TEMPORAL QUANTIZATION FOR DIFFUSION MODELS WITH CERTIFIED ERROR BOUNDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models generate high-quality samples, but repeated evaluations of large denoisers make deployment computationally expensive. Post-training quantization (PTQ) can reduce this cost, but how to verify and control quantization error through temporal quantization schemes remains challenging. To address this, we study temporal quantization of diffusion models, allowing deterministic, stochastic, and mixed-precision PTQ schemes across timesteps. By leveraging the L_2 moment generating function (L_2-MGF), we develop the first closed-form, high-probability error bound for diffusion process under arbitrary quantization schedules. Based on this theoretical guarantee, we build a predictive quantization policy which selects timestep-wise quantization schemes to minimize the terminal error bound under a computational budget. Extensive experiments evaluate both the error bounds and the resulting policies. More than 98.8% of the generated samples are within the 95%-certified quantization error bound. Across selected benchmarks, our proposed method reduces terminal L_2 error by 23.1%, and LPIPS by 23.4% on average. On an NVIDIA RTX 5070Ti GPU, our method achieves at least 16.6% less GPU memory footprint and 12.9% less denoising time than the FP16 baseline.

1 INTRODUCTION

Diffusion models generate data by iteratively refining a noisy state with a learned denoiser (Ho et al., 2020; Song et al., 2021b). They underpin high-resolution text-to-image synthesis and solvers for noisy inverse problems (Podell et al., 2024; Chen et al., 2025; Chung et al., 2023). However, this iterative process is also a deployment bottleneck: generating a single sample often requires tens to hundreds of denoiser evaluations (Song et al., 2021a), while the UNet of SDXL alone has 2.6B parameters (Podell et al., 2024). Faster samplers and distillation reduce the number of network evaluations (Lu et al., 2022; Sauer et al., 2025). Quantization instead reduces the cost of each evaluation: it represents weights and activations with fewer bits and operates supported operations at lower precision (Li et al., 2023; 2025). Among existing quantization methods, post-training quantization (PTQ) uses a small calibration set without retraining the model and has enabled low-bit image and video generation on memory-constrained GPUs (Zhao et al., 2025a; Li et al., 2025).

Most PTQ methods apply the same quantization setting at every denoising step. However, the denoiser's activation distributions and sensitivity to quantization are changing along the diffusion process (Shang et al., 2023; Li et al., 2023; Huang et al., 2024). This fact motivates temporal quantization, which attempts to allocate higher precision to more sensitive steps under a fixed computation budget (He et al., 2023; Yao et al., 2025; Lewandowski et al., 2026). It lets the precision, the rounding rule, and the attached compensator change along the trajectory, so one sampling process can combine deterministic and stochastic rounding (Gupta et al., 2015).

Designing a reliable schedule requires understanding how local quantization errors affect the final sample. Once quantization happens at a timestep, the next denoiser receives a perturbed input. Its output then deviates because of both the accumulated quantization error and newly introduced perturbation. Nonlinear updates propagate these effects through complicated mechanisms across timesteps, making it challenging to capture the quantization impacts on the final sample. The rounding rule also affects the error statistics. Deterministic rounding can introduce systematic bias, while stochastic rounding introduces fluctuations that are zero-mean at the quantization operation but may

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

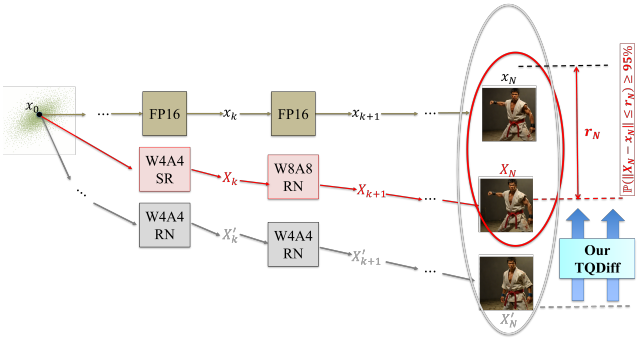

Figure 1: **Illustration of our TQDiff.** The brown trajectory is the full-precision process with the generation x_N. The grey trajectory is the typical quantization result. The red one is the diffusion process controlled by our TQDiff, which is within the certified error bound (red circle).

become biased after a nonlinear denoising update. Therefore, a schedule must account for both bias and random fluctuations, as well as their propagation through later steps.

Existing methods address different aspects of this problem. Calibration and reconstruction methods, including Q-Diffusion and SVDQuant, reduce errors in individual denoiser evaluations (Shang et al., 2023; Li et al., 2023; Huang et al., 2024; Li et al., 2025; Zhao et al., 2025a). Correction methods compensate for quantization noise, exposure bias, or accumulated errors (He et al., 2023; Yao et al., 2025; Liu et al., 2026a), while AccuQuant incorporates multiple denoising steps into calibration (Lee et al., 2025). Temporal mixed-precision methods select schedules using signal-to-noise ratios or an empirically validated additive-error model (He et al., 2023; Lewandowski et al., 2026). The closed-form error-propagation analysis of Liu et al. (2026a) relies on a first-order linearization of the denoiser. A remaining challenge is to bound, with certified probability level, the deviation between quantized and full-precision generations driven by the same initial and sampling noise. This leads to our central question: *Can we bound the quantization error for temporal quantized diffusion models, and use that bound to construct a low-cost quantization policy with error certification?*

In this work, we study temporal quantized diffusion models (TQDiffs), as illustrated in Figure 1. At each timestep, a TQDiff selects a quantization action from a library of deterministic, stochastic, and mixed-precision schemes. Our core idea is to split the one-step residual of any action into a conditional bias and a zero-mean stochastic uncertainty. We then propagate both parts through the exact nonlinear error dynamics and analyze it through an L_2 moment generating function (L_2-MGF) (Hsu et al., 2012; Liu et al., 2026b). It results in a closed-form, high-probability error bound whose parameters can be estimated offline. We develop a temporal quantization policy based on the bound: a predictive planner selects the schedule that minimizes the certified terminal error bound under a computation budget, and the selected schedule inherits the guarantee. Our contributions are as follows.

- **Certified bound on quantization error.** We derive an explicit high-probability bound on the quantization error of the terminal generation under mild assumptions (Theorem 4.2). The bound is the sum of three terms: propagated bias, accumulated variance of the stochastic uncertainty, and tail uncertainty. It covers temporal quantization schedules without linearizing the denoiser or assuming independence across timesteps, and it recovers the purely deterministic cases.
- **Temporal quantization policy with error guarantee.** We turn the bound into a criterion for selecting quantization actions under a computation budget and select the schedule through a predictive planner (11). Each PTQ method and fixed compensator is characterized by its complete update error, allowing heterogeneous actions to be compared within a common analytical framework. The resulting schedule satisfies the error bound under the stated assumptions.

Experiments on CIFAR-10 (Ho et al., 2020), SDXL-Turbo (Sauer et al., 2025), SDXL-base-1.0 (Podell et al., 2024), and PixArt (Chen et al., 2025) evaluate both the error bounds and the resulting policies. In most cases, the proposed TQDiff improves fidelity to the full-precision reference over the compared benchmarks, reducing similarity scores by more than 23% on average. Compared to

full-precision models, our TQDiff achieves at least 12.9% less GPU memory footprint and 16.6% less denoising time in our tests.

2 Related Works

2.1 Diffusion Model

Diffusion models generate samples by reversing a gradual noising process (Ho et al., 2020). In the score-based view, sampling solves a reverse-time stochastic differential equation or its probability-flow ordinary differential equation (ODE) (Song et al., 2021b). Text-conditioned models build on convolutional UNets (Podell et al., 2024) or diffusion transformers (Chen et al., 2025). Given a conditioning variable c (e.g., a text embedding), a sampler performs N updates

$$x_{k+1} = F_k(x_k; c), \qquad x_0 \sim \mu_0, \qquad k = 0, \dots, N-1, \tag{1}$$

where x_k is the noisy image or latent, μ_0 is typically Gaussian, and F_k is one full-precision update. The index k increases as the noise level decreases. A deterministic denoising diffusion implicit model (DDIM) update (Song et al., 2021a) has the form

$$F_k(x; c) = \alpha_k x + \gamma_k \epsilon_\theta(x, t_k, c), \tag{2}$$

where α_k, γ_k are sampler coefficients and ϵ_θ is the learned noise estimator at diffusion time t_k. For stochastic samplers, we condition on a shared sampling-noise path.

Generation cost is the number of steps times the cost of each step. High-order ODE solvers (Lu et al., 2022) and distillation (Sauer et al., 2025) reduce the number of steps. Quantization reduces the cost of each step, and our analysis applies to any sampler of the form (1), including distilled ones. On the theoretical side, convergence analyses bound the distance between the generated and data distributions in terms of score-estimation and discretization errors (Chen et al., 2023), but they do not model quantization. Our analysis is complementary: instead of comparing distributions, we provide high-probability bound on how far each quantized model generation deviates from its full-precision counterpart.

2.2 Quantization of Diffusion Models

Diffusion models are challenging to quantize because their activation distributions vary across noise levels and timesteps. PTQ4DM and Q-Diffusion pioneered post-training quantization (PTQ) for diffusion models (Shang et al., 2023; Li et al., 2023). Subsequent work addresses timestep dependence (Huang et al., 2024; Wang et al., 2024), quantization noise and exposure bias (He et al., 2023; Yao et al., 2025), few-step text-to-image generation (Zhao et al., 2025b), diffusion transformers (Wu et al., 2024), video generation (Zhao et al., 2025a), quantization-aware fine-tuning (He et al., 2024), and 4-bit weights and activations (Li et al., 2025). Most of these methods focus on local calibration, although each quantization error affects subsequent denoiser inputs. AccuQuant captures this accumulation by simulating multiple denoising steps during calibration (Lee et al., 2025). Timestep-aware cumulative error compensation (TCEC) uses per-step propagation equations to derive a compensator (Liu et al., 2026a), but its linearized analysis leaves the nonlinear remainder uncontrolled. A high-probability bound on the deviation from the full-precision trajectory remains missing. Our analysis complements these methods by leveraging L_2-MGF to incorporating each quantized update through its dynamics.

Formally, a quantized diffusion model (QDiff) replaces each F_k in (1) with an update $\widetilde{F}_k$ computed by a quantized denoiser $X_{k+1} = \widetilde{F}_k(X_k; c)$, $X_0 = x_0$, $k = 0, \dots, N-1$.. To accommodate randomized rounding, we work on a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_k)_{k=0}^N, \mathbb{P})$, where $\mathcal{F}_k$ contains the information available before step k. Round-to-nearest (RN), commonly used in PTQ, introduces deterministic error that can act as systematic bias. Stochastic rounding (SR) randomly rounds up or down. Without clipping, its local error is zero-mean, allowing partial cancellation across steps that yields probabilistic bounds on accumulated error that grow more slowly than deterministic bounds (Connolly et al., 2021; Croci et al., 2022; El Arar et al., 2023). SR also supports low-precision training (Gupta et al., 2015). These results concern arithmetic error accumulation, whereas diffusion sampling propagates errors through a nonlinear denoiser. Our analysis (Section 4) addresses this setting by decomposing each residual into a bias and a conditionally zero-mean fluctuation.

