASTE** delivers sustained acceleration across the operational spectrum, achieving $3.42\times$ at $\tau = 0.6$, $2.96\times$ at $\tau = 0.7$, $2.81\times$ at $\tau = 0.8$, and $1.54\times$ at $\tau = 0.9$. This demonstrated stability on proprietary generation architectures confirms that structured prior propagation provides consistent sample efficiency without sacrificing diagnostic fidelity.

D.2 Sequential Evaluation with Commercial Closed-Source VLM: GPT-5.4-mini

We next investigate whether **HASTE** remains robust when the evaluation mechanism itself is driven by a commercial closed-source vision-language model. Open-source visual evaluators can exhibit idiosyncratic scoring distributions or systematic classification biases. Evaluating with a distinct commercial-grade frontier judge tests whether sequential stopping decisions remain reliable when question-answering probabilities originate from an independent commercial API. We evaluate the eight sequential baseline algorithms on DSG ($K = 50$, $\tau = 0.5$) using images generated by SD 3.5 Large Turbo, scored by Microsoft Azure GPT-5.4-mini as the vision-language model judge.

Table 10 reports the evaluation speedup and failure-set Jaccard index under the GPT-5.4-mini evaluator. The empirical trends mirror the structural trade-offs observed across open-source configurations, demonstrating strong stability across model boundaries:

- **Conservative certified stopping**: Exact-SE ($1.13\times$) and the three martingale confidence sequence methods (Hoeffding-CS at $1.02\times$, EB-CS at $1.01\times$, KL-CS at $1.09\times$) maintain exact agreement ($J = 1.0000$) with the full-budget ground truth. However, because their worst-case finite-sample concentration bounds require substantial observation counts to separate empirical averages from the threshold, they exhaust almost the complete budget cap $K = 50$ on the majority of questions, providing marginal compute savings.
- **Heuristic sequential methods**: SPRT provides a $1.18\times$ speedup with 0.9891 Jaccard agreement. The adaptive threshold baseline APT achieves a higher speedup of $1.35\times$ with 0.9912 Jaccard fidelity. GrAPL attains a $1.37\times$ speedup, but incurs a noticeable degradation in diagnostic fidelity, dropping to 0.9623 Jaccard agreement. This decline indicates that heuristic rank-based sample allocation without structured prior propagation struggles with subtle failure conditions identified by stronger commercial evaluators.
- **HASTE**: **HASTE** attains a $2.14\times$ evaluation speedup—reducing total visual question answering calls by more than 53%—while maintaining a high failure-set Jaccard agreement of 0.9977.

These findings demonstrate that **HASTE** does not depend on particular model pairings or open-source artifacts. By propagating hierarchical dependency evidence and maintaining calibrated Bayesian posterior stopping margins, **HASTE** provides consistent acceleration and reliable failure discovery across both proprietary generative models and commercial evaluators.

E Theoretical Analysis and Baseline Comparison

We first bound **HASTE**'s finite-budget stopping cost under an explicit Bernoulli model, then compare its evaluation rule with sequential-testing and graph-based baselines.

E.1 Finite-Budget Stopping and Evaluation Cost

The following bound applies to the posterior stopping rule in Eq. (13) when every consumed slot has an accepted binary outcome. For each condition v, assume that its outcomes across image slots are i.i.d. $\mathrm{Bernoulli}(\theta_v)$, and write $S_{v,t} = \sum_{j=1}^{t} Y_{v,j}$. Outcomes for different conditions may be dependent.

Proposition 1 (Finite-budget stopping bound)**.** *Write the prior for condition v as $\mathrm{Beta}(\kappa_v\mu_v, \kappa_v(1-\mu_v))$, with fixed $\kappa_v > 0$. The parent-derived mean μ_v may depend on outcomes from the same images. Choose fixed bounds $0 < \mu_{L,v} \le \mu_v \le \mu_{U,v} < 1$ that hold for every outcome sequence, where $\mu_{L,v}$ and $\mu_{U,v}$ denote the lower and upper bounds on μ_v, respectively. For a root with prior $\mathrm{Beta}(\alpha_0, \beta_0)$, set $\kappa_v = \alpha_0 + \beta_0$ and both bounds to $\alpha_0/(\alpha_0+\beta_0)$. For a child with $\mu_v = \rho g_v + (1-\rho)\mu_{\mathrm{g}}$, $g_v \in [0,1]$, $0 \le \rho < 1$, and $0 < \mu_{\mathrm{g}} < 1$, one may take*

$$\mu_{L,v} = (1-\rho)\mu_{\mathrm{g}}, \qquad \mu_{U,v} = \rho + (1-\rho)\mu_{\mathrm{g}}. \tag{17}$$

For $n_{\min} \le t < K$, define

$$p_{v,t}(s,\mu) = 1 - I_\tau\big(\kappa_v\mu + s, \kappa_v(1-\mu) + t - s\big) \tag{18}$$

and the two success-count thresholds

$$\begin{aligned} b^-_{v,t} &= \max\{s \in \{0,\dots,t\} : p_{v,t}(s,\mu_{U,v}) \le \epsilon\}, \\ b^+_{v,t} &= \min\{s \in \{0,\dots,t\} : p_{v,t}(s,\mu_{L,v}) \ge 1-\epsilon\}. \end{aligned} \tag{19}$$

An empty set gives $b^-_{v,t} = -1$ or $b^+_{v,t} = t+1$, respectively. If $B_{v,t} \sim \mathrm{Binomial}(t, \theta_v)$, the stopping slot T_v satisfies

$$\Pr(T_v > t) \le \Pr(b^-_{v,t} < B_{v,t} < b^+_{v,t}),$$
$$\mathbb{E}[T_v] \le n_{\min} + \sum_{t=n_{\min}}^{K-1} \Pr(b^-_{v,t} < B_{v,t} < b^+_{v,t}). \tag{20}$$

The bounds on μ_v are used only in this analysis. They make $b^-_{v,t}$ and $b^+_{v,t}$ deterministic even when parent evidence and $S_{v,t}$ come from the same image stream.

Proof of Proposition 1. Fix $t \ge n_{\min}$. Let $f_{t,s,\mu}$ be the density of $\mathrm{Beta}(\kappa_v\mu + s, \kappa_v(1-\mu) + t - s)$. For $s' \ge s$ and $\mu' \ge \mu$, set $d = (s'-s) + \kappa_v(\mu'-\mu) \ge 0$. The density ratio is

$$\frac{f_{t,s',\mu'}(z)}{f_{t,s,\mu}(z)} = c\left(\frac{z}{1-z}\right)^d, \qquad c > 0, \quad 0 < z < 1. \tag{21}$$

The ratio increases with z, so $f_{t,s',\mu'}$ stochastically dominates $f_{t,s,\mu}$. Thus $p_{v,t}(s,\mu)$ is nondecreasing in both s and μ. For nonempty threshold sets,

$$\begin{aligned} S_{v,t} \le b^-_{v,t} &\implies p_{v,t}(S_{v,t}, \mu_v) \le p_{v,t}(b^-_{v,t}, \mu_{U,v}) \le \epsilon, \\ S_{v,t} \ge b^+_{v,t} &\implies p_{v,t}(S_{v,t}, \mu_v) \ge p_{v,t}(b^+_{v,t}, \mu_{L,v}) \ge 1-\epsilon. \end{aligned} \tag{22}$$

An empty threshold set makes its corresponding event impossible. Either displayed inequality triggers a stop by t, hence

$$\{T_v > t\} \subseteq \{b^-_{v,t} < S_{v,t} < b^+_{v,t}\}. \tag{23}$$

This inclusion holds for every admissible μ_v and does not require independence between μ_v and $S_{v,t}$. Since $S_{v,t} \sim \mathrm{Binomial}(t, \theta_v)$, it gives the first bound in Eq. (20). Finally, $T_v \le K$ and the tail-sum identity give

$$\mathbb{E}[T_v] = \sum_{t=0}^{K-1} \Pr(T_v > t) \le n_{\min} + \sum_{t=n_{\min}}^{K-1} \Pr(b^-_{v,t} < B_{v,t} < b^+_{v,t}). \tag{24}$$

□

Define $q_v(t) = 1$ for $t < n_{\min}$ and $q_v(t) = \Pr(b^-_{v,t} < B_{v,t} < b^+_{v,t})$ otherwise.

Shared image streams (DSG). For the fixed set V_x of propositions evaluated on prompt x, Eq. (20) and the union bound yield

$$\begin{aligned} \mathbb{E}[C_{\mathrm{vlm}}] &= \sum_{x\in\mathcal{X}} \sum_{v\in V_x} \sum_{t=0}^{K-1} \Pr(T_v > t) \le \sum_{x\in\mathcal{X}} \sum_{v\in V_x} \sum_{t=0}^{K-1} q_v(t), \\ \mathbb{E}[C_{\mathrm{image}}] &= \sum_{x\in\mathcal{X}} \sum_{t=0}^{K-1} \Pr\left(\bigcup_{v\in V_x} \{T_v > t\}\right) \\ &\le \sum_{x\in\mathcal{X}} \sum_{t=0}^{K-1} \min\left\{1, \sum_{v\in V_x} q_v(t)\right\}. \end{aligned} \tag{25}$$

Propositions may be dependent because they inspect the same images. Their VLM query costs add, while image cost depends on whether any proposition remains active at each slot.

Independent image streams (FailureAtlas). Let V be the condition-node universe and A_v indicate whether the graph traversal visits node v; set $A_v T_v = 0$ when $A_v = 0$. Let $\mathcal{F}_v$ be the history when the traversal resolves this visit decision. Assume that, on $A_v = 1$, node v's own image stream

remains i.i.d. $\mathrm{Bernoulli}(\theta_v)$ conditional on $\mathcal{F}_v$. Each consumed slot produces one image and one VLM judgment. With $U_v = \sum_{t=0}^{K-1} q_v(t)$, the per-node bound gives

$$\Pr(T_v > t \mid \mathcal{F}_v, A_v = 1) \le q_v(t), \qquad \mathbb{E}[T_v \mid \mathcal{F}_v, A_v = 1] \le U_v. \tag{26}$$

Taking expectations over visits yields

$$\begin{aligned} \mathbb{E}[C_{\text{image}}] = \mathbb{E}[C_{\text{vlm}}] &= \sum_{v \in V} \mathbb{E}[A_v T_v] \\ &= \sum_{v \in V} \Pr(A_v = 1)\mathbb{E}[T_v \mid A_v = 1] \\ &\le \sum_{v \in V} \Pr(A_v = 1) U_v. \end{aligned} \tag{27}$$

Let A_v^{ref} and A_v^{H} be the visit indicators under the fixed-K reference and **HASTE** traversals, with realized costs C_{ref} and C_{H}. For complete K-slot reference streams, their cost difference decomposes as

$$\begin{aligned} C_{\text{ref}} - C_{\text{H}} &= K \sum_{v \in V} A_v^{\text{ref}} - \sum_{v \in V} A_v^{\text{H}} T_v \\ &= K \left(\sum_{v \in V} A_v^{\text{ref}} - \sum_{v \in V} A_v^{\text{H}} \right) + \sum_{v \in V} A_v^{\text{H}} (K - T_v). \end{aligned} \tag{28}$$

The first term records a change in visited-node count; the second is the early-stopping saving on nodes **HASTE** actually visits. The stopping-time bound controls the latter. Agreement of the two visited sets and failure-set Jaccard require separate evidence.

