Drexel University

Department of Computer Science

Generative Modeling as Operator Design: From Diffusion to Attention

Candidacy Examination Report

Atieh Armin

Advisor: Dr. Ali Shokoufandeh

Abstract

This work reframes generative modeling through the lens of operator theory. We trace a progression from physically inspired Markov and Langevin dynamics to score-based diffusion and attention-based models, highlighting how operator design shapes core trade-offs between expressiveness, efficiency, and geometric fidelity. Diffusion models provide stable, local updates rooted in physical intuition but are computationally intensive; attention enables fast, global communication but lacks inductive biases such as locality, damping, and spectral structure. These tensions become especially pronounced in graph domains, where message-passing GNNs suffer from oversmoothing at depth, and Graph Transformers can become overly uniform without geometric constraints.

To address these limitations, we explore an operator-theoretic framework in which attention internalizes the structure of diffusion. As a case study, we examine a prototype mechanism for graph attention that incorporates physically inspired elements, which are diffusion-based propagation, harmonic spectral bias, and depth-wise damping, directly into the attention kernel. This design yields a globally expressive yet structurally grounded operator that respects graph geometry while maintaining stable and discriminative representations. This idea illustrates how embedding dynamics within attention can unify the strengths of diffusion and attention, pointing toward more interpretable, stable, and scalable generative architectures for structured data.

Contents

1 Introduction

Modern generative modeling has undergone a remarkable transformation, evolving from physically grounded approaches rooted in stochastic dynamics to highly expressive neural architectures like Transformers. While these models differ in formulation and application, a unifying lens emerges when we view them as operators. From this perspective, generation is not just a matter of sampling or inference, but of designing operators that shape how data is structured and traversed. The properties of these operators then play a central role in determining the model's capabilities, limitations, and alignment with the geometry of the underlying domain.

Classical methods such as Langevin dynamics view generation as a stochastic process shaped by physical principles: local gradients drive samples toward high-probability regions, while noise enables exploration. These methods are stable and geometry-aware, but often inefficient, requiring small steps and long trajectories. Score-based models generalize this idea, and instead of assuming access to a known energy landscape, they learn the score function (the gradient of log-density) directly from data. Combined with stochastic differential equations (SDEs), this leads to diffusion models that transform noise into data by simulating continuous-time dynamics, offering both theoretical grounding and impressive empirical results. However, they remain computationally demanding, as generation typically involves hundreds or thousands of small updates.

In contrast, Transformers replace slow local updates with powerful global attention operators, enabling flexible, few-step generation. Attention mechanisms are expressive, flexible, and highly parallelizable which makes them foundational in modern generative models. Yet they lack the inductive biases of physical systems leading to instability, oversmoothing, and difficulty scaling to structured domains.

The tension becomes sharper when we turn to graphs, where the irregular geometry and sparse connectivity demand operators that are both expressive and geometry-aware. Classical message-passing Graph Neural Networks (GNNs) implement a discrete diffusion: each layer averages features over neighbors. This is elegant and effective at shallow depth, but it invites over-smoothing as depth grows, homogenizing node states and erasing distinctions. Graph attention networks attempt to remedy this by attending neighbors selectively instead of averaging all of them equally. Despite this, they inherit attention's unconstrained nonlocality and can also over-smooth.

These observations lead us to ask how principled dynamics can be integrated into well-structured models such as Transformers, particularly on graphs where geometry and topology hold significant importance. Operator theory serves as the bridge between dynamics and representation. Instead of conceptualizing dynamics as an external sampling loop or representation as an unconstrained mixer, we can define the attention kernel as a physically informed operator. This operator adheres to graph geometry, appropriately filters spectra, and ensures controlled energy flow across depth.

In this manuscript, we begin by revisiting Markov chains and Langevin dynamics and the manner in which score-based models operationalize them. We then present Transformers, emphasizing how

conventional attention mechanisms lack inherent notions of locality, frequency selectivity, or damping. Building upon this foundation, we turn our focus into graphs, where these issues are exacerbated. We then walk through our hypothesis that these two worlds can be reconciled by embedding dynamics inside the architecture. Rather than treating diffusion and attention as opposing paradigms, we show that attention can internalize the structure of diffusion. As a case study, we explore a prototype graph attention mechanism that incorporates physically inspired principles directly into the attention kernel. This design illustrates how dynamic structure and global expressiveness can coexist within one operator, enabling stable, geometry-aware, and spectrally regularized representations on graphs.