2.3 Temporal Quantized Diffusion Model

Sensitivity to quantization changes during denoising, so one precision need not suit every step. PTQD selects a bit width for each step (He et al., 2023). TMPDiff decides which timesteps run in full precision, using a bisection search that assumes quantization errors accumulate additively across timesteps (Lewandowski et al., 2026). MixDQ instead varies precision across layers (Zhao et al., 2025b). We generalize these schedules. An action a specifies the precision, the rounding rule, and an optional compensator, and a finite action library $\mathcal{A}_k$ provides the updates $\{\widetilde{F}_k^a : a \in \mathcal{A}_k\}$:

$$X_{k+1} = \widetilde{F}_k^{a_k}(X_k; c), \qquad X_0 = x_0, \qquad a_k \in \mathcal{A}_k. \tag{3}$$

The action is selected before fresh rounding randomness is drawn. We call (3) a temporal quantized diffusion model (TQDiff); a QDiff is the special case of a constant action. Writing Wb_wAb_a for b_w-bit weights and b_a-bit activations, an example library is $\mathcal{A}_k = \{\text{W8A8+RN}, \text{W4A4+SR}, \text{W4A8+SR}, \text{W4A4+RN+TCEC}\}$. Unlike existing temporal allocation methods, we select schedules by a high-probability error bound rather than an empirical error proxy, and our library can mix deterministic and stochastic actions within one schedule.

3 Problem Formulation and Preliminaries

This section formulates the problem we study and provides the assumptions of problem formulation. Then, it introduces some preliminaries of L_2-MGF, which is the main mathematical tool of our theoretical analysis.

3.1 Problem Formulation

To begin with, we formulate the dynamics of quantization error and the assumptions on it. At the timestep k of (3), define the one-step residual $R_k^{a_k} := \widetilde{F}_k^{a_k}(X_k; c) - F_k(X_k; c)$, which captures the effects of reduced precision, attached correction, and internal randomization caused by the current quantization step a_k. The dynamics of TQDiff (3) is therefore equivalent to

$$X_{k+1} = F_k(X_k; c) + R_k^{a_k}. \tag{4}$$

Define the quantization error $E_k := X_k - x_k$. Subtracting (1) from (4) gives the quantization error dynamics

$$E_{k+1} = F_k(X_k; c) - F_k(x_k; c) + R_k^{a_k}. \tag{5}$$

We point out that (5) uses the exact diffusion model dynamics $F_k(x; c)$ without any approximation. The only assumption on $F_k(x; c)$ is the local-Lipschitz property, as stated below.

Assumption 3.1 (Local Lipschitz Dynamics)**.** *For each timestep k, there exists a sufficiently large bounded region $\mathcal{X} \subset \mathbb{R}^d$ and a Lipschitz constant $\rho_k > 0$ such that $x_k \in \mathcal{X}$ and*

$$\|F_k(x; c) - F_k(y; c)\| \le \rho_k \|x - y\| \quad \textit{for all } x, y \in \mathcal{X}. \tag{6}$$

The term "sufficiently large" means that the local region $\mathcal{X}$ is large enough to contain the full-precision, quantized, and auxiliary diffusion trajectories used in the theoretical analysis. The necessary size of $\mathcal{X}_k$ will be quantified in Appendix D.1. The local-Lipschitz assumption is standard for almost all the real-world diffusion models, while learning and estimating the local Lipschitz constants ρ_k could be technical in applications. Relative technical details will be discussed in Appendix D.2.

For the quantized diffusion model (4) whose terminal is N, determining how large the quantization error E_N could be on the final generation is a fundamental problem. However, this is very challenging since the E_k is propagating through the dynamics (5), which is usually a large-scale network without closed-form expressions. Our goal is to establish high-probability bounds for the terminal quantization error E_N, and leverage the bound to design certified TQDiff, as formalized below.

For the TQDiff (3) throughout $k = 1, \ldots, N$, derive the bound r_N such that $\mathbb{P}(\|E_N\| \le r_N) \ge 1 - \delta$ holds for small δ (e.g., 0.1%). Then, based on the derive bound, develop an efficient and adaptive TQDiff (3) with error-bound guarantee.

3.2 Mathematical Preliminaries: L_2-MGF

Analyzing the quantization error E_k is challenging, as it contains both deterministic drifts and stochastic uncertainties. Unlike linear models, these two parts that are propagated through nonlinear dynamics (5), and thus are coupled in a complicated way. Therefore, it is necessary to find a proper mathematical tool to decouple. Fortunately, in the field of stochastic systems, the family of moment generating functions (MGFs) has turned out to be a good tool to decouple and simplify error propagation between stochastic trajectories (Jafarpour et al., 2025; Liu et al., 2026b). For our theoretical analysis, we choose L_2-MGF, which is originated from the proof technique in (Hsu et al., 2012, Thm 2.1), as the main weapon of our theoretical analysis.

Given a random vector $z \in \mathbb{R}^d$ and a variable $s > 0$, define L_2-MGF as:

$$\Phi(s) := \mathbb{E}_z[e^{s\|z\|^2}], \tag{7}$$

where $\mathcal{L}_s(z) := e^{s\|z\|^2}$ is called the *kernel* of L_2-MGF. Obviously $\mathcal{L}_s(z)$ is positive and increasing with s and $\|z\|$ when $s > 0$. The decoupling ability of L_2-MGF will be discussed in Appendix B.

4 Method

4.1 High-probability Bound on Quantization Error

To begin with, we decompose the one-step residual $R_k^{a_k}$ into mean and zero-mean uncertainty by $R_k^{a_k} = b_k^{a_k} + \xi_k^{a_k}$, where

$$b_k^{a_k} := \mathbb{E}[R_k^{a_k} \mid \mathcal{F}_k], \quad \xi_k^{a_k} := R_k^{a_k} - b_k^{a_k}. \tag{8}$$

Then $\mathbb{E}[\xi_k^{a_k} \mid \mathcal{F}_k] = 0$. Specifically, $\xi_k^{a_k} = 0$ when a_k is a deterministic quantization scheme. We consider the following assumption on $b_k^{a_k}$ and $\xi_k^{a_k}$.

Assumption 4.1 (Bound & Sub-Gaussian Residual)**.** *There exists a nonnegative $\mathcal{F}_k$-measurable variable β_k and an $\mathcal{F}_k$-measurable SPD matrix Σ_k such that:*

- $\|b_k^{a_k}\| \le \beta_k$ *holds almost surely.*
- *for every $\eta \in \mathbb{R}^d$, the following inequalities hold almost surely.*

$$\mathbb{E}\left[e^{\langle \eta, \xi_k^{a_k}\rangle} \,\middle|\, \mathcal{F}_k\right] \le e^{\eta^\top \Sigma_k \eta/2}, \quad \operatorname{tr}\Sigma_k \le \tau_k^2, \quad \lambda_{\max}(\Sigma_k) \le \ell_k. \tag{9}$$

The assumption on $b_k^{a_k}$ indicates that the quantization error has finite expectation, which is standard for real-world diffusion models. For ξ_k, we say that Σ_k is its *conditional proxy* if (9) holds. This assumption satisfies a wide range of ξ_k with Gaussian, Bernoulli or bounded stochasticity. The numerical methods to effectively estimate β_k, τ_k^2 and ℓ_k will be discussed in Appendix E.

Based on this decomposition, the high-probability bound on the terminal quantization error E_N can be established as the following theorem.

Theorem 4.2 (High-probability Bound on Quantization Error)**.** *Consider the full-precision diffusion model (1), the TQDiff (3), and their associated quantization error E_k that satisfies (5) along the denoising execution timeline $k = 0, \dots, N$ ($k = N$ corresponds to terminal generations). Suppose that Assumptions 3.1, 4.1 hold. At each timestep k, Define*

$$\begin{aligned} &D_0 = T_0 = V_0 = 0, \\ &D_{k+1} = \rho_k D_k + \beta_k, \quad T_{k+1} = \rho_k^2 T_k + \tau_k^2, \quad V_{k+1} = \rho_k^2 V_k + \ell_k, \\ &r_k = D_k + \sqrt{T_k} + \sqrt{2V_k \log(1/\delta)} \end{aligned} \tag{10}$$

Then, for any $\delta \in (0,1)$, it holds that $\mathbb{P}(\|E_N\| \le r_N) \ge 1 - \delta$.

Sketch of Proof. The full proof will be displayed in Appendix C. In short, the proof has three steps.

- First, we show that through a scaling technique introduced in Jafarpour et al. (2025), the quantization error E_k can be decomposed into a bounded drift and a centered stochastic uncertainty, without introducing any auxiliary trajectories. The decomposition technique is demonstrated in Lemma C.1.

- Second, we show that the two parts of E_k can be decoupled through L_2-MGF. This is the core step if the proof, and uses the decoupling property of L_2-MGF derived in Lemma B.1.
- Finally, a supermartingale $\mathcal{M}_k$ is constructed based on the decoupled E_k. Use martingale-based techniques to analyze $\mathcal{M}_k$, then we arrive the final conclusion.

To the best of our knowledge, Theorem 4.2 is the first closed-form quantization error for general quantized diffusion models. With this theorem, any TQDiff, including typical single-precision quantized DM, can be processed under certified error bound as long as the parameters ρ_k, β_k and Σ_k are properly estimated. In our experiments, we show that it is computationally tractable to obtain a good estimation for them. The details on parameter estimation are recorded in Appendix D,E.

Moreover, this theorem can be bridged to the error quantization mechanism of single-precision quantized diffusion model. When $\Sigma_k \equiv 0$, TQDiff reduces to a standard deterministic QDiff, and the bound of (10) reduces to D_k, which corresponds to the theoretical analysis in Liu et al. (2026a); Lewandowski et al. (2026).

4.2 Predictive Temporal Quantization Policy with Error Certification

With the certified quantization error bound, a natural thought is to leverage it to develop quantization policy with low-error-bound guarantee. Now suppose that the user needs to choose a quantization scheme a_k from a finite library $\mathcal{A}$ under the certification probability level δ. Define $C(a_k)$ as the computation cost of a_k and $\bar{C}$ as the computation cost budget. A predictive plan is to choose the sequence of a_k that can minimize the terminal error bound r_N under the budgets $\bar{C}$. To compute r_N, the local Lipschitz constants ρ_k can be estimated in prior, so D_k, T_k and V_k as (10) can be computed offline. For each candidate $a \in \mathcal{A}$, their related $\beta_k(a)$ and $\Sigma_k(a)$ are fixed and clear to the user, see Appendix E. The value of τ_k and ℓ_k follows Σ_k. Consequently, we can build a predictive policy that generates temporal quantization plans offline: Along $k = 0, \ldots, N$, the policy solves the following problem and choose a_k as its optimizer:

$$\begin{aligned} &\min\ r_N \\ s.t.\quad &a_k \in \mathcal{A},\ \sum_k C(a_k) \le \bar{C},\ \forall k = 0, \ldots, N-1, \\ &\tau_k \ge \sqrt{\operatorname{tr} \Sigma_k},\ \ell_k \ge \lambda_{\max}(\Sigma_k). \end{aligned} \tag{11}$$

Based on Theorem 4.2, this predictive temporal quantization method naturally provides a certified error bound. Compared to existing quantization methods, the proposed temporal quantization policy provides additional certified error bound and actively reduces the gap to the full-precision diffusion process. The effectiveness of this policy is validated in the experiments.