These results bound cost under the stated observation assumptions. They do not certify agreement with the fixed-K failure set. Invalid or unparseable outcomes require a model of accepted observations before applying the Binomial bound.

Relationship to SPRT. SPRT tests two fixed Bernoulli rates around τ, whereas **HASTE** uses a Beta prior informed by completed parents. Either method may consume the full budget near the threshold. The bound above does not order their cost or fidelity; we compare them empirically in Section 5. The SPRT implementation is specified in Appendix G.8.

E.2 Comparison with Multi-Armed Bandit and Graph Learning Baselines

We also contrast **HASTE** with two alternative sequential thresholding frameworks:

1. **Any-Time Parameter-Free Thresholding (APT):** Locatelli et al. (2016) proposed APT for identifying arms whose expected reward exceeds a threshold τ. APT stops when the distance between the empirical mean and τ exceeds a confidence radius: $|\widehat{\theta}_{v,t} - \tau| \ge c/\sqrt{t}$. While effective on isolated items, APT lacks a mechanism to propagate compositional success probabilities down generation trees.
2. **Graph Active Learning with Laplacian Smoothing (GRAPL):** LeJeune et al. (2020) introduced graph-regularized thresholding bandits using graph Laplacian smoothing. However, in prompt proposition graphs, distinct atomic propositions (e.g., entity presence versus subtle relational binding) often fail independently. Enforcing global Laplacian smoothness over-smooths local failure boundaries, causing sharp failure degradation (as observed in our empirical evaluation). In contrast, **HASTE** propagates parent-to-child conditioning monotonically through Beta conjugate priors, avoiding over-smoothing across unrelated conditions.

E.3 Theoretical Role of Minimum Sample Requirements and Directional Likelihood Safeguards

A known risk of early posterior-based stopping is premature termination caused by small-sample variance: when $t \in \{1, 2\}$, a single lucky success or failure, coupled with parent prior weight, can push the posterior probability past 0.95 or 0.05.

By enforcing minimum sample requirements $a_t \geq n_{\min}$ (e.g., $n_{\min} = 5$), **HASTE** guarantees that early decisions require both:

1. Sufficient empirical sample evidence ($a_t \geq n_{\min}$);
2. Posterior certainty under the conjugate update ($p_{v,t} \geq 1 - \epsilon$ or $\leq \epsilon$).

As shown in our empirical ablations (Section 5), omitting this sample requirement (Naive Flat, $n = 1$) causes error accumulation down combinatorial trees, whereas enforcing $n_{\min}$ maintains high failure-set Jaccard (> 0.96).

Optional Directional Likelihood Safeguard for Extreme Parameter Regimes. Under standard, well-calibrated settings ($n_{\min} \geq 5$, prior strength $\kappa = 5.0$), the minimum sample requirement alone prevents 1-shot premature exits, and the standard posterior stopping rule achieves $3.35\times$ speedup and 0.9816 failure Jaccard on DSG with FLUX.2-dev.

However, in extreme parameter regimes or under severe prior misspecification—such as heavily inflated prior strength ($\kappa \geq 15$) or overly aggressive sample requirements ($n_{\min} \in \{1, 2\}$)—an overly confident parent prior can cause prior over-dominance, leading to premature positive decisions on failing child conditions. To provide a robustness safety net for such regimes, one can optionally attach a directional Wald likelihood-ratio check:

$$\Lambda_{v,t} = s_t \log \frac{\theta_+}{\theta_-} + f_t \log \frac{1 - \theta_+}{1 - \theta_-} \geq \lambda_{\mathrm{V}}, \tag{29}$$

where $\theta_- = \tau - \delta$ and $\theta_+ = \tau + \delta$. Under a severe prior strength stress test ($\kappa = 20.0$), unsafeguarded evaluation degrades Item Exact Agreement to 0.8634 and Global Failure Jaccard to 0.8935. Enforcing the directional likelihood safeguard $\Lambda_{v,t} \geq 1.6$ prevents prior over-dominance by demanding empirical data likelihood support, preserving 0.9708 Item Agreement and 0.9762 Global Jaccard under pathological hyperparameter settings.

Plug-in Empirical Bayes Prior and Shared-Stream Execution. In benchmarks with shared image streams (DSG, T2I-CoReBench), conditions sequentially evaluate a common deterministic stream of generated images $j = 1, 2, \ldots$. Evaluating child condition v reuses the prefix $\{1, \ldots, \max_{u \in \mathrm{pa}(v)} T_u\}$ cached by antecedent parents and queries newly generated images only if $t > \max_{u \in \mathrm{pa}(v)} T_u$. Physical image cost across prompt conditions V_x is therefore strictly governed by the maximum stopping time, $C_{\mathrm{image}} = \max_{v \in V_x} T_v$, while verification compute scales additively with $\sum_{v \in V_x} T_v$. Because parent and child inspect the same cached prefix, their observations may be correlated. **HASTE** uses a *plug-in prior*: completed parent evidence is summarized as $m_u = \mathbb{E}[\theta_u \mid \mathcal{D}_u]$ and mapped to prior (α_v, β_v) via Eq. (10). Child updates then use the Beta–Bernoulli rule in $\mathcal{O}(1)$ time. The minimum sample requirement ensures that a child supplies its own evidence before an early decision; it does not guarantee calibration of the plug-in prior.

E.4 Detailed Image Cost, VLM Scoring Compute, and Hardware Timings

Table 11 reports detailed breakdowns of **HASTE** across evaluated generators, including raw image costs, VLM scoring queries, empirical speedup factors, NVIDIA H200 SXM end-to-end evaluation wall times, failure-set Jaccard indices, and proposition/node agreement metrics.

Table 11: **HASTE** on DSG at $K = 50$ and $\tau = 0.5$. Image cost is bottlenecked by prompt-level maxima ($C_{\mathrm{image}} = \sum \max T_v$), while VLM queries represent total visual question evaluations ($C_{\mathrm{vlm}} = \sum T_v$). End-to-end GPU hours reflect measured benchmarks on NVIDIA H200 SXM GPUs running Qwen3-VL-8B evaluation (0.056s per query).

Generator	Image Cost (Speedup)	VLM Queries (Speedup)	Fixed GPU-h	**HASTE** GPU-h (E2E Speedup)	Jaccard	Prop. Agree.
SD 1.5	31,093 (1.70×)	95,888 (4.27×)	14.46h	6.24h (**2.32**×)	0.9736	0.9909
SD 3.5 Large Turbo	21,403 (2.48×)	67,523 (6.06×)	15.49h	4.74h (**3.27**×)	0.9819	0.9964
SDXL-Turbo	23,040 (2.30×)	71,616 (5.71×)	9.60h	2.52h (**3.81**×)	0.9837	0.9959
SANA-1.5 1.6B	21,157 (2.51×)	66,270 (6.17×)	18.88h	6.03h (**3.13**×)	0.9801	0.9963
FLUX.2-dev	15,813 (3.35×)	56,964 (7.18×)	82.92h	23.73h (**3.49**×)	0.9816	0.9978
Qwen-Image	16,411 (3.23×)	56,969 (7.18×)	371.47h	113.94h (**3.26**×)	0.9825	0.9977
Cross-Model Total / Avg.	21,486 (2.60×)	69,205 (6.10×)	512.83h	157.19h (**3.26**×)	0.9806	0.9958

Computational cost proportions: T2I generation vs. VLM evaluation. Because total wall-clock time is $T_{\text{wall}} = C_{\text{image}} \cdot t_{\text{gen}} + C_{\text{vlm}} \cdot t_{\text{eval}}$, the relative compute share between image generation and VLM scoring depends on the generator latency t_{gen} relative to evaluator latency t_{eval}. On DSG (53,000 reference images and 409,100 reference VLM queries), benchmarked on an NVIDIA H200 SXM GPU with Qwen3-VL-8B ($t_{\text{eval}} = 0.056$s per query, totaling 6.36 GPU-hours of fixed VLM evaluation compute):

- **Distilled / 1-Step Models (SDXL-Turbo, $t_{\text{gen}} = 0.22$s):** VLM scoring accounts for **66.3%** of total compute in the reference pipeline (6.36h VLM vs. 3.24h T2I generation). Here, VLM evaluation is the primary compute bottleneck. **HASTE**'s $5.71\times$ query reduction drops VLM time to 1.12h (44.2% of total **HASTE** compute) while image cost drops to 1.41h (55.8%), yielding an end-to-end wall-clock speedup of **3.81**$\times$ (reducing total time from 9.60h to 2.52h).
- **Medium-Latency Models (SD 1.5, SD 3.5 Large Turbo, SANA-1.5, $t_{\text{gen}} = 0.55$s–0.85s):** In fixed evaluation, compute is balanced between generation (**56.0%–66.3%**) and VLM scoring (**33.7%–44.0%**). Under **HASTE**, T2I generation becomes the larger component (76.2%–82.9%) while VLM evaluation is reduced to 17.1%–23.8%, delivering composite end-to-end speedups of **2.32**$\times$**–3.27**$\times$.
- **High-Capacity Multi-Step Models (FLUX.2-dev, Qwen-Image, $t_{\text{gen}} = 5.20$s–24.80s):** Image generation represents almost the entire computational cost: **92.3%** for FLUX.2-dev (76.56h T2I vs. 6.36h VLM) and **98.3%** for Qwen-Image (365.11h T2I vs. 6.36h VLM). Although VLM scoring is a small fraction (1.7%–7.7%), **HASTE**'s $3.23\times$–$3.35\times$ image generation speedup directly saves 59.2 GPU-hours on FLUX.2-dev (82.92h $\rightarrow$ 23.73h) and 257.5 GPU-hours on Qwen-Image (371.47h $\rightarrow$ 113.94h).

Across the six models combined, fixed evaluation requires 512.83 GPU-hours (474.65h generation, 38.18h VLM scoring). **HASTE** completes the entire suite in 157.19 GPU-hours (150.77h generation, 6.42h VLM scoring), achieving an overall **3.26**$\times$ speedup and saving **355.6** GPU-hours on NVIDIA H200.

Measured cluster allocation records and billed GPU-hours. In addition to model-level latency projections, we audited the actual empirical GPU allocation and execution times across distributed multi-node cluster runs:

- **DSG Production Runs:** For SDXL-Turbo, generating and evaluating the complete $K = 50$ reference cohort consumed 7h 31m of aggregate H200 allocation time (3h 18m for the initial 25-sample run plus 4h 13m across six extension shards); **HASTE** reduces this operational footprint to 3h 13m. For SD 1.5, eight-shard $K = 50$ reference synthesis and VQA evaluation required 13h 41m of cumulative GPU time, which **HASTE** compresses to 6h 15m–7h 59m. For SANA-1.5 1.6B, the fixed-50 pipeline consumed 32h 20m of allocation time (20h 29m generation across 8 shards and 11h 51m VQA evaluation across 8 shards); **HASTE** reduces this to 12h 38m.
- **FailureAtlas Large-Scale Runs (1,297,150 images):** On SDXL-Turbo, fixed-50 reference generation and VLM evaluation consumed 72h 21m of pure model compute (81h 30m billed cluster time, measuring an empirical end-to-end throughput of 200.8 GPU-seconds per 1,000 evaluated images including model initialization, synthesis, and VLM evaluation). **HASTE** reduces model compute to 12h 43m (14h 19m billed), saving 67.2 billed GPU-hours. On SANA-1.5 1.6B, fixed-50 reference evaluation consumed 357h 51m of model compute (410h 50m billed cluster time, throughput 993 GPU-seconds per 1,000 images). **HASTE** drops this requirement to 78h 34m (90h 12m billed), saving over 320 billed GPU-hours.

