Diffusion Inference with Dynamic Classifier-free Guidance

Aryan V S
Department of Computer Science
PES University
Bengaluru, Karnataka, India
contact.aryanvs@gmail.com

Ashish Kulkarni
Department of Computer Science
PES University
Bengaluru, Karnataka, India
ashish2002kulkarni@gmail.com

Dhruv Chawla
Department of Computer Science
PES University
Bengaluru, Karnataka, India
dchawla228@gmail.com

Aryan Rawther
Department of Computer Science
PES University
Bengaluru, Karnataka, India
aryanedu5786@gmail.com

Dr. Jayashree Rangareddy
Professor & Chairperson
Department of Computer Science (AI & ML)
PES University
Bengaluru, Karnataka, India
jayashree@pes.edu

***Abstract*—The introduction of the diffusion architecture has led to rapid progress in image generation. These models learn to iteratively generate images by predicting noise from an initial Gaussian distribution, which is learnt through forward and reverse diffusion. The process of iterative sampling occurs over a number of inference steps. Classifier guidance is a popular technique that guides the generation process of latent diffusion models (LDMs). It does this by adding an external classifier into the training process. This however requires additional training and severely limits the diversity of the synthesized data along with several other limitations. To improve on this, a novel approach, known as Classifier-Free Guidance (CFG), was introduced that removed the need for a separate classifier. It controls the sample generation process using a parameter known as the CFG scale. Previously, this scale has been set to a constant value during inference. This research work proposes varying the CFG scale values across the inference steps by making use of various scheduling functions, which not only results in better images but also unlocks the full potential of the rich latent space representation of diffusion models by allowing for the sampling of various different images from the same initial conditions by only varying the CFG scale and keeping all other parameters constant.**

***Index Terms*—classifier-free guidance, diffusion, image generation, inference, text-to-image**

I. Introduction

In the evolving landscape of generative models, diffusion-based models have emerged as a versatile and expressive family of models, demonstrating their effectiveness and mastery in tasks ranging from image, audio and video synthesis given a text prompt. The introduction of diffusion models, most notably Stable Diffusion [1], challenged well-established benchmarks in comparison to traditional autoregressive models.

Diffusion models can generate high quality samples by iteratively applying a series of transformations to a noise vector. The high level view of their training procedure involves sampling random data points from a data distribution and adding small increments of Gaussian noise to this sample over a series of time steps. This is known as the forward diffusion process. The model is trained to predict the noise addition using the reverse diffusion process, which essentially tries to recover the original sample given a noise vector. With the case of latent diffusion models, an additional step to encode training samples into a lower dimensional latent space is performed before forward diffusion. This also requires learning to decode a latent space vector into higher dimensional data space and is usually done by employing a variational autoencoder. Additionally, with the use of attention mechanisms [2], the model can be conditioned on various modals such as input in the form of text, audio, video, or anything else. Due to their versatile nature and sample quality, diffusion models have quickly rose to fame, challenging well-established benchmarks and leading to the creation of new ones. Some examples include the famous DALL-E models [3]–[5], Stable Diffusion [1] and SDXL models [6], Midjourney [7], Make-A-Video [8], Imagen [9], and many more.

Despite the success of diffusion based models, these models face a problem common between many generative approaches - striking a balance between sample fidelity and diversity. This quest for equilibrium has given rise to various strategies, with one notable technique being classifier guidance. This method tries to enhance the sample quality of diffusion models by incorporating an external classifier into the training process. This is effective, but introduces various complexities as it requires training an additional classifier on noisy data and severely limits the diversity of the synthesized data. In LDMs, the additional step of decoding latent space vectors into their higher dimensional data space adds a heavy computational burden, and although methods have been proposed to bypass this issue, there are various other limitations as well.

Enter the focus of this paper's exploration: classifier-free guidance for diffusion models [10]. This innovative approach challenges the conventional wisdom by eliminating the need for a separate classifier. Instead, two models are trained, one being conditionally aligned and the other being unconditional, aiming to achieve a nuanced trade-off between sample quality and diversity similar to that accomplished by classifier guidance. The construction of the ”classifier” from a generative model sets the stage for a robust gradient, facilitating a straightforward implementation for conditioning which streamlines the training process and results in much better sample quality and alignment to different modalities.