Computation cost with precision switch. For a TQDiff (3) that adopts this quantization policy with mixed precision, its computation cost $\sum_k C(a_k)$ is actually related to the time of precision switch. This is because for an $a_k \in \mathcal{A}$, $C(a_k)$ stays low if the precision of a_k is consistent with a_{k-1}, but actually becomes much higher if a_k requires a precision switch, which results in model reload on the GPUs. To avoid this hidden cost, we set an additional constraint to the predictive planner (11) that it should return a quantization schedule with very few precision switches. In our experiments, precision switch never happens more than 3 times per generation process, and reduces to a single-precision plan as possible when the budget $\bar{C}$ is tight.

Due to the sequential feature of (11), (11) can be handled through DP-based techniques. When the terminal time N is small, brutal search can be tractable. When N is large, an efficient pruning technique becomes necessary. Details of our pruning technique are provided in Appendix F.3.

5 Experiments

5.1 Experiment Setups

Models & Hardwares. We implement our methods with pretrained CIFAR-10 DDPM (Ho et al., 2020), SDXL-Turbo (Sauer et al., 2025), SDXL-base-1.0 (Podell et al., 2024), and PixArt (Chen

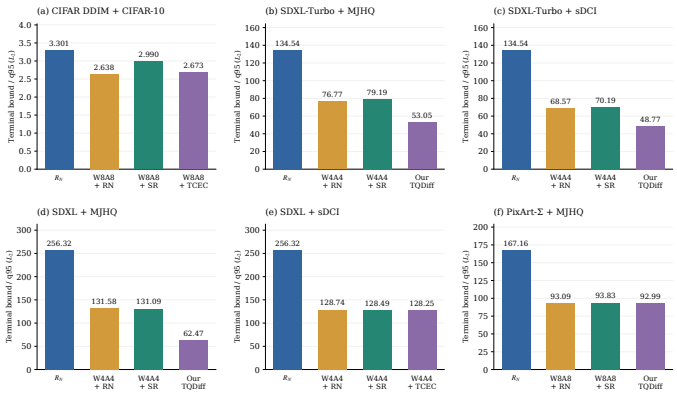

Figure 2: **Terminal error bounds and empirical error quantiles.** In each panel, the leftmost bar is R_N and the remaining bars are method-specific terminal q95 values. TQDiff denotes the proposed quantization policy; TCEC is the benchmark compensator in proposed in Liu et al. (2026a). Each method samples 5000 trajectories. Both datasets share the same radius for each text-to-image model.

et al., 2025). More details are provided in Appendix F.2. Most of our experiments are operated on a server with 5 NVIDIA RTX-A6000 GPUs unless specified.

Datasets. For text-to-image generation, the parameter estimation procedure uses randomly sampled COCO prompts (Lin et al., 2014). To validate the certified bound and and evaluate the performance of our quantization policy, we use fixed subsets of MJHQ-30K[1] and the sDCI benchmark used in SVDQuant (Li et al., 2025), with 5000 prompts per method per dataset. More details are provided in Appendix F.2.

Baselines of Quantization Methods. The TQDiff with our predictive temporal quantization policy is compared with round-to-nearest (RN), RN with timestep-aware cumulative error compensation (RN+TCEC) (Liu et al., 2026a), stochastic rounding (SR), and TMPDiff proposed in Lewandowski et al. (2026). For the methods implemented on CIFAR, we consider W4A8 and W4A6 quantization precisions. For those implemented on text-to-image tasks, we consider W4A4 and W8A8 precisions. More details are provided in Appendix F.2.

5.2 Main Results

Certification of Derived Error Bounds To begin with, we show that the error bound derived by Theorem 4.2 is an effective certification in the experiments. For each model, the parameters $\rho_k,\ \beta_k, \Sigma_k$ used to compute r_N are estimated through sample-based methods, and certified by a confidence verification mechanism. The details are discussed in Appendix E and D.

The certification results are reported in Figure 2. The six panels of Figure 2 compare the 95% terminal error bound $(r_N|\delta = 5\%)$ with the empirical 95% percentile of terminal error (q95) $\|X_N - x_N\|$ over 5000 sampled trajectories. Each model uses the same fixed certified error bound over all the evaluated datasets and tasks, indicating the effectiveness of the certified error bound across tasks. The derived r_N covers all the empirical q95 values, about 1.10–4.10 times over them. These comparisons indicates the effective error bound certifications of our derived bound. The conservatism level is acceptable, because it stays on the same level as q95 while leaves some robustness to worse cases.

Image Generation Quality Evaluation We conducted extensive experiments based on the aforementioned setup. Part of the comparison on the image generation quality is reported in Table 1, whose full version is in Appendix F.4, and a group of generated images is displayed in 3. Among most of the examined models, our predictive temporal quantization policy shows advantage on the similarity with FP16 generations except PixArt, where our quantization policy finally chose single-precision W8A8+RN.

[1] https://huggingface.co/datasets/playgroundai/MJHQ-30K

Table 1: **Part of Image generation quality comparison.** WxAy+RN means this method uses round-to-nearest to realize x-digit weight quantization, y-digit activation quantization. TCEC (Liu et al., 2026a) is a compensator-based benchmark quantization method. TMPDiff means the temporal mixed-precision QDiff proposed in (Lewandowski et al., 2026). Our TQDiff has an extinct advantage on LPIPS, IR, PSNR and Terminal L_2 error. Due to page limitation, the full table is in Appendix F.4.

Model	Dataset	Method	FID ↓	LPIPS ↓	Image Reward ↑	PSNR (dB) ↑	Terminal L_2 ↓	Error Bound Coverage (%)
CIFAR	-	W4A8 + RN	5.895	0.1102	-	26.867	6.646	99.22
		W4A8 + RN + TCEC	6.746	0.1022	-	27.807	6.113	99.04
		Our TQDiff	14.360	0.0625	-	33.724	2.395	99.00
SDXL	MJHQ	W4A4 + RN	51.794	0.3581	-0.3791	15.234	76.022	99.44
		TMPDiff	50.806	0.1566	-0.2020	20.728	41.511	100.00
		Our TQDiff	53.257	0.098	-0.1540	24.124	29.413	100.00
	sDCI	W4A4 + SR	64.337	0.3781	-0.6945	14.261	80.805	99.54
		TMPDiff	65.549	0.1586	-0.6807	19.759	43.278	100
		Our TQDiff	64.534	0.1045	-0.6659	22.418	32.449	100.00
SDXL-Turbo	MJHQ†	W4A4 + RN	24.424	0.2413	0.8143	18.483	51.502	100.00
		W4A4 + SR	24.575	0.2611	0.8119	18.030	54.198	100.00
		Our TQDiff	24.370	0.1562	0.8136	21.255	37.967	100.00
	sDCI	W4A4 + RN	25.103	0.2479	0.6607	18.590	48.906	99.98
		W4A4 + RN + TCEC	25.511	0.2500	0.6653	18.547	48.956	98.82
		Our TQDiff	25.280	0.1641	0.6738	21.331	36.556	100.00
PixArt-Σ	MJHQ	W8A8 + RN	37.741	0.0956	1.1417	24.58	51.006	99.32
		W8A8 + SR	37.594	0.1007	1.1387	24.29	52.461	99.80
		W4A4 + RN	81.686	0.3067	0.8625	18.501	100.550	100.00
		TMPDiff	38.074	0.2279	0.8625	19.630	87.97	100.00
		Our TQDiff	37.798	0.0959	0.9382	24.57	50.987	99.36

†On this setting, TMPDiff offers the exact "W4A4 + RN" plan.

On CIFAR, the quantization planner attempts to leverage the TCEC to reduce the gap to FP16 generations. Though the similarity scores are improved compared to the baseline methods, the FID score becomes much worse. This abnormally high FID appeared in repeated experiments, indicating that the relationship between FID score and quantization error bound can be highly nonlinear. It remains unclear to us and will be treated as our future directions. For the experiments on SDXL-Turbo + MJHQ, where we set a relatively tight budget on the computation time, both our quantization policy and the TMPDiff benchmark choose to use single-precision schemes to avoid the time on model reload. However, since our strategy can leverage stochastic rounding in the steps where β_k is potentially large, the proposed TQDiff performs a better ability to reduce the gap to FP16 generations, while the TMPDiff directly reduces to a single-precision RN scheme. On the other experiments where the precision switch are allowed, our quantization policy tends to actively choose higher precision schemes at the early stages, and switch to low-precision quantization later. This behavior partly corresponds to the sensitivity analysis of QDiff produced in Lewandowski et al. (2026).

Runtime and GPU Memory Savings We compare the speedup effect of our TQDiff with FP16 inference and other quantization methods using CUDA graphs on an NVIDIA GeForce RTX 5070 Ti, as our implementations are not perfectly compatible with 4-bit precision by Nunchaku on A6000. The comparison measures denoising time per image, excluding the time on text encoding, image decoding, and parameter estimation. Figure 4 (left) shows our TQDiff against this baseline. The records show 12.9%, 19.3%, 47.0%, and 21.3% time saving respectively.

For the GPU memory usage comparison, we point out that among all the experiments, at most one model stays in the GPU memory at any timestep. Figure 4 (right) reports the mid number of GPU memory usage for different methods among all the models we use. GPU memory usage of our

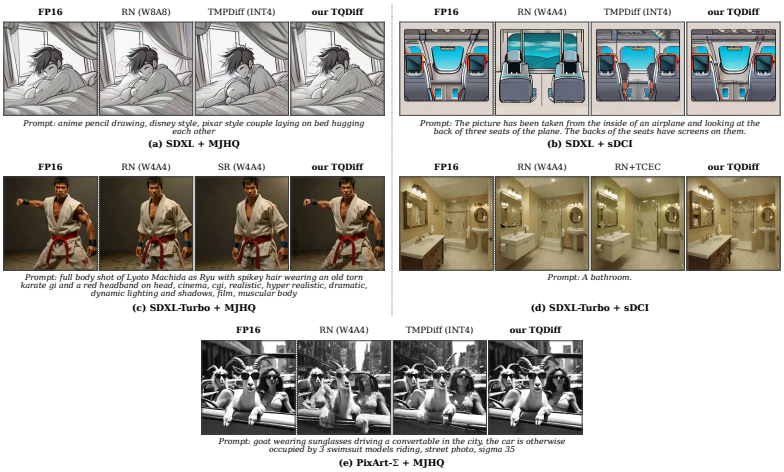

Figure 3: Visualization of the generated images. Each contains four images generated by full-precision model, benchmark quantization methods, and our quantization policy. More images can be found in Appendix F.4.