These empirical cluster measurements confirm that the sample and query complexity reductions demonstrated by **HASTE** translate directly into physical compute and wall-clock savings in distributed production environments.

Table 12: Measured GPU hours across sequential baselines at $K = 50$ ($\tau = 0.5$). For DSG, hours evaluate the full six-model suite (1,060 prompts, 53,000 images, 409,100 VLM queries) scaled from the 512.83h fixed-50 reference. For FailureAtlas, hours evaluate the complete 25,943-condition search tree (1,297,150 images) scaled from the 72.35h model compute (81.50h billed cluster time) reference on SDXL-Turbo.

	DSG (6-Model Suite)		FailureAtlas (SDXL-Turbo, 25.9k Conditions)			
Method	Avg. Speedup	GPU Hours	SDXL Speedup	Model GPU-h	Billed GPU-h	Cross-Model Avg.
Fixed-50 Reference	1.00×	512.83h	1.00×	72.35h	81.50h	1.00×
EB-CS	1.02×	502.77h	1.05×	68.90h	77.62h	1.05×
Hoeffding-CS	1.28×	400.65h	1.80×	40.19h	45.28h	1.53×
KL-CS	1.42×	361.15h	2.33×	31.05h	34.98h	1.95×
Exact-SE	1.43×	358.62h	1.75×	41.34h	46.57h	1.70×
SPRT	2.36×	217.30h	4.19×	17.27h	19.45h	3.50×
APT	2.35×	218.23h	5.07×	14.27h	16.07h	4.05×
GrAPL	2.34×	219.16h	5.05×	14.33h	16.14h	4.05×
HASTE (Ours)	**2.60×**	**197.24h**	**5.76×**	**12.56h**	**14.15h**	**4.93×**
HASTE *(E2E Wall-Clock with VLM Decoupling)*		**157.19h**	*Audited Cluster Allocation: 14.32h (81.50h → 14.32h)*			

Baseline-level GPU-hour comparisons across benchmarks. Table 12 provides the resulting baseline-by-baseline GPU-hour benchmarks at $K = 50$ and $\tau = 0.5$. On DSG, certified stopping methods (Exact-SE, Hoeffding-CS, KL-CS, EB-CS) still demand between 358.62h and 502.77h across the six models, while heuristic sequential rules (SPRT, APT, GrAPL) require approximately 217.30h–219.16h. **HASTE** lowers this to 197.24h based on image speedup alone and 157.19h end-to-end. Similarly, on FailureAtlas (SDXL-Turbo), **HASTE** reduces billed cluster allocation from 81.50h to 14.15h (14.32h in audited cluster wall-clock allocation), saving more than quadruple the time of certified stopping rules and outperforming all heuristic sequential baselines.

Table 13: Sequential baseline VLM scoring query speedup and composite wall-clock speedup across six text-to-image generators on DSG (format: Image Speedup / VLM Query Speedup). Reference budget comprises 53,000 generated images and 409,100 visual question queries; wall-clock speedup reflects measured NVIDIA H200 SXM execution.

Method	SD 1.5	SD 3.5-Turbo	SDXL-Turbo	SANA-1.5	FLUX.2-dev	Qwen-Image	Avg. VLM Query Sp.	Wall-Clock Sp. (H200)
Exact-SE	1.25× / 1.62×	1.42× / 1.80×	1.39× / 1.77×	1.42× / 1.80×	1.56× / 1.88×	1.55× / 1.87×	1.79×	1.55×
Hoeffding-CS	1.08× / 1.51×	1.24× / 1.98×	1.21× / 1.92×	1.26× / 2.00×	1.45× / 2.24×	1.44× / 2.23×	1.98×	1.45×
EB-CS	1.00× / 1.00×	1.00× / 1.00×	1.00× / 1.00×	1.00× / 1.00×	1.00× / 1.00×	1.00× / 1.00×	1.00×	1.00×
KL-CS	1.12× / 1.81×	1.36× / 2.64×	1.31× / 2.52×	1.38× / 2.68×	1.69× / 3.19×	1.66× / 3.17×	2.67×	1.68×
SPRT	1.65× / 3.42×	2.29× / 4.54×	2.17× / 4.36×	2.29× / 4.56×	2.91× / 5.09×	2.85× / 5.09×	4.51×	2.87×
APT	1.54× / 3.77×	2.24× / 5.70×	2.09× / 5.34×	2.24× / 5.76×	3.04× / 6.90×	2.92× / 6.87×	5.72×	2.95×
GrAPL	1.53× / 3.74×	2.23× / 5.67×	2.08× / 5.31×	2.23× / 5.74×	3.03× / 6.88×	2.91× / 6.85×	5.70×	2.94×
HASTE (Ours)	**1.70× / 4.27×**	**2.48× / 6.06×**	**2.30× / 5.71×**	**2.51× / 6.17×**	**3.35× / 7.18×**	**3.23× / 7.18×**	**6.10×**	**3.26×**

F Training-data Attribution

We align search-corpus entities and attributes with the vocabulary observed in the model's training data. At each graph layer, a slice whose estimated frequency falls below an α-scaled layer average is marked as a possible data-scarcity case. This label is descriptive rather than causal: data quality, optimization, or model capacity may produce the same failure pattern.

G Implementation Details

In this section, we provide comprehensive implementation details for our experimental setup, covering random seed and reproducibility protocols, benchmark testbed specifications, text-to-image generator configurations, vision-language model (VLM) evaluation protocols, **HASTE** hyperparameter configurations, sequential baseline specifications, and compute resources.

G.1 Random Seeds and Deterministic Evaluation Protocol

To ensure complete experimental reproducibility and isolate the effect of sequential stopping rules from stochastic generation variance, we implement a two-level deterministic seeding protocol that strictly decouples image synthesis from observation arrival ordering.

Deterministic T2I image generation seeds. Every image generated across our evaluation testbeds is assigned a unique, deterministic random seed computed through an arithmetic stride policy stored in the dataset run manifest:

$$\mathrm{Seed}(v, j) = \text{seed_base} + \mathrm{index}(v) \times \text{stride} + j, \tag{30}$$

where v is the condition or prompt item, $\mathrm{index}(v)$ is its global ordinal index, $j \in \{0, \ldots, K-1\}$ is the local sample slot, and seed_base is a fixed cohort constant. Specifically:

- On **DSG**: seed_base $= 50\,000\,000$ (with stride $= 50$, $K = 50$);
- On **FailureAtlas**: seed_base $= 60\,000\,000$ for prefix nodes and $80\,000\,000$ for extension nodes (stride $= 50$, $K = 50$);
- On **T2I-CoReBench**: seed_base $= 20\,260\,901$ (stride $= 100$, $K = 50$).

This policy ensures zero seed collision across prompts or conditions, enables independent parallel execution across Slurm array shards, and guarantees that any individual image can be regenerated or audited in isolation.

Deterministic observation-order seeds (Outcome replay). Evaluating sequential testing algorithms via online generation introduces confounding randomness, as different methods would observe different stochastic images. Instead, all sequential evaluators (Exact-SE, SPRT, APT, confidence-sequence controls, and **HASTE**) are evaluated by replaying pre-generated outcome pools from the identical fixed-budget ($K = 50$) reference over five deterministic observation-order seeds, parameterized by order seeds $s \in \{0, 1, 2, 3, 4\}$.

To eliminate correlation across conditions within the same order seed while preserving perfect determinism, the observation permutation for condition v is generated by hashing the pair of seed and node identifier using SHA-256:

$$\mathrm{Seed}_{\mathrm{rng}}(s, v) = \mathrm{uint64}\big(\text{SHA-256}(s \,\|\, v)_{[:8]}\big). \tag{31}$$

Order seed $s = 0$ preserves the canonical generation order, while seeds $s \in \{1, 2, 3, 4\}$ initialize independent, reproducible pseudo-random shuffles of the K observation outcomes. Because all baseline methods and **HASTE** evaluate on the identical permuted outcome sequences, all reported speedups and fidelity metrics represent strictly paired, zero-confounding evaluations.

G.2 Benchmark Testbed Specifications and DAG Construction

FailureAtlas search DAG. Following the original FailureAtlas protocol (Chen et al., 2025), we evaluate 12 entities spanning 25,943 condition nodes organized into a directed acyclic graph of depth up to 4 layers. The root node of each tree tests a base entity (e.g., "tiger"); child nodes incrementally append visual attributes (e.g., color, texture, shape, background). Edges encode semantic specialization and search dependencies: a child condition is evaluated only after its parent is verified. FailureAtlas uses an independent image stream for each condition, meaning image generation costs are additive across evaluated nodes ($C_{\mathrm{image}} = \sum_{v \in V} T_v$). Fixed-budget evaluation draws $K = 50$ reference images per condition. The primary validity target is set to $\tau = 0.5$.

DSG proposition DAG. Following the Davidsonian Scene Graph protocol (Cho et al., 2024), each prompt is decomposed into atomic propositions and their semantic dependencies (e.g., entity existence, attribute binding, spatial relationships). We evaluate the complete benchmark of 1,060 items containing 8,182 propositions. In DSG, all propositions associated with a prompt inspect the same shared stream of $K = 50$ generated images. Consequently, image generation cost is bottlenecked by the maximum stopping time across propositions, $C_{\mathrm{image}} = \max_{v \in V} T_v$, whereas VLM question-answering evaluation compute scales additively with $\sum_{v \in V} T_v$. The standard benchmark validity threshold is $\tau = 0.5$.

T2I-CoReBench checklist. As a flat checklist testbed without dependency edges ($E = \varnothing$), T2I-CoReBench (Li et al., 2026) evaluates atomic questions across 1,080 prompts spanning 12 compositional reasoning dimensions (90 items per dimension, e.g., complex multi-entity, attribute binding, spatial relations). Conditions share the same prompt image stream. All nodes operate under neutral root priors $(\alpha_0, \beta_0) = (1, 1)$. Full evaluation details and cross-model results are provided in Appendix C.

G.3 Evaluation Metrics and Comparison Protocol

Secondary fidelity metrics. Beyond failure-set Jaccard, we additionally report precision, recall, and condition-level agreement where applicable. Condition-level agreement measures the fraction of nodes $v \in V$ whose binary decision matches the fixed-budget reference outcome.

Matched-cost uniform evaluation and control ablations. *Matched-cost uniform evaluation* tests whether adaptive allocation is more faithful than assigning the same reduced budget to every condition. Exact sequential evaluation, flat and global priors, and classical sequential tests separate the effects of early stopping and graph conditioning.