Classifier-free guidance requires two forward passes, one for conditional and unconditional models each, followed by a linear extrapolation (not interpolation because values in the ranges $[0,\ 1]$ are not typically used) between the two. In most implementations, however, the forward passes are combined into a single pass by concatenating their input latents. The factor for extrapolation has, traditionally, been fixed throughout the inference process to a constant scalar value called the CFG scale or $guidance\_scale$ (Stable Diffusion v1.5, for example, produces high quality images for CFG scale values between 5 and 15). This study explores dynamically scheduling of the CFG scale using various schedules and compare the results using FID [11] (Fréchet Inception Distance calculates distance between multivariate Gaussian distributions fitted to real and generated image features, providing a quantitative measure of their similarity), IS [12] (Inception Score evaluates the quality of sampled images by assessing two main aspects: diversity and classification confidence) and CLIP [13] (CLIP Score is a metric, which does not require a reference, that can be used to evaluate the correlation between the caption and content of the image. CLIP’s evaluation is not based on a single, universal metric but rather on task-specific metrics).

The main contributions of this research study are as follows:

- Investigation of the effect of dynamically scheduling the CFG scale at inference time on the resulting outputs through metrics like CLIP Score, FID and IS.
- For a fixed starting noise and CFG scale, the results are always the same (assuming hardware is the same). This study shows that dynamically scheduling CFG allows for a wide variety of generations and demonstrates the richness of the latent space of diffusion models.
- This study shows that certain CFG scheduling techniques produce more aesthetically appealing images overall, based on the evaluation metrics. These scheduling techniques are applied to multiple schedulers such as DDIM, DDPM, EulerAncestral, DPM Solver Singlestep, DPM Solver Multistep, UniPC, and LCM among others. Results are shown in the Appendix section of the paper along with other implementation details and prompts used. For evalutation, a model trained on the ImageNet-1k dataset and benchmarked on the Tiny ImageNet-1k split is used, due to compute limitations.

II. Related Works

A. Diffusion Models

Diffusion models are a type of generative models, which draw inspiration from non-equilibrium thermodynamics in physics. A Markov chain of diffusion steps is used to systematically introduce random noise (usually Gaussian) into data and subsequently learn to reverse this process, thereby enabling the generation of desired data samples starting from random noise. In contrast to variational autoencoders (VAEs) or flow models, diffusion models do not encode their data into a smaller latent space and instead operate in higher dimensionality space of the original data. Note that Stable Diffusion models, however, employ a VAE to first encode original data into lower dimensionality latent space and then perform diffusion learning as this enables faster training and yields better results.

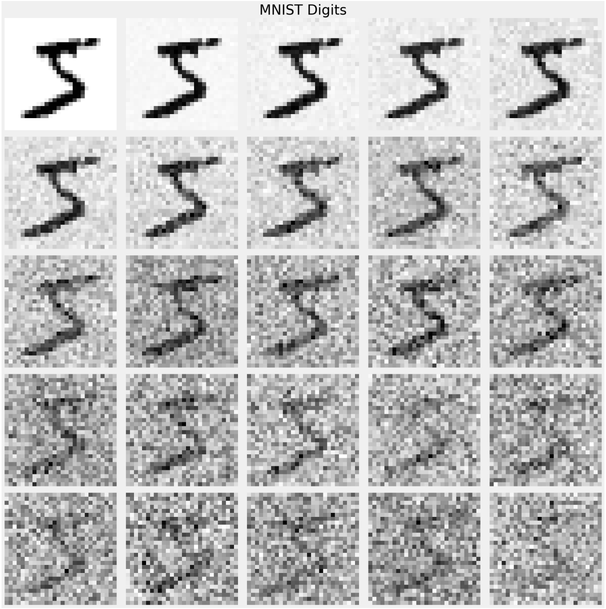

Fig. 1. Forward diffusion on MNIST

The training a diffusion model is performed in two steps: forward and reverse diffusion. The forward diffusion process involves taking a data point, sampled at random from a data distribution, and adding small increments of Gaussian noise to this sample over a series of timesteps (see Figure 1). The magnitude of noise added is controlled by variance schedules. With more forward diffusion steps, the sample progressively loses its distinctive features until nearly approximating a sample from a Gaussian distribution. The reverse diffusion process is similar to forward diffusion. Let x_t denote the noisy sample at timestep t and x_{t+1} denote the sample after the t’th Gaussian noise addition step in forward diffusion. During the reverse process, the goal is to recover the true sample from the noisy data. To achieve this, a model is trained to approximate the conditional probabilities that are necessary for this transformation. Mathematically, the model learns to estimate the conditional probability of transitioning from x_{t+1} back to x_t given the noise level and variance schedule.