TQDiff shows 39.8%, 22.6&, 22.9% and 16.6% saving compared to the full-precision model, but is higher than single-precision quantized models under the same setup. It indicates that the memory usage of our proposed method is between single-precision quantization and full-precision process. This observation satisfies the intuition that the TQDiff takes more memory when switching to a higher-precision model, but less memory than constantly holding a full-precision model in GPUs.

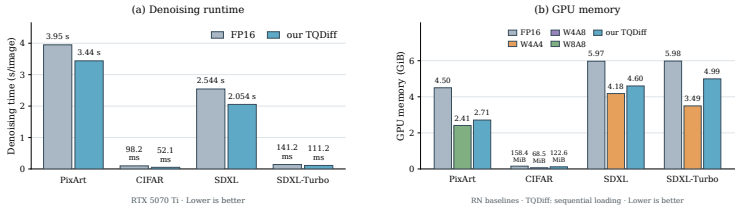

Figure 4: **Denoising runtime and GPU memory comparison.** Each pair corresponds to one model respectively. The quantizations are based on CUDA graphs. We show a clear speedup compared to FP16, and the GPU memory usage between FP16 and single-precision quantization.

6 Conclusion

In this paper we provide a temporal quantization method with certified error bound for diffusion models. By leveraging a mathematical tool called L_2-MGF, we established a high-probability bound on the quantization error of general nonlinear diffusion models. Furthermore, we developed a predictive temporal quantization policy that adaptively schedules quantization schemes to minimize the quantization error on the final generations. Extensive experiments were conducted to evaluate the derived error bounds and the proposed quantization policy. Reports show that our policy yields better similarity on the generated images with certified error bound. Future directions include further providing FID certifications to guarantee the generation quality of quantized models, and improving the implementation efficiency.

Reproducibility Statement

The manuscript states all mathematical assumptions and supplies complete proofs for the main theorem. A final submission should release all the experiment configurations for the reproduction of our experimental results.

AI Use Statement

Generative AI tools were used to assist paper writing and to improve the readability of the manuscript. All AI-assisted edits were reviewed and verified by the authors. The authors take full responsibility for the final content of the paper.

References

Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael Jordan. Distribution-free, risk-controlling prediction sets. *Journal of the ACM*, 68(6):1–34, 9 2021. ISSN 1557-735X. doi: 10.1145/3478535. URL http://dx.doi.org/10.1145/3478535.

Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. PIXART-Σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 74–91, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73411-3.

Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=zyLVMgsZ0U_.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OnD9zGAGT0k.

Michael P. Connolly, Nicholas J. Higham, and Theo Mary. Stochastic rounding and its probabilistic backward error analysis. *SIAM Journal on Scientific Computing*, 43(1):A566–A585, January 2021. ISSN 1095-7197. doi: 10.1137/20m1334796. URL http://dx.doi.org/10.1137/20M1334796.

Matteo Croci, Massimiliano Fasi, Nicholas J. Higham, Theo Mary, and Mantas Mikaitis. Stochastic rounding: implementation, error analysis and applications. *Royal Society Open Science*, 9(3), March 2022. ISSN 2054-5703. doi: 10.1098/rsos.211631. URL http://dx.doi.org/10.1098/rsos.211631.

El-Mehdi El Arar, Devan Sohier, Pablo de Oliveira Castro, and Eric Petit. Stochastic rounding variance and probabilistic bounds: A new approach. *SIAM Journal on Scientific Computing*, 45(5):C255–C275, October 2023. ISSN 1095-7197. doi: 10.1137/22m1510819. URL http://dx.doi.org/10.1137/22M1510819.

Suyog Gupta, Ankur Agrawal, Kailash Gopalakrishnan, and Pritish Narayanan. Deep learning with limited numerical precision. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1737–1746, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/gupta15.html.

Yefei He, Luping Liu, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Ptqd: Accurate post-training quantization for diffusion models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 13237–13249. Curran Associates, Inc., 2023. doi: 10.52202/075280-0580. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/2aab8a76c7e761b66eccaca0927787de-Paper-Conference.pdf.

Yefei He, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. EfficientDM: Efficient quantization-aware fine-tuning of low-bit diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=UmMa3UNDAz.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Daniel Hsu, Sham Kakade, and Tong Zhang. A tail inequality for quadratic forms of subgaussian random vectors. *Electronic Communications in Probability*, 17, January 2012. ISSN 1083-589X. doi: 10.1214/ecp.v17-2079. URL http://dx.doi.org/10.1214/ECP.v17-2079.

Yushi Huang, Ruihao Gong, Jing Liu, Tianlong Chen, and Xianglong Liu. Tfmq-dm: Temporal feature maintenance quantization for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7362–7371, June 2024.

Saber Jafarpour, Zishun Liu, and Yongxin Chen. Probabilistic reachability analysis of stochastic control systems. *IEEE Transactions on Automatic Control*, 70(11):7080–7094, 2025.

Seunghoon Lee, Jeongwoo Choi, Byunggwan Son, JaeHyeon Moon, Jeimin Jeon, and Bumsub Ham. Accuquant: Simulating multiple denoising steps for quantizing diffusion models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 110108–110143. Curran Associates, Inc., 2025. doi: 10.52202/085713-3679. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/9f8265edac41c4f8b4445b0d0cb83b80-Paper-Conference.pdf.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113 (523):1094–1111, 6 2018. ISSN 1537-274X. doi: 10.1080/01621459.2017.1307116. URL http://dx.doi.org/10.1080/01621459.2017.1307116.

Basile Lewandowski, Simon Kurz, Aditya Shankar, Robert Birke, Jian-Jia Chen, and Lydia Y. Chen. Tmpdiff: Temporal mixed-precision for diffusion models, 2026. URL https://arxiv.org/abs/2603.14062.

Muyang Li, Yujun Lin, Zhekai Zhang, Tianle Cai, Xiuyu Li, Junxian Guo, Enze Xie, Chenlin Meng, Jun-Yan Zhu, and Song Han. Svdquant: Absorbing outliers by low-rank component for 4-bit diffusion models. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 97831–97858, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/f34f0630c33be15b8c89426bb8056798-Paper-Conference.pdf.

Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-diffusion: Quantizing diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17535–17545, October 2023.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10602-1.

Songwei Liu, Chao Zeng, Chenqian Yan, Xurui Peng, XING WANG, Fangmin Chen, and Xing Mei. Error propagation mechanisms and compensation strategies for quantized diffusion models. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=4YIY66tpvQ.

Zishun Liu, Saber Jafarpour, and Yongxin Chen. Safety verification of nonlinear stochastic systems via probabilistic tube. *IEEE Transactions on Automatic Control*, 71(8):5173–5187, August 2026b. ISSN 2334-3303. doi: 10.1109/tac.2026.3670756. URL http://dx.doi.org/10.1109/TAC.2026.3670756.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan LI, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 5775–5787. Curran Associates, Inc., 2022. doi: 10.52202/068431-0418. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/260a14acce2a89dad36adc8eefe7c59e-Paper-Conference.pdf.

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 *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=di52zR8xgf.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 87–103, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73016-0.

Yuzhang Shang, Zhihang Yuan, Bin Xie, Bingzhe Wu, and Yan Yan. Post-training quantization on diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1972–1981, June 2023.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=St1giarCHLP.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=PxTIG12RRHS.

Changyuan Wang, Ziwei Wang, Xiuwei Xu, Yansong Tang, Jie Zhou, and Jiwen Lu. Towards accurate post-training quantization for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16026–16035, June 2024.

Junyi Wu, Haoxuan Wang, Yuzhang Shang, Mubarak Shah, and Yan Yan. Ptq4dit: Post-training quantization for diffusion transformers. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 62732–62755. Curran Associates, Inc., 2024. doi: 10.52202/079017-2006. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/72d32f4fe0b7af03732bd227bf1c4a5f-Paper-Conference.pdf.

Yuzhe Yao, Feng Tian, Jun Chen, Haonan Lin, Guang Dai, Yong Liu, and Jingdong Wang. Timestep-aware correction for quantized diffusion models. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 215–232, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-72848-8.

Tianchen Zhao, Tongcheng Fang, Haofeng Huang, Rui Wan, Widyadewi Soedarmadji, Enshu Liu, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, Huazhong Yang, Xuefei Ning, and Yu Wang. Vidit-q: Efficient and accurate quantization of diffusion transformers for image and video generation. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=E1N1oxd63b.

Tianchen Zhao, Xuefei Ning, Tongcheng Fang, Enshu Liu, Guyue Huang, Zinan Lin, Shengen Yan, Guohao Dai, and Yu Wang. Mixdq: Memory-efficient few-step text-to-image diffusion models with metric-decoupled mixed precision quantization. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 285–302, Cham, 2025b. Springer Nature Switzerland. ISBN 978-3-031-72630-9.

A Notations in Theoretical Analysis

We denote $\|\cdot\|$ as the L_2 norm, $B(a,r)$ as the ball $\{x \in \mathbb{R}^d : \|x-a\| \le r\}$ in the $\mathbb{R}^d$ space, $\mathbb{P}$ as the probability, $\mathrm{tr}(\cdot)$ as the trace of the matrix, $\lambda_{\max}(\cdot)$ as the maximal eigenvalue of the matrix.

B Decoupling Property of L_2-MGF

The core advantage of L_2-MGF for us is that it can decouple the drift and the centered stochastic uncertainty of a random variable, which is stated as below.

Lemma B.1 (Decoupling Property)**.** *Consider the two variables $h,\ \xi \in \mathbb{R}^d$, where h is $\mathcal{F}_k$-measurable and ξ satisfies for every $\eta \in \mathbb{R}^d$*

$$\mathbb{E}\Big[e^{\langle \eta,\xi\rangle}\ \Big|\ \mathcal{F}_k\Big] \le e^{\eta^\top \Sigma \eta/2}, \quad \mathrm{tr}\,\Sigma \le q, \quad \lambda_{\max}(\Sigma) \le v. \tag{12}$$

Then, for $0 \le 2sv < 1$,

$$\mathbb{E}[\mathcal{L}_s(h+\xi)\mid \mathcal{F}] \le \det(I-2s\Sigma)^{-1/2}\exp\big(sh^\top (I-2s\Sigma)^{-1}h\big) \tag{13}$$

$$\le \exp\left(\frac{sq}{1-2sv}\right)\mathcal{L}_{s/(1-2sv)}(h). \tag{14}$$

Proof. Notice that

$$\mathcal{L}_s(h+\xi) = e^{s\|h+\xi\|^2} = \mathbb{E}_{G\sim\mathcal{N}(0,I_d)} e^{\sqrt{2s}\langle G, h+\xi\rangle}. \tag{15}$$

Then, by Conditional Tonelli Theorem and (12), (15) implies

$$\mathbb{E}[\mathcal{L}_s(h+\xi)\mid\mathcal{F}] = \mathbb{E}_G\left[e^{\sqrt{2s}\langle G,h\rangle}\mathbb{E}\Big[e^{\sqrt{2s}\langle G,\xi\rangle}\mid \mathcal{F}, G\Big]\right] \tag{16}$$

$$\le \mathbb{E}_G \exp\Big(\sqrt{2s}\,\langle G,h\rangle + sG^\top\Sigma G\Big). \tag{17}$$

The case $s=0$ is immediate, so suppose $s>0$. The condition $2sv<1$ and $\Sigma \preceq vI$ imply $A := I-2s\Sigma \succ 0$ Writing the standard Gaussian density explicitly, the last line of (16) equals to

$$(2\pi)^{-d/2}\int_{\mathbb{R}^d}\exp\left(-\frac{1}{2}g^\top A g + \sqrt{2s}\,h^\top g\right)dg.$$

Set $m = A^{-1}\sqrt{2s}\,h$. then it can be further computed as

$$\begin{aligned}
&\mathbb{E}_G\exp\Big(\sqrt{2s}\,\langle G,h\rangle + sG^\top\Sigma G\Big)\\
&\quad= \exp\big(sh^\top A^{-1}h\big)\,(2\pi)^{-d/2}\int_{\mathbb{R}^d} e^{-\frac{1}{2}(g-m)^\top A(g-m)}\,dg\\
&\quad= \det(A)^{-1/2}\exp\big(sh^\top A^{-1}h\big),
\end{aligned}$$

Substituting $A = I-2s\Sigma$ proves (13).