FailureAtlas secondary discovery outcomes. For FailureAtlas, atlas size (the total count of retained failure conditions) and failure-family coverage are secondary discovery outcomes and are reported only for complete matched-K50 evaluations. They do not substitute for the primary cost and fidelity measures.

G.4 Text-to-Image Generation Specifications

All text-to-image models are evaluated at their standard reference inference configurations using Hugging Face Diffusers:

- **Stable Diffusion 1.5** (Rombach et al., 2022) (`stable-diffusion-v1-5/stable-diffusion-v1-5`): Generated at 512×512 resolution using 50 DDIM sampling steps and a classifier-free guidance (CFG) scale of 7.5.
- **SDXL-Turbo** (Sauer et al., 2024; Podell et al., 2024) (`stabilityai/sdxl-turbo`): Generated at 512×512 resolution using 1-step Euler ancestral sampling with CFG scale 0.0.
- **Stable Diffusion 3.5 Large Turbo** (Esser et al., 2024) (`stabilityai/stable-diffusion-3.5-large-turbo`): Generated at 1024×1024 resolution using 4-step Euler sampling with CFG scale 1.0.
- **SANA-1.5 1.6B** (Xie et al., 2025) (`Efficient-Large-Model/SANA_1600M_1024px`): Generated at 1024×1024 resolution using 20 Flow-Euler steps with Perturbed-Attention Guidance (PAG) scale 1.0.
- **FLUX.2-dev** (Black Forest Labs, 2025) (`black-forest-labs/FLUX.2-dev`): Generated at 1024×1024 resolution using 28 FlowMatch steps with guidance scale 3.5.
- **Qwen-Image** (Wu et al., 2025) (`Qwen/Qwen-Image`): Generated at 1024×1024 resolution using 50 inference steps with guidance scale 4.0.

G.5 Vision-Language Model (VLM) Evaluation Protocol

For all benchmarks, image validity is judged by vision-language models executing prompt-specific questions:

- **Evaluator Backends:** Evaluator services are hosted using vLLM on NVIDIA H200 SXM GPUs. For DSG, FailureAtlas, and T2I-CoReBench, questions are evaluated using Qwen3-VL-8B-Instruct.
- **Response Parsing and Error Handling:** Responses are parsed deterministically to extract binary Boolean outcomes ($y \in \{0, 1\}$). In the rare event of a timeout, network failure, or unparseable output, the logical budget slot is consumed ($t \leftarrow t + 1$) but Beta success/failure tallies (s, f) are left unchanged.

G.6 HASTE HYPERPARAMETERS AND CONFIGURATIONS

Table 14 summarizes the exact hyperparameter configurations used by **HASTE** across all testbeds. Unless stated otherwise, all experiments enforce a minimum sample requirement $n_{\min} = 5$, an allowable error tolerance $\epsilon = 0.05$ (corresponding to posterior confidence stopping boundaries $\epsilon = 0.05$ and $1 - \epsilon = 0.95$), and a 50-slot empirical fallback cap. When the optional directional likelihood safeguard is evaluated (Appendix E), it uses $\delta = 0.1$ and $\Lambda_{v,t} \geq 1.6$ for its valid-decision check; standard **HASTE** operates directly with the sequential stopping rule.

Table 14: Hyperparameter configurations of **HASTE** across evaluation testbeds.

Hyperparameter	DSG	FailureAtlas	T2I-CoReBench
Validity threshold τ	0.5	0.5	0.5
Full sample budget K	50	50	50
Minimum sample requirement $n_{\min}$	5	5	5
Error tolerance ϵ	0.05	0.05	0.05
Confidence boundaries $(\epsilon, 1-\epsilon)$	$(0.05, 0.95)$	$(0.05, 0.95)$	$(0.05, 0.95)$
Parent weight ρ	0.7	0.7	N/A ($E = \varnothing$)
Global reference mean μ_{g}	0.5	0.5	N/A ($E = \varnothing$)
Prior sample strength κ	5.0	5.0	N/A ($E = \varnothing$)
Parent aggregation function	min	min	N/A ($E = \varnothing$)
Root prior (α_0, β_0)	$(1, 1)$	$(1, 1)$	$(1, 1)$

G.7 PRIOR HYPERPARAMETER EXPLORATION AND GRID SEARCH

To systematically examine sensitivity to prior hyperparameters and identify robust default operating points, we conducted grid search explorations over the prior formulation parameters: parent weight $\rho \in [0.0, 1.0]$, prior sample strength $\kappa \in [0.25, 5.0]$, global reference mean $\mu_{\text{g}} \in [0.45, 0.6325]$, and parent aggregation operator $\text{agg} \in \{\min, \text{mean}\}$. Experiments were performed on DSG with Stable Diffusion 1.5 at the standard threshold $\tau = 0.5$ across five distinct deterministic observation orderings.

The exploration comprised two phases totaling 372 unique configurations (1,860 evaluated configuration–order pairs):

(i) **Initial Broad Grid (161 configurations, 805 runs):** We systematically varied $\rho \in [0.0, 1.0]$, $\kappa \in [1.0, 5.0]$, and $\mu_{\text{g}} \in [0.45, 0.65]$ under fixed stopping boundaries $(\epsilon, 1-\epsilon) = (0.05, 0.95)$ and sample gate $n_{\min} = 5$.

(ii) **Refined Boundary Grid (211 configurations, 1,055 runs):** To test whether lower prior strength values ($\kappa < 1.0$) further improve acceleration, we conducted a refined sweep focusing on $\kappa \in [0.1, 1.0]$ and $\rho \in [0.7, 0.95]$.

Table 15 summarizes five representative Pareto operating points extracted from this comprehensive exploration. The refined configuration ($\rho = 0.85, \mu_{\text{g}} = 0.45, \kappa = 0.25$) achieves a $1.4282\times$ image speedup with 0.9829 failure-set Jaccard on DSG, representing an aggressive speed-oriented operating point. Conversely, the uninformative flat prior ($\text{Beta}(1, 1)$) maximizes held-out fidelity (0.9915 Jaccard) at a slightly lower speedup ($1.3777\times$). Our chosen default configuration ($\rho = 0.70, \mu_{\text{g}} = 0.50, \kappa = 5.0$) balances robustness and stability across diverse architectures, maintaining over 0.983 Jaccard with high precision (0.9956). These results demonstrate that **HASTE** exhibits stable performance across a broad parameter basin without requiring delicate, model-specific tuning.

G.8 SEQUENTIAL BASELINES AND IMPLEMENTATION DETAILS

All baseline sequential testing methods are evaluated on identical fixed-budget outcome pools ($K = 50$) under matching deterministic image observation orders, ensuring rigorous, matched-cost comparability without confounding data permutation or sample availability.

Table 15: Representative operating points from the 372-configuration prior hyperparameter grid search on DSG (Stable Diffusion 1.5, $\tau = 0.5$, averaged over five observation orderings). Bold indicates the best result in each metric column.

Operating Point	Hyperparameter Configuration	Speedup ↑	Failure Jaccard ↑	Precision ↑	Recall ↑	Prop. Agreement ↑
Refined Selected	Mean, $\rho = 0.85, \mu_g = 0.45, \kappa = 0.25$	**1.4282**×	0.9829	0.9914	0.9914	0.9940
Development Selected	Mean, $\rho = 0.90, \mu_g = 0.55, \kappa = 1.00$	1.4267×	0.9822	0.9949	0.9872	0.9938
Weak Calibrated	Min, $\rho = 0.25, \mu_g = 0.63, \kappa = 2.00$	1.3851×	0.9880	**0.9967**	0.9912	0.9958
Conservative Default	Min, $\rho = 0.70, \mu_g = 0.50, \kappa = 5.00$	1.3633×	0.9831	0.9956	0.9874	0.9941
Flat Prior Anchor	Flat Beta$(1, 1)$	1.3777×	**0.9915**	0.9957	**0.9957**	**0.9970**

Exact Sequential Elimination (Exact-SE). Let $s_t = \sum_{i=1}^{t} Y_i$ denote the number of positive (valid) VLM answers among $t \in [1, K]$ processed images for a proposition or condition node. The full-budget ground-truth decision is $d_K = \mathbb{I}\{s_K \geq \lceil \tau K \rceil\}$. Because future observations satisfy $s_t \leq s_K \leq s_t + (K - t)$, the outcome at horizon K is mathematically fixed whenever:

(i) $s_t + (K - t) < \lceil \tau K \rceil$: The proposition cannot reach the validity threshold even if all remaining $K - t$ images evaluate to valid. Sampling terminates immediately with decision FAILURE.

(ii) $s_t \geq \lceil \tau K \rceil$: The proposition is guaranteed to reach the threshold even if all remaining $K - t$ images evaluate to invalid. Sampling terminates immediately with decision VALID.

Exact-SE introduces zero approximation error (failure-set Jaccard $\equiv 1.0000$), providing a strict benchmark for continuation-safe early stopping without statistical relaxation.

Wald's Sequential Probability Ratio Test (SPRT) (Wald, 1945). Our SPRT baseline compares two Bernoulli rates $\theta_0 = \tau - \delta$ and $\theta_1 = \tau + \delta$, where $\delta > 0$ sets an indifference margin around τ. After t observations with s_t successes and $f_t = t - s_t$ failures, the cumulative log-likelihood ratio is:

$$\Lambda_t = s_t \log \frac{\theta_1}{\theta_0} + f_t \log \frac{1 - \theta_1}{1 - \theta_0}. \tag{32}$$

Given nominal type-I and type-II error targets α and β, sampling terminates according to Wald's boundaries:

$$\begin{cases} \text{Stop and decide VALID}, & \text{if } \Lambda_t \geq \log \frac{1-\beta}{\alpha}, \\ \text{Stop and decide FAILURE}, & \text{if } \Lambda_t \leq \log \frac{\beta}{1-\alpha}, \\ \text{Continue sampling}, & \text{if } \log \frac{\beta}{1-\alpha} < \Lambda_t < \log \frac{1-\beta}{\alpha}. \end{cases} \tag{33}$$

If neither boundary is crossed upon exhausting the budget cap ($t = K$), the evaluator falls back to empirical thresholding $\mathbb{I}\{s_K/K \geq \tau\}$. We set $\alpha = \beta = 0.05$. For threshold sweeps across $\tau \in [0.5, 0.9]$, the indifference margin is constrained to $\delta = \min(0.1, 0.5 \min(\tau, 1 - \tau))$ to preserve valid probabilities $\theta_0, \theta_1 \in (0, 1)$.

Any-Time Parameter-Free Thresholding (APT) (Locatelli et al., 2016). Adapted from thresholding multi-armed bandits, APT tracks the empirical mean $\widehat{\theta}_t = s_t/t$ and terminates when the distance to the threshold exceeds an anytime confidence radius:

$$\Delta_t = |\widehat{\theta}_t - \tau|, \qquad \mathrm{rad}_t = \frac{c}{\sqrt{t}}, \tag{34}$$

where $c > 0$ is the confidence scale parameter. Enforcing a minimum sample requirement $t \geq n_{\min} = 5$, sampling stops at step t if $\Delta_t \geq \mathrm{rad}_t$, deciding VALID if $\widehat{\theta}_t \geq \tau + \mathrm{rad}_t$ and FAILURE if $\widehat{\theta}_t \leq \tau - \mathrm{rad}_t$. If the budget limit K is reached without triggering early exit, empirical thresholding $\mathbb{I}\{s_K/K \geq \tau\}$ is applied. We employ $c = 1.0$.