B. Classifier Guidance

Guidance has proven to be a very powerful method for sampling and allows for addition control in diffusion models. It involves making use of gradients from an external classifier to provide a learning signal and steer the generation process so that model can align better with different input conditionings. This was shown to be very effective and dramatically improving generation quality by Dhariwal et. al. [14].

While using classifier guidance has its perks, it comes with additional costs. It requires an external image classifier that is robust to noisy images (as every step of the reverse diffusion process requires a gradient signal to make sure that the alignment to conditionings takes place). Even if there was such a classifier, there are other problems such as additional computational requirement and limited effectiveness due to information loss during classification because of a fixed set of output class activations. It is unsuitable and not scalable for a large corpus of training data, especially when working with different modalities.

C. Classifier-Free Guidance

Classifier-free guidance [10] guides the iterative sampling process of a diffusion model towards a conditioning signal by mixing unconditional and conditional noise predictions using linear extrapolation. N_{cond} is the noise prediction from the conditional model, N_{uncond} is the noise prediction from the unconditional model, N_{pred} is the final predicted noise that will be removed using a diffusion sampler and g_i is the scheduled guidance scale at the i'th step during inference.

$$N_{pred}(i) = N_{uncond} + g_i \cdot (N_{cond} - N_{uncond}) \tag{1}$$

A negative CFG scale would result in the model working to generate what it believes is the opposite of the representation of the prompt. A higher CFG scale results in better quality outputs, but only to a certain limit beyond which the creativity of the model is stunted and the output begins to develop problems like overexposure, grain, etc. On the other hand, a lower CFG scale allows the model to exercise more creativity in the generation process, but at the price of accuracy of representation. Through trial and error, a constant CFG scale belonging to $[7, 12]$ was found to generate optimal images in most cases and is used by most models today.

Models utilizing classifier-free guidance scale do so by applying the same scale at each step of the inference process, such that plotting CFG scale against the inference step would result in a line with zero slope. The following sections outline various methods of scheduling the CFG scale along different functions.

III. Methodology

Dynamic Classifier-Free Guidance

The aim was to observe the effect of varying the CFG scale on the output, or in other words how the varying of the level of creativity the model uses, during the inference process affects the generated image. Using the below scheduling equations, various images are generated to observe the effects dynamic CFG scale would have on the output.

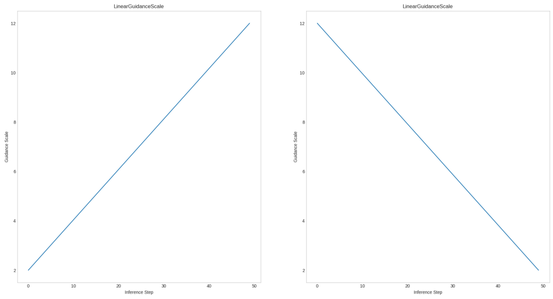

Fig. 2. Linear Guidance Scale Scheedules

Traditionally, for a fixed seed and parameters, the generated images using CFG are always the same. That is, for a given set of initial conditions the outputs are the same. However, the latent space representation of diffusion models is very rich and can be explored further using dynamically scheduled guidance resulting in several different high quality generations from the same initial conditions.

The equations mentioned below are used to schedule the guidance. The start and final values of the CFG scale are denoted by g_{start} and g_{final}. The total number of inference steps is N. g_i is the CFG scale values used at the inference step i.

A. Linear Scheduling Equation

In the linear scheduling equation, the classifier-free guidance scale is incremented or decremented by a constant value at each inference step, in other words, plotting the CFG scale versus the inference step number would result in a straight line. The values of guidance scale in the linear scheduling equation vary as described by Eq. (2).

$$g_i = g_{start} + \frac{i}{N-1} \cdot (g_{final} - g_{start}) \tag{2}$$

B. Cosine Scheduling Equation

The cosine scheduling equation models the CFG scale along a sinusoidal path through the inference process. The scheduler takes in parameters allowing the user to determine the period (T), amplitude, phase shift (ϕ) and vertical shift (δ) of the sine wave. Additionally, the user can add linear warm-up steps to the scheduler, which increments the CFG scale uniformly over a fixed number inference steps, between a start and end value.