It remains to prove (14). Since $0 \preceq \Sigma \preceq vI$, $I-2s\Sigma \succeq (1-2sv)I$, we get $(I-2s\Sigma)^{-1} \preceq (1-2sv)^{-1}I$, and thus

$$sh^\top(I-2s\Sigma)^{-1}h \le \frac{s}{1-2sv}\|h\|^2. \tag{18}$$

For the determinant term, let $\lambda_1,\dots,\lambda_d$ be the eigenvalues of Σ. Each satisfies $0\le\lambda_i\le v$, and

$$-\frac{1}{2}\log\det(I-2s\Sigma) = \frac{1}{2}\sum_{i=1}^d -\log(1-2s\lambda_i) \tag{19}$$

$$= \sum_{i=1}^d\int_0^{\lambda_i}\frac{s}{1-2su}\,du \tag{20}$$

$$\le \sum_{i=1}^d \frac{s\lambda_i}{1-2sv} = \frac{s\,\mathrm{tr}\,\Sigma}{1-2sv} \le \frac{sq}{1-2sv}. \tag{21}$$

Combining (18)(19) with (13) yields

$$\mathbb{E}[\mathcal{L}_s(h+\xi) \mid \mathcal{F}] \le \exp\left(\frac{sq}{1-2sv}\right) \exp\left(\frac{s}{1-2sv}\|h\|^2\right).$$

By (7), $\exp\left(\frac{s}{1-2sv}\|h\|^2\right) = \mathcal{L}_{s/(1-2sv)}(h)$, proving (14). □

C Proof of Theorem 4.2

Let $D_k,\ V_k,\ T_k$ be as Theorem 4.2. To begin with, we show that the quantization error E_k can be also decomposed into a bounded drift and a centered stochastic uncertainty, without introducing any auxiliary trajectories. This is stated in the following lemma.

Lemma C.1 (Projection-Based Decomposition)**.** *Suppose that Assumptions 3.1, 4.1 hold with $\mathcal{X}$ satisfying Appendix D.1. Then, there exists a decomposition $E_k = P_k + S_k$ such that*

$$\|P_k\| \le D_k, \quad P_0 = S_0 = 0, \quad S_{k+1} = H_k + \xi_k \tag{22}$$
$$\text{where } H_k \text{ is } \mathcal{F}_k\text{-measurable}, \ \|H_k\| \le \rho_k \|S_k\|.$$

Proof. Set $P_0 = 0$, and define $y_k = F_k(X_k;c) - F_k(x_k;c) + b_k$. Construct $P_k,\ H_k,\ S_k$ as:

$$P_{k+1} = \mathrm{proj}_{B(0,D_{k+1})}(y_k), \qquad H_k = y_k - P_{k+1}, \qquad S_k = E_k - P_k. \tag{23}$$

By (5), $E_{k+1} = y_k + \xi_k$, so the recursion in (22) is exact. The projection operator on P_k naturally ensures that $\|P_k\| \le D_k$. It remains to prove the gain bound on H_k. Notice that,

$$\left\| y_k - \mathrm{proj}_{B(0,D_k)}(y_k) \right\| = (\|y\|_k - D_k)_+,$$

where $(x)_+ := \max(x,0)$, hence $\|H_k\| = (\|y_k\| - D_{k+1})_+$. Then, by Assumption 3.1 and triangle inequality:

$$\begin{aligned}
\|y_k\| &\le \|F_k(X_k;c) - F_k(x_k;c)\| + \|b_k\| \\
&\le \rho_k \|X_k - x_k\| + \beta_k \\
&= \rho_k \|P_k + S_k\| + \beta_k \\
&\le \rho_k \|S_k\| + \rho_k \|P_k\| + \beta_k \\
&\le \rho_k \|S_k\| + \rho_k D_k + \beta_k.
\end{aligned}$$

Using $D_{k+1} = \rho_k D_k + \beta_k$, $\rho_k > 0$ and the monotonicity of $x \mapsto (x)_+$:

$$\begin{aligned}
\|H_k\| &= (\|y_k\| - D_{k+1})_+ \\
&\le \left(\rho_k \|S_k\| + \rho_k D_k + \beta_k - D_{k+1}\right)_+ \\
&= (\rho_k \|S_k\|)_+ = \rho_k \|S_k\|,
\end{aligned}$$

This proves Lemma C.1. □

The full proof of Theorem 4.2 can be unfolded based on Lemma C.1.

Proof of Theorem 4.2. Adopt the notations in Lemma C.1, then we know

$$S_{k+1} = H_k + \xi_k, \qquad \|H_k\| \le \rho_k \|S_k\|.$$

We start with the case where $\Sigma_k \neq 0$. Define $\Gamma_0 = 1,\ \Gamma_{k+1} = \rho_k \Gamma_k$. Obviously $\Gamma_k > 0$ for all k as ρ_k are positive. Scale the following variables:

$$Z_k = \frac{S_k}{\Gamma_k}, \quad h_k = \frac{H_k}{\Gamma_{k+1}}, \quad \zeta_k = \frac{\xi_k}{\Gamma_{k+1}}, \quad q_k = \frac{\tau_k^2}{\Gamma_{k+1}^2}, \quad v_k = \frac{\ell_k}{\Gamma_{k+1}^2}. \tag{24}$$

A direct computation derives that:

$$Z_{k+1} = \frac{H_k + \xi_k}{\Gamma_{k+1}} = h_k + \zeta_k,$$

and

$$\|h_k\| = \frac{\|H_k\|}{\Gamma_{k+1}} \le \frac{\rho_k \|S_k\|}{\rho_k \Gamma_k} = \|Z_k\|.$$

Moreover, $\widehat{\Sigma}_k := \frac{\Sigma_k}{\Gamma_{k+1}^2}$ is a conditional proxy of ζ_k, because for every $u \in \mathbb{R}^d$,

$$\mathbb{E}[e^{\langle u, \zeta_k \rangle} \mid \mathcal{F}_k] = \mathbb{E}\Big[e^{\langle u/\Gamma_{k+1}, \xi_k \rangle} \mid \mathcal{F}_k\Big] \le \exp\left(\frac{1}{2} u^\top \widehat{\Sigma}_k u\right).$$

It also implies that $\operatorname{tr} \widehat{\Sigma}_k \le q_k,\ \lambda_{\max}(\widehat{\Sigma}_k) \le v_k$.

For each timestep k, define $\widetilde{T}_k = \sum_{i<k} q_i = \frac{T_k}{\Gamma_k^2}$, $\widetilde{V}_k = \sum_{i<k} v_i = \frac{V_k}{\Gamma_k^2}$. These two variables will help us construct a nonnegative supermartingale. Fix a deterministic $w > 0$ and define

$$s_k = \frac{w}{1 + 2w\widetilde{V}_k} < w, \qquad \mathscr{M}_k = \exp\Big(s_k \|Z_k\|^2 - w\widetilde{T}_k\Big). \tag{25}$$

Since $\widetilde{V}_{k+1} = \widetilde{V}_k + v_k$,

$$1 - 2s_{k+1}v_k = 1 - \frac{2wv_k}{1 + 2w(\widetilde{V}_k + v_k)} = \frac{1 + 2w\widetilde{V}_k}{1 + 2w\widetilde{V}_{k+1}} > 0,$$

so Lemma B.1 is applicable with parameter s_{k+1}. In addition, $\frac{s_{k+1}}{1-2s_{k+1}v_k} = \frac{w}{1+2w\widetilde{V}_k} = s_k$. Therefore, conditionally on $\mathcal{F}_k$, Lemma B.1 gives

$$\begin{aligned}
\mathbb{E}[e^{s_{k+1}\|Z_{k+1}\|^2} \mid \mathcal{F}_k] &\le \exp\left(\frac{s_{k+1}q_k}{1 - 2s_{k+1}v_k}\right) \exp\left(\frac{s_{k+1}}{1 - 2s_{k+1}v_k} \|h_k\|^2\right) \\
&= e^{s_k q_k} e^{s_k \|h_k\|^2} \\
&\le e^{s_k q_k} e^{s_k \|Z_k\|^2}.
\end{aligned}$$

We now verify the supermartingale property of $\mathscr{M}_k$ explicitly. Since $\widetilde{T}_{k+1} = \widetilde{T}_k + q_k$,

$$\begin{aligned}
\mathbb{E}[\mathscr{M}_{k+1} \mid \mathcal{F}_k] &= e^{-w\widetilde{T}_{k+1}} \mathbb{E}[e^{s_{k+1}\|Z_{k+1}\|^2} \mid \mathcal{F}_k] \\
&\le \exp\Big(-w\widetilde{T}_k - wq_k + s_k q_k + s_k \|Z_k\|^2\Big) \\
&= \mathscr{M}_k\, e^{(s_k - w)q_k} \le \mathscr{M}_k,
\end{aligned}$$

Thus $(\mathscr{M}_k)_{k=0}^N$ is a nonnegative supermartingale. Also $\mathscr{M}_0 = 1$. Ville's inequality therefore gives

$$\mathbb{P}\left(\sup_{k \le N} \mathscr{M}_k \ge \delta^{-1}\right) \le \delta.$$

Let $\mathcal{E}$ denote the complementary event of "$\sup_{k \le N} \mathscr{M}_k \ge \delta^{-1}$", then $\mathbb{P}(\mathcal{E}) \ge 1 - \delta$. On $\mathcal{E}$, recall the definition of $\mathscr{M}_k$, we know simultaneously for every $k \le N$:

$$s_k \|Z_k\|^2 - w\widetilde{T}_k < \log(1/\delta).$$

Recall that $s_k = w/(1 + 2w\widetilde{V}_k)$, rearranging yields

$$\|Z_k\|^2 < \frac{w\widetilde{T}_k + \log(1/\delta)}{s_k} = (1 + 2w\widetilde{V}_k)\left(\widetilde{T}_k + \frac{\log(1/\delta)}{w}\right).$$

Finally, Lemma C.1 gives

$$\|X_k - x_k\| \le \|P_k\| + \|S_k\| \le D_k + \Gamma_k \|Z_k\|,$$

which proves that with probability at least $1 - \delta$:

$$\|X_k - x_k\| \le D_k + \Gamma_k \sqrt{(1 + 2w\widetilde{V}_k)\left(\widetilde{T}_k + \frac{\log(1/\delta)}{w}\right)} \quad \text{for every } k \le N. \tag{26}$$

Finally, set $t = \log(1/\delta)$, $k = N$ and $w = \sqrt{\frac{t}{2\widetilde{T}_k \widetilde{V}_k}}$, then (26) becomes the conclusion of Theorem 4.2. This completes the proof with $\Sigma_k \neq 0$.