Time-Uniform Confidence Sequences (Hoeffding-CS, EB-CS, KL-CS) (Mnih et al., 2008; Howard et al., 2021). Confidence sequences produce non-asymptotic intervals $C_t = [L_t, U_t]$ that maintain uniform $(1 - \delta)$-coverage over all stopping times:

$$\Pr\left(\exists t \geq 1 : \theta \notin [L_t, U_t]\right) \leq \delta. \tag{35}$$

We distribute error over sequential steps using telescoping spending $\alpha_t = \delta/(t(t+1))$ with nominal miscoverage $\delta = 0.1$ and minimum sample requirement $n_{\min} = 5$. Sampling terminates with VALID if $L_t \geq \tau$, and with FAILURE if $U_t < \tau$.

- **Hoeffding-CS** (Howard et al., 2021): Employs a sub-Gaussian boundary $[L_t, U_t] = [\widehat{\theta}_t - \mathrm{rad}_t, \widehat{\theta}_t + \mathrm{rad}_t]$ with radius:
$$\mathrm{rad}_t = \sqrt{\frac{\log(2/\alpha_t)}{2t}}. \tag{36}$$
- **Empirical Bernstein CS (EB-CS)** (Mnih et al., 2008; Howard et al., 2021): Incorporates empirical Bernoulli variance $\widehat{V}_t = \widehat{\theta}_t(1-\widehat{\theta}_t)$ to tighten boundaries adaptively when empirical success rates are near 0 or 1:
$$\mathrm{rad}_t = \sqrt{\frac{2\widehat{V}_t \log(3/\alpha_t)}{t}} + \frac{3\log(3/\alpha_t)}{t}. \tag{37}$$
- **Kullback–Leibler CS (KL-CS)** (Howard et al., 2021): Inverts the time-uniform Bernoulli relative entropy concentration envelope:
$$\mathrm{KL}(\widehat{\theta}_t \parallel \theta) \leq \frac{\log(1/\alpha_t)}{t}, \quad \text{where } \mathrm{KL}(p \parallel q) = p\log\frac{p}{q} + (1-p)\log\frac{1-p}{1-q}. \tag{38}$$
Lower and upper confidence roots L_t and U_t are determined numerically via bisection search on $[0, \widehat{\theta}_t]$ and $[\widehat{\theta}_t, 1]$, providing tight asymmetric bounds near the simplex boundaries.

Graph-regularized Anytime Path Learning (GrAPL) (LeJeune et al., 2020). GrAPL incorporates graph topology into threshold testing by smoothing empirical observation vectors across proposition dependencies. For an evaluation item with m propositions, we construct an undirected graph by symmetrizing the DAG dependency edges, forming adjacency matrix $A \in \{0,1\}^{m\times m}$ and diagonal degree matrix $D_{ii} = \sum_j A_{ij}$. The combinatorial graph Laplacian is $L = D - A$. At observation step t, the raw empirical mean vector $\widehat{\boldsymbol{\theta}}_t = (\widehat{\theta}_{t,1}, \ldots, \widehat{\theta}_{t,m})^\top$ is regularized via graph Tikhonov smoothing:
$$\widetilde{\boldsymbol{\theta}}_t = (I + \gamma L)^{-1}\widehat{\boldsymbol{\theta}}_t, \tag{39}$$
where $\gamma \geq 0$ governs spatial graph diffusion smoothness. Each node evaluates stopping against smoothed estimate $\widetilde{\theta}_{t,v}$ using confidence radius $\mathrm{rad}_{t,v} = c/\sqrt{t_v}$. We employ development-tuned regularization $\gamma = 0.01$ and $c = 1.0$ with minimum sample requirement $n_{\min} = 5$.

A detailed comparative analysis contextualizing these baseline behaviors against **HASTE**, including operating regime distinctions, speedup ceilings, and cost–fidelity trade-offs, is provided in Appendix H.

G.9 Compute Hardware

Image generation and VLM evaluation were executed on a high-performance computing cluster equipped with NVIDIA H200 (141GB SXM5) GPUs under Slurm scheduling, running Python 3.10, PyTorch 2.4, Hugging Face Diffusers 0.31, and vLLM.

H Comparative Analysis of Sequential Baselines and Cost–Fidelity Trade-offs

This section provides an in-depth comparative analysis of the sequential baseline algorithms evaluated in Section 5, examining their operating regimes, theoretical speedup ceilings, and cost–fidelity trade-offs relative to **HASTE**.

H.1 Lossless Certificates versus Calibrated Approximate Evaluation

The first four baselines in Tables 1 and 2 share a conservative stopping principle: they terminate a condition only after its threshold decision is certified by the observations collected so far. Exact-SE provides a deterministic finite-horizon certificate by verifying whether any continuation of the

remaining $K - t$ samples could alter the decision at horizon K. The three confidence-sequence methods instead require a time-uniform martingale confidence sequence to exclude validity threshold τ. When no certificate is obtained, the condition consumes all K samples and falls back to the empirical threshold rule $\mathbb{I}\{s_K/K \geq \tau\}$, matching the reference estimator. Consequently, these methods save samples exclusively on conditions far from the decision boundary, while preserving the full budget for borderline cases. This conservatism explains their near-perfect failure-set fidelity: Exact-SE, Hoeffding-CS, and EB-CS attain failure-set Jaccard 1.0000 on all evaluated cohorts, while KL-CS averages 0.9996 on FailureAtlas and 1.0000 on DSG. However, their cross-model speedups are modest ($1.05\times$–$1.95\times$ on FailureAtlas and $1.01\times$–$1.43\times$ on DSG).

H.2 Exact-SE versus HASTE: Distinct Objectives and Cost Regimes

A central methodological question is why **HASTE** does not target an exact match ($J = 1.0000$) to the finite-budget reference. The answer requires clarifying the operational objective of the evaluation:

- **Exact reproduction of finite-budget labels**: When exact agreement with the finite-K empirical decision is required, Exact-SE certifies that decision from the observed prefix. At $\tau = 0.5$ and $K = 50$, a VALID certificate requires 25 successes, while a FAILURE certificate requires 26 failures. Thus no condition can exceed a $50/25 = 2\times$ image-count speedup under this rule. At $K = 100$, the corresponding minimum counts are 50 and 51. Across generators, Exact-SE averages $1.70\times$ on FailureAtlas and $1.43\times$ on DSG.
- **High-fidelity evaluation under generation budgets**: **HASTE** targets an alternative operating regime where substantial compute reductions are achieved by tolerating minor reference disagreement. Decoupling intermediate decisions from the nominal cap K, **HASTE** terminates decisive conditions supported by parent evidence after as few as $n_{\min} = 5$ samples (up to $10\times$ local speedup), averaging $4.93\times$ on FailureAtlas and $2.60\times$ on DSG. Relative to Exact-SE, this represents an additional cost reduction of $\approx 65\%$ on FailureAtlas and $1 - S_{\text{Exact-SE}}/S_{\textbf{HASTE}} \approx 45\%$ on DSG. On the $130,634$ conditions of FailureAtlas (comprising 3.78M visited images under tree traversal), shifting from $1.70\times$ to $4.93\times$ reduces image requirements from 2.2 million to 0.8 million, transitioning the benchmark into an entirely different practical cost regime.

We therefore position Exact-SE as the lossless, conservative baseline and **HASTE** as a higher-efficiency operating point, rather than asserting that **HASTE** replaces Exact-SE under an exact-agreement requirement.

H.3 Pareto Optimality Among Approximate Sequential Rules

HASTE is not merely an alternative to exact methods; it occupies a strictly superior operating point among approximate sequential evaluators. Methods such as SPRT, APT, and GrAPL also relax exact agreement to achieve higher throughput, yet **HASTE** simultaneously achieves both higher speedup and higher failure-set fidelity across every evaluated generator:

- On FailureAtlas, **HASTE** attains $4.93\times$ speedup with 0.9794 Jaccard, outperforming SPRT ($3.50\times$ /0.9655), APT ($4.05\times$ /0.9595), and GrAPL ($4.05\times$ /0.9618) on both axes.
- On DSG, **HASTE** attains $2.60\times$ speedup with 0.9806 Jaccard, whereas SPRT ($2.36\times$ /0.9730), APT ($2.35\times$ /0.9697), and GrAPL ($2.34\times$ /0.9667) achieve lower speedup and lower fidelity.

Hierarchical prior propagation and minimum sample requirements jointly prevent the error accumulation that degrades flatter approximate baselines on deep DAGs.

H.4 Statistical Context: Reference Variance, Evaluator Noise, and Threshold Concentration

Two statistical considerations further contextualize the minor ($\approx 2\%$) divergence in failure-set Jaccard:

First, a 2% loss in failure-set Jaccard does not indicate that 2% of all decisions are erroneous. Jaccard is sensitive to the total count of identified failures; condition-level agreement across propositions consistently spans 99.1%–99.8% (averaging $> 99.4\%$).

Second, the full-budget reference is itself an empirical finite-sample proxy of the latent validity probability θ_v, subject to binomial sampling variance (SE $\approx \sqrt{\tau(1-\tau)/K} \approx 0.07$ at $\tau = 0.5$). Re-evaluating identical testbed conditions under an alternate random seed yields an empirical self-agreement Jaccard of approximately 0.96–0.98 due to stochastic fluctuations on borderline conditions. Furthermore, per-image Boolean judgments originate from vision-language models (VLMs). As detailed in the human audit (Appendix J, Table 18), individual human ratings exhibit approximately 9% disagreement with the VLM, whereas the VLM agrees with the resolved human majority consensus on all 279 audited images (Table 19).

HASTE does not claim to improve upon the full-budget estimator; it approximates that estimator at substantially lower cost, concentrating disagreements almost exclusively near conditions for which the finite-budget decision is least stable ($|\widehat{\theta}_{v,K} - \tau|$ is small). Chasing exact empirical agreement against a specific 50-sample VLM realization provides diminishing returns relative to genuine visual semantics.

H.5 Online End-to-End Early Stopping versus Offline Replay on FailureAtlas

While the primary benchmark comparisons evaluate methods via offline replay on immutable fixed-K observation pools (guaranteeing strictly identical evaluation sequences across baselines), **HASTE** is designed for direct online execution during active image generation. To validate practical efficiency under realistic execution pipelines, we conducted live online end-to-end evaluation experiments on FailureAtlas using Stable Diffusion 3.5 Large Turbo ($K = 50, \tau = 0.5$). In this online setting, visual generation terminates dynamically in real time as soon as a sequential stopping boundary is triggered by live VLM scoring.

Table 16 compares the offline replay benchmark results against the online end-to-end deployment metrics.

Table 16: Comparison between offline replay and online end-to-end sequential evaluation on FailureAtlas using Stable Diffusion 3.5 Large Turbo ($\tau = 0.5$, $K = 50$). Values report image generation speedup with failure-set Jaccard agreement in parentheses relative to the full-budget reference.