This wave-based scheduler allows us to oscillate the model's creativity sinusoidally between any two values.

$$g_i = g_{start} + \frac{g_{final} - g_{start}}{2} \left(1 + \cos\left(2\pi T \left(\frac{i}{N-1} - \phi\right)\right)\right) + \delta \tag{3}$$

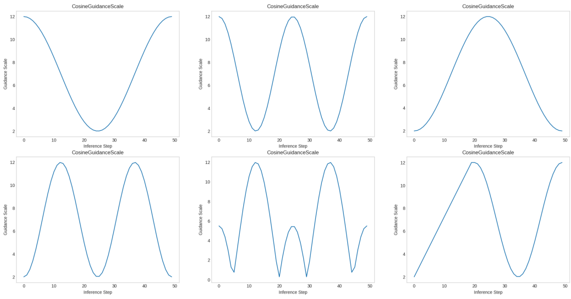

Fig. 3. Sinusoidal Guidance Scale Schedules

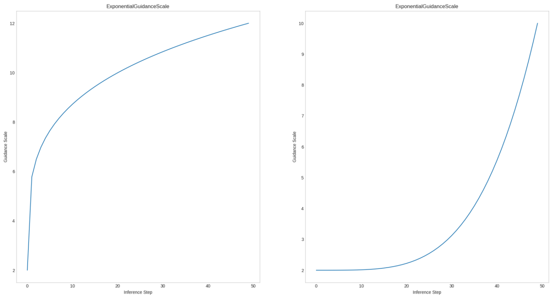

Fig. 4. Exponential Guidance Scale Schedules

C. Exponential Scheduling Equation

The exponential classifier-free guidance scale scheduler varies the CFG scale exponentially between each inference step, described by Eq. 4. This scheduler takes a single parameter, the rate of decay k_{decay}. A smaller value results in a slower decay.

$$g_i = g_{start} + (g_{final} - g_{start}) \cdot \left(\frac{i}{N-1}\right)^{\frac{1}{k_{\text{decay}}}} \tag{4}$$

D. Sawtooth Scheduling Equation

The sawtooth function is a non-sinusoidal periodic wave function that is characterized by a periodic linear increase over a fixed interval followed by a sudden drop. The inverted/reverse sawtooth wave follows an opposite periodic trend of linear decrease followed by a sudden rise. Both of these have been implemented by the Eq. 5. The scheduler allows the user to specify the phase shift (ϕ).

$$g_i = g_{start} + \frac{g_{final} - g_{start}}{2} \left(\pm \text{sawtooth}\left(2\pi\left(i - \phi\right)\right)\right) \tag{5}$$

IV. Results

Upon generating images with the various different dynamic CFG scale schedulers, many substantial changes for certain diffusion samplers (like DDIM, DDPM, etc.) and insignificant changes for certain samplers (such as DPMSolverMultistep) are found. Subjectively, some images look better than others, but to qualitatively decide which DCFG works best, comparisons have been performed using contrastive language-image pretraining (CLIP) scores, frechet inception distance (FID) and inception score (IS). A higher CLIP score means better prompt adherence, lower FID means more similarity between training data vs. generated samples, and a higher IS means better quality and diversity as more samples are correctly identifiable.

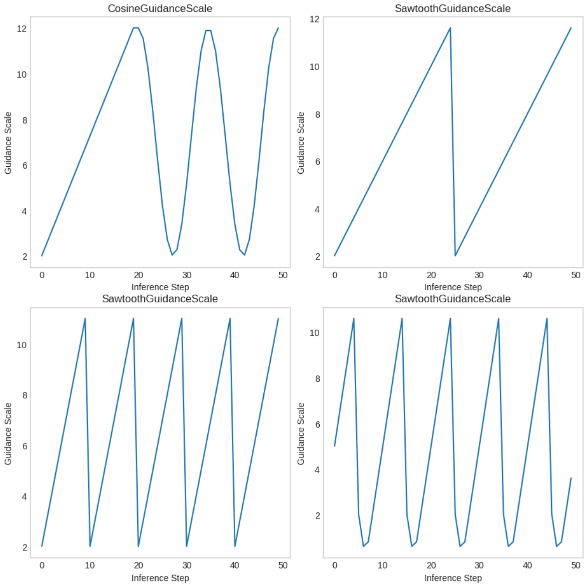

Fig. 5. Sawtooth Guidance Scale Schedules

Comparing the results tabulated in Table 1, it is seen that $Constant_3$ gives us the best CLIP score, $Exponential_5$ has the best FID score (lower is better) and $Cosine_4$ has the best IS score (higher is better). Certain schedules work well and achieve a great balance between prompt adherence and sample quality. From experimentation and subjective human preference analysis of the generated images, it is found that using exponential and cosine guidance scheduling tend to produce better looking generations overall.