If $\Sigma_k = 0$, then $\zeta_k = 0$ and $\xi_k = 0$ almost surely, which means the zero-mean stochastic uncertainty sequence vanishes almost surely. Starting from $S_0 = 0$, the inequality $\|H_k\| \le \rho_k \|S_k\|$ and the recursion $S_{k+1} = H_k$ then imply inductively that $S_k = 0$ for all $k \le N$. The theorem follows directly.

D Further Discussions on Assumption 3.1

D.1 Quantify The Sufficient Size of $\mathcal{X}$

We mentioned in Assumption 3.1 that the assumptions should hold in a "sufficiently large" domain $\mathcal{X}$, and all the diffusion trajectories should be processed within it. One may be carious about how large the $\mathcal{X}$ should be. In the following proposition, we quantify the area such that if $\mathcal{X}$ covers it, then the theoretical analysis we made remains correct. In short, everything remains correct as long as $\mathcal{X}$ covers a relatively large tube $B(x_k, \bar{r}_k)$, $k = 0, \dots N$ around the full-precision trajectory.

Proposition D.1. *Suppose that Assumption 4.1 holds and the terminal timestep $N \geq 2$. Adopt all the notations used in Theorem 4.2 and Appendix C. For $k = 1, \dots, N$, define $R_k(\varepsilon) := \sqrt{T_k} + \sqrt{2V_k \log(1/\varepsilon)}$, and*

$$\bar{r}_k = D_k + R_k\left(\delta/N\right), \quad 1 \leq k \leq N,$$

If $B(x_k, \bar{r}_k) \subseteq \mathcal{X}$ hols for every $k = 1, \dots, N$, then $\mathcal{X}$ is sufficiently large for Assumption 3.1 and Theorem 4.2. More precisely,

$$\mathbb{P}(\|X_k - x_k\| \leq \bar{r}_k \text{ for every } 1 \leq k \leq N) \geq 1 - \delta. \tag{27}$$

Proof. Use the decomposition $X_k - x_k = P_k + S_k$ and $Z_k = S_k/\Gamma_k$ from Appendix C, with $\|P_k\| \leq D_k$. Define

$$\kappa := \inf\{k \in \{0, \dots, N\} : X_k \notin \mathcal{X}\}, \qquad \overline{Z}_k := Z_{k \wedge \kappa}.$$

Since $X_0 = x_0 \in \mathcal{X}$, we have $\kappa \geq 1$. Before stopping, the local gain and innovation certificates hold; after stopping, the scaled drift is the identity and the stochastic uncertainty is zero.

For each fixed $k \in \{1, \dots, N\}$, define the event $\mathcal{E}_k := \left\{\Gamma_k \|\overline{Z}_k\| \leq R_k(\delta/N)\right\}$. Apply the argument of Appendix C with endpoint k and failure probability δ/N, it gives $\mathbb{P}(\mathcal{E}_k^c) \leq \frac{\delta}{N}$. Thus, for $\mathcal{E} := \bigcap_{k=1}^N \mathcal{E}_k$, a union bound yields

$$\mathbb{P}(\mathcal{E}) \geq 1 - \sum_{k=1}^{N} \mathbb{P}(\mathcal{E}_k^c) \geq 1 - \delta.$$

Suppose that $\kappa \leq N$ on $\mathcal{E}$. The stopped and original processes agree through κ, so

$$\|X_\kappa - x_\kappa\| \leq D_\kappa + \Gamma_\kappa \|\overline{Z}_\kappa\| \leq D_\kappa + R_\kappa(\delta/N) = \bar{r}_\kappa.$$

By the condition, this implies $X_\kappa \in B(x_\kappa, \bar{r}_\kappa) \subseteq \mathcal{X}$, contradicting the definition of κ. Hence $\kappa > N$ on $\mathcal{E}$, and therefore, simultaneously for all $k \leq N$,

$$\|X_k - x_k\| \leq D_k + \Gamma_k \|\overline{Z}_k\| \leq \bar{r}_k.$$

Together with $\mathbb{P}(\mathcal{E}) \geq 1 - \delta$, this proves (27). □

In many cases where the terminal step N is not large, the tube $B(x_k, \bar{r}_k), k = 1, \dots, N$ is only a little looser than the derived error bound r_N, as $\bar{r}_k$ at most introduces an additional $O(\log N)$ term which grows slowly with N. When the planned quantization schedule is deterministic, $\bar{r}_N = r_N$ almost surely. The following table gives a brief comparison between $\bar{r}_N$ and r_N in some of our experiments. Consequently, it is reasonable to estimate the error bound and verify its effectiveness within a bit larger area than the interested region in practice.

Table 2: Comparison between $\bar{r}_N$ and r_N in 3 experiments

Model	SDXL-Turbo	SDXL	PixArt-Σ
r_N	134.54	256.32	167.16
$\bar{r}_N$	139.07	260.516	181.68

D.2 Numerical Estimation on Local Lipschitz Constants ρ_k

For a complicated diffusion model dynamics $F_k(X_k; c)$, it is intractable to obtain the closed-form expression of ρ_k. Additionally, by Appendix D.1 it suffices that the bound holds on a reasonable tube around the reference trajectory. Therefore, we choose to estimate the local ρ_k through numerical methods in a restricted area where the diffusion process operates.

Paired-trajectory estimator. For each calibration prompt we roll out the full-precision reference $x_{0:N}$ and a coarse trajectory $X_{0:N}$ under a random schedule drawn from the certified action library, sharing conditioning and noise. By Assumption 3.1, each step yields one secant of the map:

$$\hat{\gamma}_k = \frac{\|F_k(X_k; c) - F_k(x_k; c)\|}{\|X_k - x_k\|}, \tag{28}$$

evaluated at a realized state and along a realized quantization-error direction. Collecting P pairs per step across prompts and schedules, we set $\hat{\rho}_k$ to the empirical $1.1 \max\{\hat{\gamma}_k\}$ and monitor the 0.9 quantile and maximum as sensitivity checks. At $k = 0$, we set $\rho_0 = 0$.

Sample-based estimation with confident certification. To ensure that the parameter estimations are reliable, it is necessary to provide finite-sample confidence verification. After estimating $(\rho_k, \beta_k, \Sigma_k)$, freeze the bound $\hat{r}_N$ computed from them. On an independent calibration split of m complete paired trajectories, sampled i.i.d. from the deployment distribution, define

$$u_N = \max\{\hat{r}_N, \varepsilon\}, \qquad S_i = \frac{\left\|E_N^{(i)}\right\|}{u_N}, \qquad \varepsilon > 0, \tag{29}$$

where ε avoids the case where $\hat{r}_N$ is close to 0. Write $S_{(j)}$ for the ordered scores, set $S_{(m+1)} = \infty$, and take $j = \lceil (m+1)(1-\delta) \rceil$ and $q = \max\{1, S_{(j)}\}$. Split-conformal calibration (Lei et al., 2018) gives, conditional on the specific way to compute r_N,

$$\mathbb{P}\{\|E_N^{\text{new}}\| \le q u_N \} \ge 1 - \delta. \tag{30}$$

The maximum score controls the confidence verification without independence across steps, nor conditional on each prompt. Appendix E strengthens this to an explicit confidence guarantee. In our experiments, q always return to 1.

E Numerical Ways to Approximate β and Σ for Assumption 4.1

Bias certificate β_k. At a fixed captured state X_k, we draw R independent residual samples $R_k^{a,(1)}, \ldots, R_k^{a,(R)}$, and form $\hat{b} = \frac{1}{R}\sum_i R_k^{a,(i)}$. Under our assumption, $\mathbb{E}\left\|\hat{b}\right\|^2 = \|b_k^a\|^2 + \operatorname{tr}\Sigma_k/R$, so the norm of $\hat{b}$ can be filtered by:

$$\hat{\beta}_k(a) = \left(\left\|\hat{b}\right\|^2 - \hat{\sigma}^2/R\right)_+^{1/2}, \qquad \hat{\sigma}^2 = \frac{1}{R-1}\sum_i \left\|R_k^{a,(i)} - \hat{b}\right\|^2, \tag{31}$$

where $(x)_+ = \max(x, 0)$ and $\hat{\sigma}^2$ is the unbiased estimate of $\operatorname{tr}\Sigma_k$. Deterministic actions need a single draw ($\xi_k^a \equiv 0$). The certificate $\beta_k(a)$ is the empirical envelope of $\hat{\beta}_k(a)$ over the probe states (we use the 0.8 quantile), a constant and hence trivially $\mathcal{F}_k$-measurable.

Proxy certificates τ_k^2, ℓ_k. From n separate replays at representative states we center the draws, $\hat{\xi}^{(i)} = R_k^{a,(i)} - \hat{b}$, and take the sample covariance $\hat{\Sigma}$: $\hat{\tau}_k^2(a) = \operatorname{tr}\hat{\Sigma} = \frac{1}{n-1}\sum_i \|\hat{\xi}^{(i)}\|^2$ and $\hat{\ell}_k(a) = \lambda_{\max}(\hat{\Sigma})$, computed from the $n \times n$ Gram matrix of the draws at cost $O(n^2 d)$ with $n \ll d$. Adopting $\hat{\Sigma}$ as its proxy is empirical and can be overly loose, while the confidence certification can verify whether we need to change a method. In experiments, the confidence certification always passed, and we consider the more rigorous estimation as a future work.

Deployment and Computation Cost. Nominally deterministic kernels can be nondeterministic in deployment (e.g., split-K atomic reductions on our GPUs). All certificates are per-cell and schedule-independent: they are estimated once and reused by the planner across all $\prod_k |\mathcal{A}_k|$ schedules (Appendix F.3), so the estimation cost is linear in $N\,|\mathcal{A}_k|\,(R+n)$, never exponential. In our most expensive case study ($N = 50$, $|\mathcal{A}_k| = 8$), the full certificate stack amounts to $\sim 4 \times 10^4$ one-step denoiser calls, and costs ¡0.5h on a single consumer GPU, which is small considering that it is an offline one-time computation.