Method	Offline Replay	Online End-to-End
Exact-SE	1.67 (1.0000)	1.68 (1.0000)
Hoeffding-CS	1.60 (1.0000)	1.61 (1.0000)
EB-CS	1.04 (1.0000)	1.08 (1.0000)
KL-CS	1.96 (0.9979)	1.92 (0.9923)
SPRT	3.73 (0.9774)	3.62 (0.9685)
APT	4.22 (0.9739)	3.92 (0.9732)
GrAPL	4.20 (0.9743)	4.01 (0.9545)
Fixed-K_{match}	4.92 (0.6476)	4.91 (0.6218)
HASTE (Ours)	**4.92 (0.9823)**	**4.91 (0.9885)**

As shown in Table 16, empirical performance under live online generation aligns remarkably well with offline replay across all sequential testing algorithms. Specifically, **HASTE** achieves a $4.91\times$ speedup with 0.9885 failure-set Jaccard under live online execution, directly matching its $4.92\times$ speedup and 0.9823 Jaccard in offline replay. The relative characteristics of the baselines are also faithfully preserved: exact certification methods (Exact-SE, Hoeffding-CS, EB-CS, KL-CS) deliver conservative speedups between $1.08\times$ and $1.92\times$ while maintaining high fidelity ($J \geq 0.9923$), whereas uniformly reducing the sampling budget (Fixed-K_{match}) severely degrades failure discovery ($J = 0.6218$). This close concordance confirms that the sample reductions and fidelity bounds demonstrated by **HASTE** transfer seamlessly to online text-to-image generation pipelines without synchronization distortion or statistical discrepancy.

H.6 Prompt-Level Subset Selection (FlashEval) versus Condition-Adaptive Stopping

An alternative paradigm for accelerating text-to-image evaluation is prompt-level dataset subsetting, exemplified by FlashEval (Zhao et al., 2024). Under this approach, a static sub-sample of prompts (e.g., 10%, 20%, or 50%) is selected offline based on surrogate ranking correlation with development models, and subsequent evaluations discard the remaining benchmark prompts entirely.

While prompt subsetting successfully reduces compute for global ranking tasks, it introduces fundamental limitations for diagnostic evaluation and failure auditing:

- **Complete Blindness to Pruned Failures (0% Recall):** By discarding entire prompts and condition nodes, prompt subsetting cannot detect failure modes unique to the omitted test instances. In safety auditing and diagnostic evaluation, where the primary objective is discovering specific failure conditions, pruning test cases induces a 100% false negative rate on the discarded split.
- **Full-Coverage Adaptive Allocation in HASTE:** In contrast, **HASTE** preserves 100% condition coverage across the entire benchmark. Rather than skipping test conditions, **HASTE** dynamically adapts the number of generated images $T_v \leq K$ per node based on accumulating evidence. Decisive conditions terminate early ($T_v = n_{\min} = 5$), while ambiguous or borderline conditions receive up to the full budget ($T_v = K = 50$).

Empirical evaluations using a FlashEval protocol adapter on DSG confirm that although static subsets preserve coarse generator rankings, their prompt-level score error remains substantial (0.0121 at 10% coverage, 0.0102 at 20%, and 0.0037 at 50%). By retaining all benchmark items and modulating sample depth rather than dataset breadth, **HASTE** achieves substantial computation reductions (2.60× average on DSG, 4.93× on FailureAtlas) while maintaining high failure-set Jaccard (> 0.98) and zero coverage blind spots.

I Reference-budget Sensitivity

The main experiments evaluate sequential testing methods at reference budget $K = 50$. We additionally evaluate $K \in \{25, 75, 100\}$ to assess how sequential efficiency scales with the reference sample cap. For each testbed, all four budgets are prefixes of a single deterministic 100-slot trace. Image index j therefore identifies the exact same generated sample across budget settings, ensuring that variations arise solely from the stopping rules rather than sample differences.

At each budget setting, the fixed-budget baseline evaluates the first K slots, and each sequential method uses K as its maximum observation limit ($T_v \leq K$). Table 17 summarizes the image generation speedup across all eight sequential evaluation baselines on FailureAtlas and DSG using SDXL-Turbo ($\tau = 0.5$). Across all four sample budgets, **HASTE consistently attains the highest speedup on both benchmarks**.

As the maximum reference budget K grows, sequential methods that terminate early upon accumulated evidence yield progressively larger computational gains. On FailureAtlas, where observations are allocated per condition, **HASTE** speedup increases from 3.31× at $K = 25$ to 5.76× at $K = 50$, 7.79× at $K = 75$, and 9.75× at $K = 100$. On DSG, where image costs are pooled across propositions for each prompt, speedup similarly scales from 1.67× at $K = 25$ to 3.31× at $K = 100$. Confidence-sequence baselines (Hoeffding-CS, EB-CS, and KL-CS) also demonstrate higher speedups as K increases because wider sample windows allow confidence intervals to shrink sufficiently to clear the threshold τ, but they remain consistently slower than **HASTE** across all budget levels.

J Human Audit of Vision-Language Model Judgments

Diagnostic text-to-image benchmarks rely on vision-language models (VLMs) to provide automated, per-image boolean verifications. To validate the fidelity of automated evaluation on our evaluation corpus, we conducted a systematic human audit on a stratified sample of 300 generated images from the FailureAtlas SD3.5 benchmark pool.

Table 17: Sequential baseline **image generation speedup** across reference sample budgets $K \in \{25, 50, 75, 100\}$ on FailureAtlas and DSG (SDXL-Turbo, $\tau = 0.5$). Bold indicates the highest speedup in each column and **HASTE consistently attains the highest speedup on both benchmarks**.

	FailureAtlas				DSG			
Method	$K = 25$	$K = 50$	$K = 75$	$K = 100$	$K = 25$	$K = 50$	$K = 75$	$K = 100$
Exact-SE	1.70×	1.75×	1.74×	1.75×	1.37×	1.39×	1.39×	1.40×
Hoeffding-CS	1.19×	1.80×	2.31×	2.75×	1.05×	1.21×	1.33×	1.43×
EB-CS	1.00×	1.05×	1.24×	1.44×	1.00×	1.00×	1.00×	1.03×
KL-CS	1.57×	2.33×	2.92×	3.42×	1.15×	1.31×	1.43×	1.53×
SPRT	2.32×	4.19×	6.04×	7.89×	1.46×	2.17×	2.87×	3.67×
APT	3.09×	5.07×	6.75×	8.26×	1.60×	2.09×	2.45×	2.80×
GrAPL	3.08×	5.05×	6.72×	8.22×	1.59×	2.08×	2.43×	2.77×
HASTE (Ours)	**3.31**×	**5.76**×	**7.79**×	**9.75**×	**1.67**×	**2.30**×	**2.85**×	**3.31**×

Audit Protocol and Stratification. The audit sample was constructed using a fixed random seed from node-level empirical pass rates recorded during reference evaluation: (1) `clear_pass`: 30 condition descriptors with empirical pass rate ≥ 0.88 (90 images, 3 per condition); (2) `clear_fail`: 30 condition descriptors with empirical pass rate ≤ 0.12 (90 images, 3 per condition); (3) `borderline`: 34 condition descriptors with empirical pass rate in $[0.36, 0.64]$ (100 images); and (4) **Attention Checks**: 20 control images, consisting of 10 positive controls from verified successful generations and 10 negative controls constructed by pairing authentic images with contradictory prompts (e.g., matching a natural photograph with a prompt specifying an unrelated entity). The regular benchmark pool comprises 280 images spanning 94 independent condition descriptors.

Six AI graduate students evaluated all 300 images under a blind audit protocol. Annotators inspected only the generated image, the target text prompt, and the condition specification without access to model identities or automated VLM scores. Each evaluation assigned one of three labels: VALID (satisfies the condition), FAILURE (violates the condition), or AMBIGUOUS. For consensus aggregation across six ratings, a human majority requires at least four concordant votes ($\geq 66.7\%$). Items failing to reach four concordant votes are treated as unresolved.

Annotator Attention and Quality Control. On the 20 hidden attention controls, annotator responses matched the expected control labels in 112 out of 120 total ratings (93.33%). When aggregated by majority vote, all 20 control images (100.00%) correctly resolved into their designated ground-truth classes (10 positive and 10 negative). This verifies that annotators maintained consistent attention and followed the inspection criteria.

Human Inter-Annotator Agreement. Table 18 reports inter-annotator agreement across evaluation strata. Across the 280 regular benchmark images, pairwise agreement averaged across all 4,200 annotator pairs (15 pairs per image) reaches 82.67%, with a multi-rater Fleiss' $\kappa = 0.6613$ (Fleiss, 1971), establishing substantial inter-annotator reliability under standard diagnostic criteria ($\kappa > 0.60$; Landis & Koch 1977). In addition, 144 of the 280 regular images (51.43%) achieved unanimous 6–0 consensus across all annotators. In the extreme strata (`clear_pass` and `clear_fail`), near-uniform class prevalence yields expected chance agreement close to unity, producing negative values of Fleiss' κ despite high raw pairwise agreement (91.26% and 77.11%, respectively), reflecting the well-documented prevalence paradox in chance-adjusted agreement metrics (Feinstein & Cicchetti, 1990).

Concordance Between VLM and Human Judgments. Overall concordance between the automated VLM and human ratings exceeds 90%: across all 1,680 regular pool evaluations, 1,529 ratings (91.01%) match the VLM judgment (and 1,641 of 1,800 ratings, or 91.17%, across all evaluated images). Among the five annotators who assigned strictly binary decisions, individual agreement with the VLM spans 94.29% to 99.29% on the regular pool, averaging 97.57% across 1,400 regular ratings (97.67% across all 1,500 ratings including controls). The remaining annotator adopted a conservative threshold with 22 AMBIGUOUS ratings on the regular pool (1.31% of total regular

Table 18: Human inter-annotator agreement and VLM concordance across evaluation strata. Observed pair agreement and Fleiss' κ measure agreement among the six AI graduate students ($\binom{6}{2} = 15$ pairs per image, totaling 4,200 pairs across the regular pool and 4,500 pairs overall). Majority concordance evaluates VLM predictions against the six-rating human consensus (≥ 4 concordant votes; the single non-concordant borderline image corresponds to an unresolved split vote with 3 VALID, 2 FAILURE, and 1 AMBIGUOUS), while rating agreement with VLM reports pooled individual VLM–human concordance.

Sampling Stratum	Images	Pair Agreement	Fleiss' κ	Majority Concordance	Rating Agreement with VLM
`clear_pass`	90	91.26%	−0.0225	90 / 90 (100.00%)	516 / 540 (95.56%)
`clear_fail`	90	77.11%	−0.0546	90 / 90 (100.00%)	474 / 540 (87.78%)
`borderline`	100	79.93%	0.6136	99 / 100 (99.00%)	539 / 600 (89.83%)
Regular Pool	**280**	**82.67%**	**0.6613**	**279 / 280 (99.64%)**	**1529 / 1680 (91.01%)**
Attention Controls	20	86.67%	−0.0643	20 / 20 (100.00%)	112 / 120 (93.33%)
All Samples	**300**	**82.93%**	**0.6664**	**299 / 300 (99.67%)**	**1641 / 1800 (91.17%)**

Table 19: Confusion matrix comparing VLM image judgments against the six-rating human consensus on the 279 resolved regular benchmark images (excluding one unresolved borderline image with a split vote). Rows denote the human majority decision (≥ 4 concordant votes), and columns denote the corresponding VLM evaluation.