V. Conclusion

In conclusion, various schedules for dynamically controlling the CFG scale during inference were explored. This allows for a better exploration of the rich latent space of diffusion models. Subjectively, after having generated thousands of images using various different dynamic schedules, the generated image quality is sometimes better or at least equivalent in comparison to the original constant scheduling, in terms of prompt alignment and diversity, when using certain dynamic schedules.

As more research is carried out into diffusion models, sampling techniques like CFG are being used in new ways (see latent consistency models, and sometimes not at all (see SDXL-Turbo). As new techniques for sampling are developed, it would be interesting to see the latent space representation of diffusion models exploited to its fullest extent in delivering high quality, high fidelity and diverse generations.

TABLE I
EVALUATION

DCFG Schedule	CLIP (↑)	FID (↓)	IS (↑)
$Constant_1$	29.74	17.32	128.4
$Constant_2$	29.68	17.96	130.7
$Constant_3$	**30.05**	16.56	132.3
$Constant_4$	**29.95**	18.01	131.1
$Linear_1$	29.04	16.59	137.1
$Linear_2$	29.97	16.93	135.9
$Exponential_1$	29.36	22.43	142.2
$Exponential_2$	29.22	21.30	143.9
$Exponential_3$	29.71	19.69	141.3
$Exponential_4$	29.34	19.44	141.7
$Exponential_5$	29.53	**15.09**	145.5
$Exponential_6$	29.23	**15.67**	145.3
$Exponential_7$	29.36	**16.02**	144.1
$Exponential_8$	29.24	16.28	143.6
$Cosine_1$	**29.99**	18.24	142.5
$Cosine_2$	29.62	18.65	140.3
$Cosine_3$	29.68	18.35	140.6
$Cosine_4$	29.58	18.91	**150.7**
$Cosine_5$	29.90	17.93	**146.6**
$Cosine_6$	29.73	17.89	143.0
$Cosine_7$	29.24	18.07	144.2
$Sawtooth_1$	29.90	19.47	**149.2**
$Sawtooth_2$	29.72	24.09	128.1
$Sawtooth_3$	29.68	19.03	133.9
$Sawtooth_4$	29.61	18.72	135.4

Note: The results have been obtained by running the CLIP, FID and IS metrics on images generated using 50 inference steps with the DDIM [17] Scheduler on the Tiny ImageNet-1k dataset. The three best scores in each metric are marked in bold.

VI. ACKNOWLEDGMENT

We express our sincere gratitude to Dr. Jayashree R. at PES University for her invaluable guidance. All our experiments were carried out on Google Colab free-tier, and we are grateful to Google for providing such a platform. We also appreciate the understanding and support from our families and friends. Their encouragement was pivotal throughout this academic journey.

REFERENCES

[1] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-Resolution Image Synthesis With Latent Diffusion Models," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2022, pp. 10684–10695.

[2] A. Vaswani et al., "Attention is All you Need," in Advances in Neural Information Processing Systems, Curran Associates, Inc., 2017.

[3] A. Ramesh et al., "Zero-Shot Text-to-Image Generation," in Proceedings of the 38th International Conference on Machine Learning, M. Meila and T. Zhang, Eds., in Proceedings of Machine Learning Research, vol. 139. PMLR, Jul. 2021, pp. 8821–8831.

[4] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical Text-Conditional Image Generation with CLIP Latents." arXiv, Apr. 12, 2022. doi: 10.48550/arXiv.2204.06125.

[5] Betker et al., "Improving Image Generation with Better Captions,"

[6] D. Podell et al., "SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis," 2023, doi: 10.48550/ARXIV.2307.01952.

[7] J. Oppenlaender, "The Creativity of Text-to-Image Generation," in Proceedings of the 25th International Academic Mindtrek Conference, in Academic Mindtrek '22. New York, NY, USA: Association for Computing Machinery, Nov. 2022, pp. 192–202. doi: 10.1145/3569219.3569352.