Sample-based estimation with confident certification. As mentioned in Appendix D.2, a finite-sample guarantee can certify the deployment risk of the estimated parameters, following the risk-control perspective of (Bates et al., 2021). Use the independent i.i.d. calibration scores in (29), with the $u_{0:N}$ frozen beforehand. For $\delta \in (0,1)$ and calibration failure probability $\eta \in (0,1)$, choose the smallest deterministic rank $j_\eta \in \{1,\dots,m\}$ satisfying

$$\sum_{\ell=j_\eta}^{m} \binom{m}{\ell} (1-\delta)^\ell \delta^{m-\ell} \le \eta, \qquad q_\eta = \max\{1, S_{(j_\eta)}\}. \tag{32}$$

If no such rank exists, set $q_\eta = \infty$. The binomial order-statistic bound yields, conditional on the fitting split,

$$\mathbb{P}\{\mathbb{P}[\|E_N^{\text{new}}\| > q_\eta u_N \mid \text{cal}] \le \delta\} \ge 1 - \eta. \tag{33}$$

Thus, with confidence $1-\eta$ over calibration, the frozen procedure has deployment violation probability at most δ. Exact parameter recovery and infinitely many trajectories are unnecessary, while this finite-sample mechanism determine the confidence and tightness.

F Additional Implementation Details

F.1 Hardware Configurations of Experiments

Two compters are used in the experiments described in Section 5. (A) contains 5 NVIDIA RTX-A6000 GPUs, 2 AMD EPYC-73F3 CPUs, and 256GB RAM, and most of the experiments are conducted here. (B) has one NVIDIA RTX-5070Ti GPU, one Intel Ultra 7 CPU, and 96GB RAM, only experiments that are not compatible with the original hardware of (A) were conducted here.

F.2 Details about Experiment Setup

We study unconditional generation with a pretrained CIFAR-10 DDPM (Ho et al., 2020) and text-to-image generation with SDXL-Turbo (Sauer et al., 2025) and SDXL-base-1.0 (Podell et al., 2024). These settings cover both short and long denoising trajectories. The CIFAR model generates 32×32 images using 100 deterministic DDIM steps (Song et al., 2021a). SDXL-Turbo uses four Euler ancestral steps without classifier-free guidance, whereas SDXL uses 50 DDIM steps with guidance scale 5; both are evaluated at 512×512. Within each comparison, the quantized and reference processes share prompts, initial noise, and sampler noise. The standalone CIFAR comparison retains its FP32 reference; temporal mixed-precision and text-to-image comparisons use matched FP16 references. CIFAR and the text-to-image experiments run on RTX A6000 GPUs, with latency and memory compared within the same hardware and deployment protocol.

The unconditional CIFAR-10 evaluation generates 5000 samples per method. For text-to-image generation, parameter estimation on $\rho_k,\ \beta_k,\ \Sigma_k$ uses 1024 COCO prompts (Lin et al., 2014), while the confidence validation for the estimation uses 256 separate COCO prompts and 500 MJHQ-30K test trajectories. To evaluate our derived bounds and proposed quantization policy, we use fixed subsets of MJHQ-30K and the sDCI benchmark used in SVDQuant (Li et al., 2025), with 5,000 prompts per method per dataset. Evaluation for the certified error bound additionally uses 1000 randomly planned denoising trajectories, which are isolated from the main generation tasks. We measure distributional quality with FID, text–image preference with ImageReward, and fidelity to the coupled full-precision reference with trajectory L_2 error, LPIPS, image MSE, and PSNR. FID implementations, LPIPS backbones, and evaluation sample counts are identified in the result tables.

As mentioned in Section 5, our method is compared with RN, RN+TCEC (Liu et al., 2026a), SR, and TMPDiff Lewandowski et al. (2026). For each method, Wb_wAb_a specifies the weight and activation

bit widths at quantized operators, e.g., W4A4 means the weight and activation quantization are both 4-digit. CIFAR uses an independent packed-W4A8 backend which is also compatible with W4A6. For the text-to-image tasks, RN and RN+TCEC realize W4A4 using SVDQuant/Nunchaku (Li et al., 2025) with rank-32, group-64 quantization. W8A8 is realized based on the MixDQ INT8 backend Zhao et al. (2025b). TCEC is a compensator added upon RN. It evaluated through our independent implementation under the guidance of (Liu et al., 2026a), and is orthogonal to any quantization choice. SR modifies the exposed activation-quantization operations. Through our implementation, SR, RN and RN+TCEC share the *same* quantized weight/ activation packages if they have the same precision.

In our method, $\mathcal{A}$ is allowed to include RN, RN+TCEC, SR and full-precision 16 digit (FP16, which means we do not quantize it at this step). The actual setting of $\mathcal{A}$ varies in different experiments.

F.3 Dynamic Programming and Pruning Techniques for Solving (11)

Iterating (11) forward and committing to a minimizer at each step costs only $O(\sum_k |\mathcal{A}_k|)$ certificate evaluations, but the decision is myopic. Enumerating (11) is cheap for short samplers (e.g., $8^4 = 4096$ schedules for $N = 4$, solved in milliseconds), but blows up exponentially in N (e.g., $8^{50} \approx 10^{45}$). We solve (11) to a certified accuracy by a forward dynamic program over *certificate states* with two pruning rules.

State-space form. The influence of $a_{0:k-1}$ on all future radii is fully summarized by the state (D_k, T_k, V_k), augmented with the accumulated cost $J_k = \sum_{i<k} C(a_i)$ when a trajectory-level budget is enforced (the per-step budgets of (11) simply restrict $\mathcal{A}_k$). Each action updates the state affinely with nonnegative coefficients, $(D, T, V) \mapsto \big(\rho_k D + \beta_k(a),\ \rho_k^2 T + \tau_k^2(a),\ \rho_k^2 V + \ell_k(a)\big)$, and the objective $r_N = D_N + \sqrt{T_N} + \sqrt{2V_N \log(1/\delta)}$ and all constraints are non-decreasing in every coordinate.

Rule 1: dominance pruning (lossless). If two partial schedules reach states with $(D, T, V, J) \le (D', T', V', J')$, the second state is discarded. This is because the objective and constraints are monotone. Order preservation only requires $\rho_k \ge 0$, no stronger assumption is needed, so the rule remains valid on our problem.

Rule 2: conservative grid rounding (bounded relaxation). Dominance alone can leave exponentially many mutually incomparable states. After every transition we therefore round each of D, T, V *up* onto the multiplicative grid $\{0\} \cup \{(1+\nu)^j : j \in \mathbb{Z}\}$ and merge states sharing a grid key, retaining the smallest J. Rounding up never invalidates a certificate as it only enlarges the reported radius. One can verify that the rounded state satisfies $\widehat{D}_N \le (1+\nu)^N D_N$, and likewise for T_N, V_N, hence for every schedule

$$r_N \ \le\ \widehat{r}_N \ \le\ (1+\nu)^N\, r_N. \tag{34}$$

A standard induction then shows that the returned schedule solves (11) up to the factor $(1+\nu)^N$. When $\nu = 10^{-3}$ and $N = 50$, this is at most a 5.2% relaxation.

Moreover, we require that the obtained plan should contain no more than 3 precision switches, otherwise the DP should be restarted. By leveraging this two rules, the computation burden of solving (11) becomes so lightweight that can be ignored. The following table records the time spent on solving (11) with different timesteps N.

Table 3: DP search time for solving (11).

Steps N	DP search time (s)
50	3.304234
4	0.002169
20	0.016298

Table 4: **Image generation quality comparison.** WxAy+RN means this method uses round-to-nearest to realize x-digit weight quantization, y-digit activation quantization. TCEC (Liu et al., 2026a) is an additional compensator added to RN-based quantization outputs. TMPDiff is the temporal mixed-precision QDiff proposed by (Lewandowski et al., 2026). Our TQDiff has an extinct advantage on LPIPS, IR, PSNR and Terminal L_2 error.

Model	Dataset	Method	FID ↓	LPIPS ↓	Image Reward ↑	PSNR (dB) ↑	Terminal L_2 ↓	Error Bound Coverage (%)
CIFAR	-	FP16	4.990	0.0000	-	∞	0.000	100.00
		W4A8 + RN	5.895	0.1102	-	26.867	6.646	99.22
		W4A8 + RN + TCEC	6.746	0.1022	-	27.807	6.113	99.04
		W4A8 + SR	6.725	0.0934	-	28.303	5.866	99.16
		Our TQDiff	14.360	0.0625	-	33.724	2.395	99.00
SDXL	MJHQ	FP16	52.672	0.0000	-0.2963	∞	0.000	100.00
		W4A4 + RN	51.794	0.3581	-0.3791	15.234	76.022	99.44
		W4A4 + RN + TCEC	81.180	0.4687	-0.5833	14.168	80.738	99.20
		W4A4 + SR	47.726	0.3741	-0.3306	15.065	77.033	99.34
		TMPDiff	50.806	0.1566	-0.2020	20.728	41.511	100.00
		Our TQDiff	53.257	0.098	-0.1540	24.124	29.413	100.00
	sDCI	FP16	71.521	0.0000	-0.6909	∞	0.000	100.00
		W4A4 + RN	69.408	0.3619	-0.7386	14.406	79.816	99.84
		W4A4 + RN + TCEC	104.016	0.4762	-0.9652	13.399	84.554	99.98
		W4A4 + SR	64.337	0.3781	-0.6945	14.261	80.805	99.54
		TMPDiff	65.549	0.1586	-0.6807	19.759	43.278	100
		Our TQDiff	64.534	0.1045	-0.6659	22.418	32.449	100.00
SDXL-Turbo	MJHQ†	FP16	24.393	0.0000	0.8246	∞	0.000	100.00
		W4A4 + RN	24.424	0.2413	0.8143	18.483	51.502	100.00
		W4A4 + RN + TCEC	24.962	0.2430	0.8093	18.435	51.590	100.00
		W4A4 + SR	24.575	0.2611	0.8119	18.030	54.198	100.00
		Our TQDiff	24.370	0.1562	0.8136	21.255	37.967	100.00
	sDCI	FP16	25.001	0.0000	0.6924	∞	0.000	100.00
		W4A4 + RN	25.103	0.2479	0.6607	18.590	48.906	99.98
		W4A4 + RN + TCEC	25.511	0.2500	0.6653	18.547	48.956	98.82
		W4A4 + SR	25.353	0.2653	0.6778	18.200	51.139	99.98
		TMPDiff	25.293	0.1655	0.6692	21.429	36.132	100.00
		Our TQDiff	25.280	0.1641	0.6738	21.331	36.556	100.00
PixArt-Σ	MJHQ	FP16	38.032	0.0000	1.1421	∞	0.000	100.00
		W8A8 + RN	37.741	0.0956	1.1417	24.58	51.006	99.32
		W8A8 + SR	37.594	0.1007	1.1387	24.29	52.461	99.80
		W4A4 + RN	81.686	0.3067	0.8625	18.501	100.550	100.00
		TMPDiff	38.074	0.2279	0.8625	19.630	87.97	100.00
		Our TQDiff	37.798	0.0959	0.9382	24.57	50.987	99.36

†On this setting, TMPDiff offers the exact "W4A4 + RN" plan.