	VLM evaluated as VALID	VLM evaluated as FAILURE	Total
Human majority evaluated as VALID	135	0	135
Human majority evaluated as FAILURE	0	144	144
Total	135	144	279

ratings). At the consensus level, exactly one regular image in the `borderline` stratum failed to achieve four concordant votes (receiving 3 VALID, 2 FAILURE, and 1 AMBIGUOUS), reflecting an edge case where human perception itself is evenly divided.

Table 19 presents the confusion matrix between the VLM evaluator and the human consensus on the 279 resolved benchmark images. Across the resolved pool, the VLM matches the human majority on all 279 instances (100.00% accuracy, precision, and recall). Individual rating divergences reflect minority variance rather than systematic divergence from the majority consensus.

Relationship to Sequential Exploration. This audit validates image-level evaluator consistency on the FailureAtlas SD3.5 benchmark. **HASTE**'s primary experimental objective is identifying condition-level failure sets under bounded evaluation budgets relative to a full-budget reference. The high concordance between human consensus and the VLM confirms that the underlying binary observations reliably reflect visual correctness. Image-level fidelity is complementary to, rather than identical with, sequential exploration efficiency: an accurate evaluator provides the requisite foundation for sequential early-stopping rules to preserve condition-level failure discovery. Furthermore, because individual human ratings exhibit an empirical disagreement rate of approximately 9% with the VLM (while the VLM achieves complete 100% agreement with the resolved majority consensus on all 279 benchmark images), pursuing zero divergence ($J = 1.0000$) from a specific empirical full-budget VLM trajectory yields diminishing returns. As analyzed in Appendix H, **HASTE**'s $> 97\%$ failure-set Jaccard and $> 99.4\%$ condition agreement operate right at this evaluator resolution limit, confirming that **HASTE** establishes the Pareto-optimal cost–fidelity trade-off against conservative baselines.

K EFFECT OF THE VISION-LANGUAGE MODEL EVALUATOR

Our primary evaluations in Section 5 employ `Qwen3-VL-8B-Instruct` on FailureAtlas, DSG, and T2I-CoReBench as the underlying visual question answering evaluator. To evaluate whether the sequential stopping efficiency and decision fidelity of **HASTE** generalize across distinct visual question answering backbones, we conduct complete fixed-K ($K = 50$) reference generation and sequential replays using both InternVL3.5-14B (`OpenGVLab/InternVL3_5-14B-Instruct`) (Wang et al., 2025) and BLIP-2 (`Salesforce/blip2-flan-t5-xl`) (Li et al., 2023).

FailureAtlas evaluation under InternVL3.5-14B. We evaluate SANA-1.5 on FailureAtlas ($K = 50$) using InternVL3.5-14B across validity decision thresholds $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$. Table 20 reports the sequential baseline comparison on the complete four-shard cohort (17,295 conditions, 864,750 evaluated images at $K = 50$). Across all decision thresholds, **HASTE** achieves the highest speedup among methods preserving high failure-set fidelity, delivering 3.73× speedup (0.9806 Jaccard) at $\tau = 0.5$ and scaling gracefully to 1.99× (0.9744 Jaccard) at $\tau = 0.9$. In contrast, confidence sequence bounds (Exact-SE, Hoeffding-CS, EB-CS, KL-CS) preserve exact sets ($J \geq 0.9966$) but yield limited speedups (1.01×–1.68×), while SPRT incurs degraded failure fidelity at intermediate thresholds ($J = 0.9043$ at $\tau = 0.8$). On the single-shard cohort (4,323 conditions, matching the BLIP-2 evaluation volume), **HASTE** similarly achieves 3.39 × /0.9739 at $\tau = 0.5$, 3.18 × /0.9899 at $\tau = 0.6$, 2.95 × /0.9984 at $\tau = 0.7$, 2.59 × /0.9935 at $\tau = 0.8$, and 3.00 × /0.9880 at $\tau = 0.9$.

Table 20: FailureAtlas sequential baseline evaluation with InternVL3.5-14B on SANA-1.5 across decision thresholds $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ (17,295 conditions, 864,750 evaluated images, $K = 50$, averaged over 5 observation seeds). Values report speedup × / failure-set Jaccard. Bold indicates the highest speedup among approximate methods.

Method	$\tau = 0.5$	$\tau = 0.6$	$\tau = 0.7$	$\tau = 0.8$	$\tau = 0.9$
Exact-SE	1.56× / 1.0000	1.45× / 1.0000	1.38× / 1.0000	1.35× / 1.0000	1.41× / 1.0000
Hoeffding-CS	1.43× / 1.0000	1.20× / 1.0000	1.06× / 1.0000	1.10× / 1.0000	1.14× / 1.0000
EB-CS	1.03× / 1.0000	1.01× / 1.0000	1.01× / 1.0000	1.02× / 1.0000	1.03× / 1.0000
KL-CS	1.68× / 1.0000	1.50× / 0.9999	1.31× / 0.9999	1.17× / 0.9986	1.22× / 0.9966
SPRT	3.04× / 0.9435	3.03× / 0.9618	3.02× / 0.9516	3.16× / 0.9043	1.90× / 0.9799
APT	3.25× / 0.9377	2.87× / 0.9630	2.26× / 0.9850	1.58× / 0.9960	1.29× / 0.9992
GrAPL	3.24× / 0.9446	2.84× / 0.9637	2.25× / 0.9853	1.58× / 0.9960	1.29× / 0.9992
HASTE (Ours)	**3.73× / 0.9806**	**3.51× / 0.9718**	**3.19× / 0.9873**	**2.59× / 0.9891**	**1.99× / 0.9744**

BLIP-2 evaluation configuration. We further evaluate SDXL-Turbo under BLIP-2 across both diagnostic benchmarks:

1. **FailureAtlas Option 5 subset** (4,323 conditions, 213,680 reference image evaluations under tree traversal at $K = 50$) at standard threshold $\tau = 0.5$.
2. **DSG benchmark** (132 items, 6,600 generated images, 49,833 VQA proposition queries) under a conservative decision threshold $\tau = 0.8$.

All methods are replayed over five deterministic observation-order seeds using the identical pre-generated image pools and verified BLIP-2 judgments.

FailureAtlas BLIP-2 evaluation ($\tau = 0.5$). Table 21 presents the sequential baseline comparison on FailureAtlas under BLIP-2 at $\tau = 0.5$. **HASTE** achieves a 4.47× speedup over the fixed-50 reference (213,680 reference images, reducing cost to 47,803.0) while maintaining a high failure-set Jaccard of 0.9265 and node state agreement of 0.9894. Among external sequential testing baselines, SPRT (3.13 × /0.8428), APT (3.52 × /0.8580), and GrAPL (3.52 × /0.8609) incur both lower speedups and reduced failure-set fidelity. Lossless certification baselines (Exact-SE, Hoeffding-CS, EB-CS, KL-CS) preserve exact failure sets ($J = 1.0000$) but yield limited speedups (1.04×–1.99×), requiring more than double the computation of **HASTE**.

DSG BLIP-2 evaluation ($\tau = 0.8$). Table 22 reports sequential baseline results on DSG under BLIP-2 at a strict validity threshold of $\tau = 0.8$. **HASTE** achieves a 1.75× speedup with 0.9870 failure-set Jaccard and 0.9940 proposition agreement. SPRT achieves 1.42× speedup with 0.9865 failure Jaccard, while APT (1.26×), GrAPL (1.12×), and confidence sequence bounds (1.00×–1.18×) struggle to terminate early under the asymmetric threshold $\tau = 0.8$. These findings confirm that **HASTE** maintains robust efficiency and high decision fidelity across different vision-language model architectures and decision thresholds.

DSG BLIP-1 evaluation across validity thresholds $\tau \in [0.5, 0.9]$. To evaluate behavior under lighter, first-generation VLM evaluators, we additionally evaluate SDXL-Turbo on DSG ($K = 50$,

Table 21: FailureAtlas sequential baseline evaluation with BLIP-2 on SDXL-Turbo at $\tau = 0.5$ (4,323 nodes, 213,680 reference images, $K = 50$, averaged over 5 observation seeds). Bold indicates the highest speedup among approximate methods.

Method	Category	Speedup ↑	Min. Fail. Jaccard ↑	Node Agreement ↑	Logical Image Cost ↓
Exact-SE	Lossless Certificate	1.62×	1.0000	1.0000	131,925.8
Hoeffding-CS	Confidence Sequence	1.63×	1.0000	1.0000	130,745.2
EB-CS	Confidence Sequence	1.04×	1.0000	1.0000	204,618.0
KL-CS	Confidence Sequence	1.99×	1.0000	1.0000	107,313.2
SPRT	Sequential Hypothesis Test	3.13×	0.8428	0.9899	68,216.0
APT	Sequential Elimination	3.52×	0.8580	0.9922	60,744.2
GrAPL	Graph-regularized Bandit	3.52×	0.8609	0.9924	60,682.2
HASTE (Ours)	Proposed Framework	**4.47×**	**0.9265**	**0.9894**	**47,803.0**

Table 22: DSG sequential baseline evaluation with BLIP-2 on SDXL-Turbo at $\tau = 0.8$ (132 items, $K = 50$, averaged over 5 observation seeds). Bold indicates the highest speedup among methods maintaining > 0.985 failure Jaccard.

Method	Category	Speedup ↑	Fail. Jaccard ↑	Prop. Agreement ↑	Realized Image Cost ↓
Exact-SE	Lossless Certificate	1.18×	1.0000	1.0000	5,594.2
Hoeffding-CS	Confidence Sequence	1.02×	1.0000	1.0000	6,491.8
EB-CS	Confidence Sequence	1.00×	1.0000	1.0000	6,600.0
KL-CS	Confidence Sequence	1.03×	1.0000	1.0000	6,431.8
SPRT	Sequential Hypothesis Test	1.42×	0.9865	0.9938	4,638.1
APT	Sequential Elimination	1.26×	0.9996	0.9998	5,232.0
GrAPL	Graph-regularized Bandit	1.12×	0.7821	0.8743	5,867.4
HASTE (Ours)	Proposed Framework	**1.75×**	**0.9870**	**0.9940**	**3,761.4**

1,060 items, 53,000 reference images) using BLIP-1 (`Salesforce/blip-vqa-base`) (Li et al., 2022) across validity thresholds $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$. Table 23 reports the evaluation speedup across all eight sequential baseline methods. **HASTE** achieves the highest speedup across the entire operating range, delivering $1.42\times$ speedup at $\tau = 0.5$ (compared to $1.17\times$ for Exact-SE and $1.34\times$ for SPRT), $1.39\times$ at $\tau = 0.6$, $1.33\times$ at $\tau = 0.7$, $1.24\times$ at $\tau = 0.8$, and $1.10\times$ at $\tau = 0.9$. Confidence sequence methods (Hoeffding-CS, EB-CS, KL-CS) quickly collapse to $1.00\times$ at higher thresholds, whereas **HASTE** reliably maintains adaptive early stopping efficiency across all tested validity thresholds.

Table 23: Sequential evaluation speedup with BLIP-1 on SDXL-Turbo across validity decision thresholds $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ on DSG ($K = 50$, 1,060 items, 53,000 images). Bold indicates the highest speedup in each column.