[8] U. Singer et al., "Make-A-Video: Text-to-Video Generation without Text-Video Data," in The Eleventh International Conference on Learning Representations, 2023.

[9] C. Saharia et al., "Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding," in Advances in Neural Information Processing Systems, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022.

[10] J. Ho and T. Salimans, "Classifier-Free Diffusion Guidance," in NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications, 2021.

[11] Heusel, Martin, et al. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, 2018.

[12] Salimans, Tim, et al. Improved Techniques for Training GANs, 2016.

[13] Agarwal, Sandhini, et al. Evaluating CLIP: Towards Characterization of Broader Capabilities and Downstream Implications, 2021.

[14] P. Dhariwal and A. Nichol, "Diffusion Models Beat GANs on Image Synthesis." arXiv, Jun. 01, 2021. doi: 10.48550/arXiv.2105.05233.

[15] T. Karras, M. Aittala, T. Aila, and S. Laine, "Elucidating the Design Space of Diffusion-Based Generative Models." 2022.

[16] W. Zhao, L. Bai, Y. Rao, J. Zhou, and J. Lu, "UniPC: A Unified Predictor-Corrector Framework for Fast Sampling of Diffusion Models." arXiv, Oct. 17, 2023. doi: 10.48550/arXiv.2302.04867.

[17] J. Song, C. Meng, and S. Ermon, "Denoising Diffusion Implicit Models." arXiv, Oct. 05, 2022. doi: 10.48550/arXiv.2010.02502.

[18] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models." arXiv, Dec. 16, 2020. doi: 10.48550/arXiv.2006.11239.

[19] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, "DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps." arXiv, Oct. 13, 2022. doi: 10.48550/arXiv.2206.00927.

[20] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, "DPM-Solver++: Fast Solver for Guided Sampling of Diffusion Probabilistic Models." arXiv, May 06, 2023. doi: 10.48550/arXiv.2211.01095.

VII. APPENDIX

A. Schedules

In the following subsection, explanations are provided for the different schedule parameters used in the DCFG schedules.

1) Constant Schedules: The constant schedule takes in a single parameter, $value$, which will remain constant through the inference process.

- $Constant_1$: 5.
- $Constant_2$: 7.5.
- $Constant_3$: 10.
- $Constant_4$: 12.5.

2) Linear Schedules: The linear schedule (see Eq. 2) takes in parameters in the order g_{start} and g_{final}

- $Linear_1$: (2, 12).
- $Linear_2$: (12, 2).

3) Exponential Schedules: The exponential schedule (see Eq. 4) takes in parameters in the order g_{start}, g_{end} and k_{decay}.

- $Exponential_1$: (2, 12, 4.0).
- $Exponential_2$: (2, 12, 2.0).
- $Exponential_3$: (5, 10, 4.0).
- $Exponential_4$: (5, 10, 2.0).
- $Exponential_5$: (2, 12, 0.25).
- $Exponential_6$: (2, 12, 0.5).
- $Exponential_7$: (5, 10, 0.25).
- $Exponential_8$: (5, 10, 0.5).

4) Cosine Schedules: The cosine schedule (see Eq. 3) takes in parameters in the order g_{start}, g_{end}, T (period), ϕ (phase shift), δ (vertical shift), abs (whether to make all values positive) and l_w (linear warm-up steps).

- $Cosine_1$: (2, 12, 1, 0, 0, True, 0).
- $Cosine_2$: (2, 12, 2, 0, 0, True, 0).
- $Cosine_3$: (2, 12, 1, 0, 0, True, 20).
- $Cosine_4$: (2, 12, 2, 0, 0, True, 20).
- $Cosine_5$: (-5.5, 12, 2, 0.25, 0, True, 0).
- $Cosine_6$: (2, 12, 2, 0.25, 0, True, 0).
- $Cosine_7$: (2, 12, 1, 0.5, 0, True, 0).

5) Sawtooth Schedules: The sawtooth schedule (see Eq. 5) takes in parameters in the order g_{start}, g_{end}, T (period), ϕ (phase shift), abs (whether to make all values positive) and $type$ (whether the sawtooth starts as a rising or falling).