F.4 Detailed Results of Image Generation Quality

This section displays the full version of the image generation quality comparison, including more generated figures (prompts are under each row). Evaluation and analysis have been provided in Section 5.

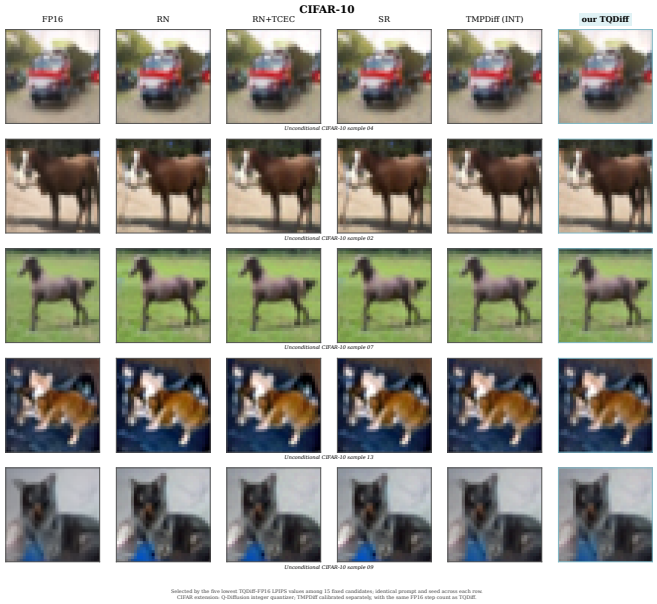

Figure 5: **Selected generated images on CIFAR-10.**

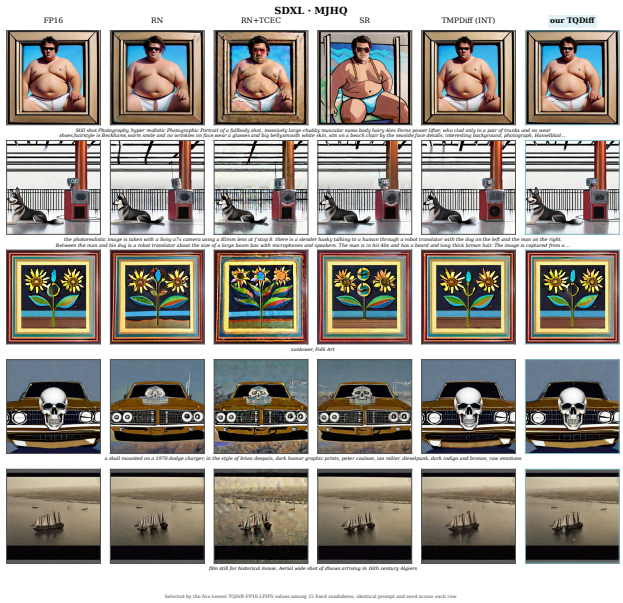

Figure 6: **Selected generated images: SDXL on MJHQ.**

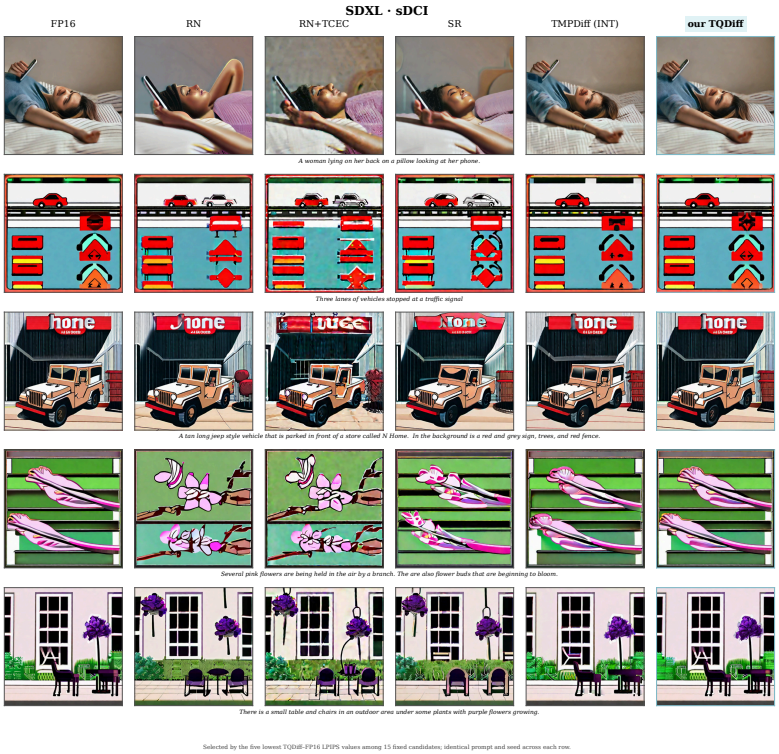

Figure 7: **Selected generated images: SDXL on sDCI.**

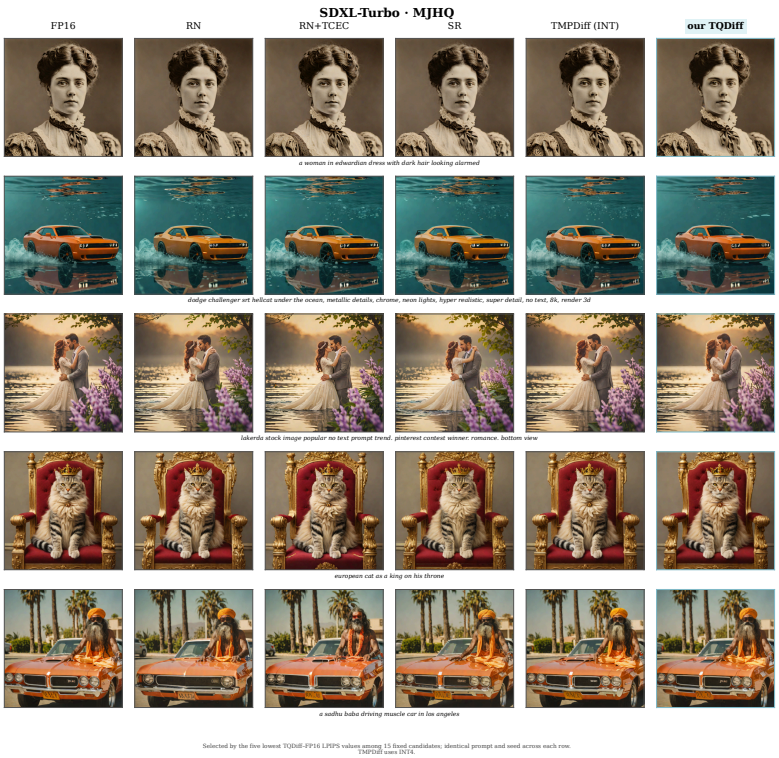

Figure 8: **Selected generated images: SDXL-Turbo on MJHQ.**

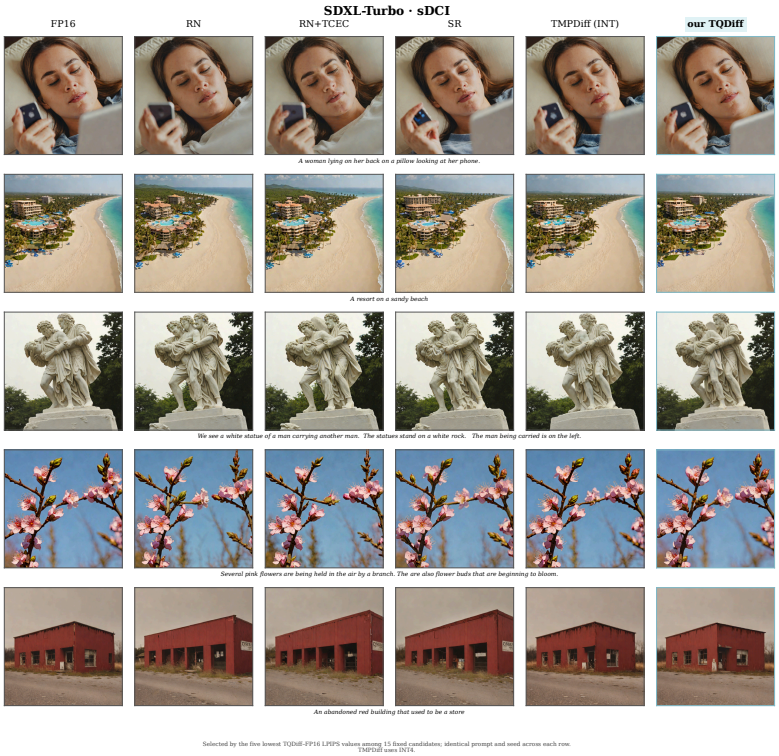

Figure 9: **Selected generated images: SDXL-Turbo on sDCI.**

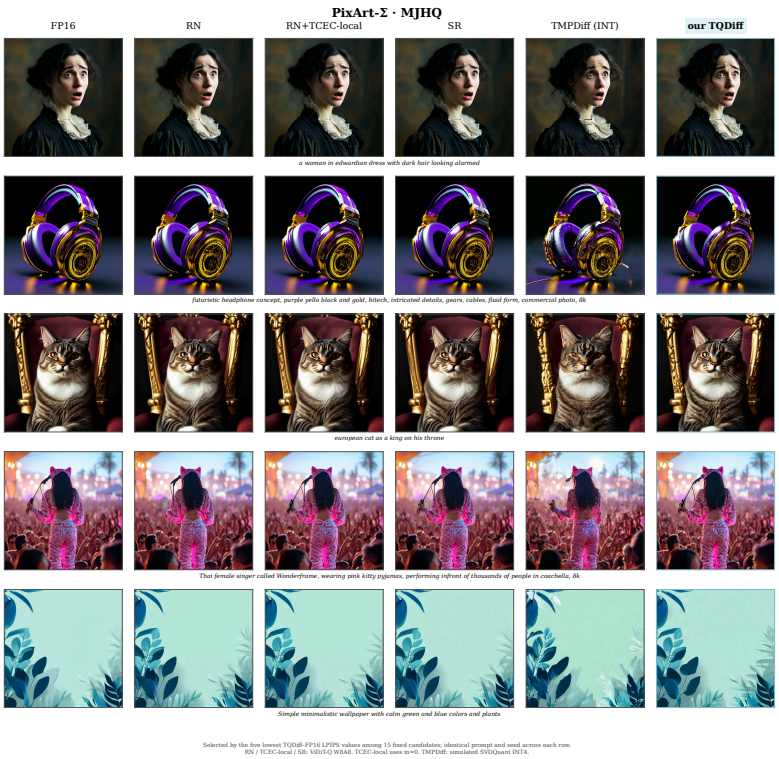

Figure 10: **Selected generated images: PixArt-Σ on MJHQ.**