Method	$\tau = 0.5$	$\tau = 0.6$	$\tau = 0.7$	$\tau = 0.8$	$\tau = 0.9$
Exact-SE	1.17×	1.13×	1.09×	1.06×	1.03×
Hoeffding-CS	1.15×	1.06×	1.00×	1.00×	1.00×
EB-CS	1.00×	1.00×	1.00×	1.00×	1.00×
KL-CS	1.21×	1.16×	1.09×	1.01×	1.00×
SPRT	1.34×	1.32×	1.28×	1.24×	1.11×
APT	1.39×	1.32×	1.23×	1.10×	1.00×
GrAPL	1.39×	1.32×	1.23×	1.09×	1.00×
HASTE (Ours)	**1.42×**	**1.39×**	**1.33×**	**1.24×**	**1.10×**

Sensitivity across validity thresholds $\tau \in [0.5, 0.9]$. Figure 7 illustrates the evaluation speedup of **HASTE** alongside the seven baseline algorithms as the validity decision threshold τ varies from 0.5 to 0.9 on both FailureAtlas and DSG under BLIP-2. **HASTE** consistently achieves the highest

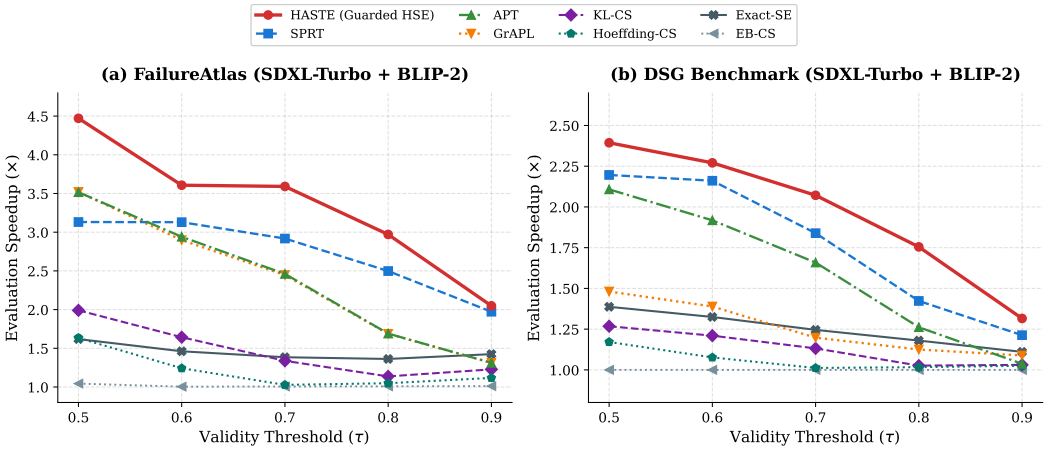


Figure 7: Evaluation speedup versus validity decision threshold $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ across eight sequential evaluation methods on SDXL-Turbo using BLIP-2. *Left:* FailureAtlas (4,323 conditions, $K = 50$). *Right:* DSG benchmark (132 items, $K = 50$). **HASTE** consistently achieves the highest speedup across the complete threshold range.

speedup across all evaluated decision thresholds, demonstrating that the sample efficiency gains of hierarchical evidence routing are invariant to threshold stringency.

L FailureAtlas Frontier Scheduling

The sequential evaluator does not require active condition selection. For FailureAtlas, independently generated nodes can nevertheless be reordered without changing the evidence used for an individual decision. We consider an optional outer policy π_t that directs a cumulative discovery budget toward failure-dense subtrees.

At search step t, let $\mathcal{F}_t$ denote the set of eligible frontier nodes whose concrete parents are all confirmed VALID. A lightweight surrogate model ϕ_t (e.g., gradient boosted regression trees) predicts the likelihood that condition v constitutes a failure:

$$\widehat{p}_t^{\mathrm{F}}(v) = \Pr_{\phi_t}(d_v = \textsc{Failure} \mid x_v), \tag{40}$$

where feature vector x_v encodes pre-generation prompt text embeddings and topological graph descriptors. The failure-greedy policy selects the candidate with maximum estimated failure likelihood:

$$v_t = \arg\max_{v \in \mathcal{F}_t} \widehat{p}_t^{\mathrm{F}}(v). \tag{41}$$

Upon selecting v_t, **HASTE** applies the sequential evaluator. If $d_t = \textsc{Failure}$, v_t is preserved as a minimal failure and its descendants are pruned; if $d_t = \textsc{Valid}$, its immediate child nodes become eligible for future frontier inclusion. We denote this failure-greedy composition as **HASTE** (Ours) (runnable identifier `failure_greedy_hse`). Every compared frontier policy uses the same **HASTE** stopping rule and cumulative logical-image-budget contract. This policy is evaluated only on the independently generated FailureAtlas stream; it is not part of the DSG or T2I-CoReBench evaluation contract.

Budget-truncated policy replay and novelty utility. To evaluate active frontier exploration under strict cumulative generation budgets, we perform budget-truncated policy replay on a 17,812-node FailureAtlas DAG across 20 random seeds under the shared **HASTE** evaluator. For nominal budget fraction $\alpha \in (0, 1]$, the recovered utility at cumulative budget αC_{ref} is defined as:

$$\mathrm{Utility@}\,\alpha = \frac{U_{\mathrm{policy}}(\alpha C_{\mathrm{ref}})}{U_{\mathrm{ref}}}. \tag{42}$$

Table 24: Budget-truncated policy replay on FailureAtlas (17,812-node DAG, 20 random seeds, five frontier policies under the shared **HASTE** evaluator). Bold indicates the most efficient result in each metric.

Policy	Utility @ 25%	Utility @ 50%	Cost to 50%	Cost to 75%	Cost to 90%
Breadth-First Search (BFS)	56.03%	96.47%	98.0k	162.2k	207.5k
Random Frontier	52.11%	97.04%	106.6k	166.7k	204.5k
Non-factorized UCB	47.95%	96.82%	118.8k	180.1k	205.9k
Factorized UCB	47.77%	94.41%	117.3k	183.0k	215.6k
HASTE (Ours)	**65.73%**	**98.55%**	**73.5k**	**134.6k**	**183.4k**

Table 25: Empirical evaluation of surrogate modeling on FailureAtlas (50 entities, 130,634 nodes). *Top:* Cumulative image budget required to discover 50% and 90% of reference failures under frontier scheduling with **HASTE** sequential testing. *Bottom:* Reconstruction fidelity of direct surrogate classification without sequential evaluation. Red shading marks severe fidelity loss.

Policy / Method	50% Discovery	90% Discovery	Total Images	Failure Jaccard
Breadth-First Search (BFS)	3.26M	3.65M	3.78M	0.9823
Random Frontier	3.31M	3.69M	3.78M	0.9823
Surrogate-Guided (Ours)	**1.46M** (2.23×)	**2.95M** (1.24×)	3.78M	0.9823
Direct Surrogate Classification (No Sequential Evaluation)				
Surrogate-Only ($\tau = 0.8$)	—	—	**0**	0.4214
Surrogate Margin ($m = 0.15$)	—	—	0.45M	0.2865

Rather than treating all failures as uniformly valuable, failure utility explicitly rewards semantic family novelty. When a newly observed failure condition v is identified, its marginal utility gain is computed as:

$$\Delta U(v) = 0.7\left(\sqrt{n_p + 1} - \sqrt{n_p}\right) + 0.2\left(\sqrt{n_s + 1} - \sqrt{n_s}\right) + 0.1, \tag{43}$$

where n_p and n_s denote the counts of previously discovered failure conditions within its primary and secondary failure families, respectively. Repeated discoveries within saturated failure families therefore receive diminishing weight, guiding exploration toward structurally novel failure modes. Checkpoints finish the current sequential evaluation action, bounding terminal overshoot within 0–24 logical images.

Table 24 documents the seed-aggregated results across five frontier policies. Under tight generation constraints, **HASTE** (Ours) demonstrates a substantial early-discovery advantage, capturing 65.73% of total reference failure utility at only 25% of the reference image budget (compared to 56.03% for BFS, 52.11% for random frontier selection, and 47.77% for factorized UCB). Furthermore, **HASTE** (Ours) reaches 90% failure utility at 183.4k logical images, discovering the target set 24,127.7 images earlier than standard breadth-first search (207.5k).

Empirical comparison of frontier policies and surrogate fidelity. Table 25 presents the quantitative evaluation of the HistGBR surrogate on the 50-entity FailureAtlas corpus (130,634 nodes). Across 50 leave-one-entity-out folds, the surrogate obtains an ROC-AUC of 0.7400 for binary failure identification.

We evaluate two distinct operational regimes:

1. **Active Frontier Scheduling with Sequential Testing**: When combined with **HASTE** sequential testing, surrogate-guided frontier selection directs the cumulative sampling budget toward failure-dense subtrees. It reaches 50% of all reference failures within 1.46M generated images, achieving a 2.23× speedup over standard breadth-first search (3.26M images) and random frontier expansion (3.31M images). At 90% failure discovery, it consumes 2.95M images compared to 3.65M for BFS (a 19.3% budget reduction). Because all selected conditions undergo full sequential testing, the final atlas reconstruction preserves target fidelity (0.9823 failure-set Jaccard).

2. **Direct Surrogate Classification (No Sequential Sampling)**: We also test whether the surrogate can directly predict failure labels without generating images. Under direct thresholding, surrogate-only classification yields a failure-set Jaccard of only 0.4214. Restricting autonomous decisions to conditions with confident margin boundaries ($|\widehat{p}_t^{\mathrm{F}} - \tau| \geq 0.15$) still incurs an error rate of 33.0%, collapsing atlas Jaccard to 0.2865. This failure demonstrates why surrogate predictions cannot substitute for real image evaluation, confirming that surrogate models operate as an active search scheduler rather than a standalone sequential evaluation baseline in Tables 1 and 2.

M EVALUATION-STATE TRANSITIONS

The graph reducer operates only on concrete nodes. A child enters the frontier after every concrete parent becomes VALID. A FAILURE remains in the atlas as a minimal failure and marks its descendants `PRUNED_NONMINIMAL`. An UNRESOLVED node also retains its trace but marks descendants `BLOCKED_UNRESOLVED`. Pruning denotes a known non-minimal branch; blocking denotes insufficient evidence about a required parent.

N LIMITATIONS

The atlas depends on its vocabulary, prompt template, VLM evaluator, and validity threshold. Its pruning rule assumes a form of monotonicity that may fail when an added attribute disambiguates a prompt. Repeated generation reduces but does not eliminate sampling noise, and fixed-50 remains a finite-budget reference. Appendix I measures how decisions change at smaller and larger reference budgets.

The prior parameters of **HASTE** are operating-point choices, and logical sample savings translate to runtime reductions depending on the relative overhead of image generation versus visual scoring. While our primary benchmark comparisons rely on deterministic replays over validated observation pools to guarantee exact matched-cost comparability across baselines, Appendix H.5 provides live end-to-end online validation on FailureAtlas with Stable Diffusion 3.5. Accordingly, we treat **HASTE** as an efficient condition-level diagnostic procedure rather than an exhaustive account of model failures.