- $Sawtooth_1$: (-2, 12, 5, 0.5, True, 'rising').
- $Sawtooth_2$: (2, 12, 5, 0, True, 'rising').
- $Sawtooth_3$: (2, 12, 2, 0, True, 'falling').
- $Sawtooth_4$: (2, 12, 2, 0, True, 'rising').

6) Graphs of various guidance schedules: The horizontal x-axis is the number of inference steps, and the vertical y-axis is the guidance values corresponding to a particular inference step.

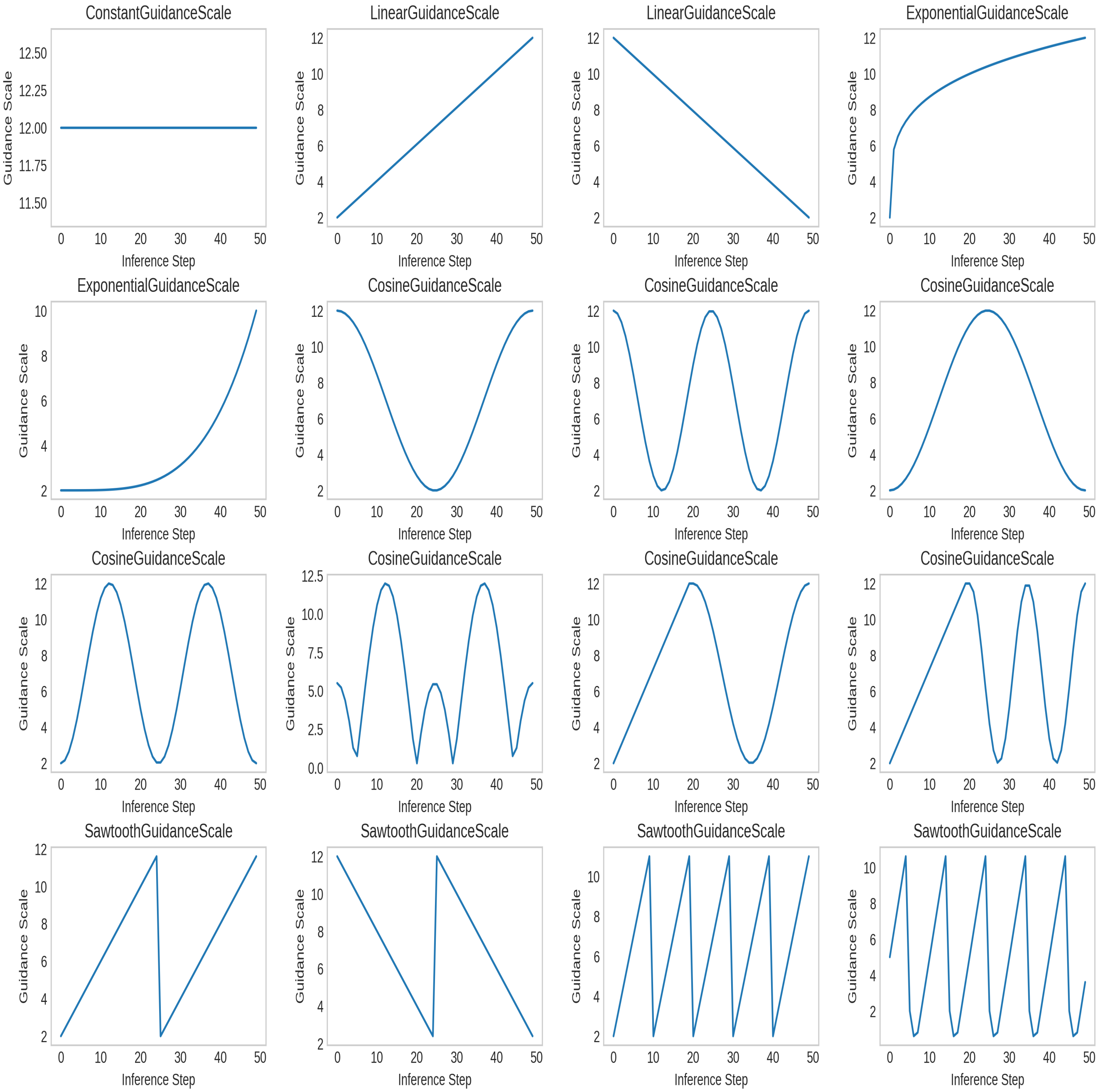

Fig. 6. Dynamic Guidance Scale Schedules