2 Markov Chains and Langevin dynamics

A generative model is a system designed to produce realistic data by simulating the process that might have generated it. At its core, such a model contains a few key components. First, it maintains a collection of hypotheses or guesses, which are possible candidates for what the data could be. Second, it requires a mechanism (dynamics) for transitioning between hypotheses, allowing the model to move through its space of possibilities rather than staying fixed. Third, it needs rules for evaluating and improving these hypotheses, guiding the model to refine or adjust them over time. Finally, it includes a reasoning mechanism (deciding which guesses are more plausible) and procedures for sampling, allowing the model to explore a range of alternatives rather than collapsing to a single answer.

As the model evolves, these components must work together to refine hypotheses without destroying the structure of the data (its geometry, dependencies, and internal consistency) they represent. We call a process stable when each update preserves this structure while still making measurable improvement. The generative model must navigate a trade-off. It needs to refine its guesses without becoming unstable. That is, updates must be strong enough to make progress but gentle enough to preserve structure. Two common failures show up in practice: *stagnation* (updates are so tiny that nothing improves) and *instability* (updates overshoot and wash out structure). Designing effective update rules is therefore a central problem.

In this section, we revisit Markov chains and Langevin dynamics as foundational tools for defining such update mechanisms. These classical methods offer a rich framework for reasoning about transitions, stability, and convergence, and they help us analyze how well a generative model explores its hypothesis space over time.

2.1 Markov Chain and MCMC

From an abstract perspective, the dynamics of a generative model can be formalized using Markov chains. A Markov chain is a sequence of random variables where the next state depends only on the current one, not on the full history. It is governed by a transition kernel $P(x' \mid x)$, which defines the probability of moving from hypothesis x to x'. When used for generative modeling, the goal is to design P so that repeated applications of it produce samples from a desired distribution $\pi(x)$. In other words, we want the distribution of states in the Markov chain to converge to π over time. For

this to happen, π must be a *stationary distribution* of the chain, meaning that if the current state is distributed according to π, then the next state will be as well, and applying P does not change the distribution.

This leads to the Markov Chain Monte Carlo (MCMC) framework [3], in which the transition operator P is carefully constructed to leave π invariant. A common sufficient condition to ensure that π is stationary is *detailed balance*, which requires that

$$\pi(x)P(x' \mid x) = \pi(x')P(x \mid x'). \tag{1}$$

This symmetry ensures that π is a stationary distribution of the chain and that, under mild conditions, the chain will eventually sample from π regardless of its starting point [11]. This makes detailed balance a powerful design principle for constructing generative models that can reliably explore and sample from complex distributions.

From a computational perspective, the key question is: how quickly can the Markov chain produce useful, independent samples from the target distribution π? This is captured by the notion of *mixing time*—how fast the chain forgets its initial state and approaches equilibrium. The operator viewpoint gives us a powerful way to analyze this behavior. The transition kernel P, when viewed as an operator, reveals rich spectral structure that governs convergence. This perspective is part of what is broadly known as *operator theory*, and it has far-reaching consequences. At a basic level, the spectral gap of P (one minus the magnitude of its next-largest eigenvalue) controls the rate at which the chain decorrelates from its starting point. A larger spectral gap implies faster mixing and more efficient sampling [8]. These insights highlight that operator theory not only explains why certain update rules converge, but also allows us to quantify how well they perform. This connection between spectral structure and computational efficiency is central to both the analysis and design of modern generative models.

2.2 Langevin Dynamics

Traditional MCMC methods, such as the Random Walk Metropolis algorithm [10], propose new states by adding Gaussian noise to the current one. These proposals are simple but they do not account for the geometry or structure of the target distribution, often resulting in slow mixing, especially in high-dimensional settings.

To address this, we turn to a well-defined mathematically grounded concept: *Langevin dynamics*. Intuitively, when sampling from a distribution, we want to move more decisively toward regions of high probability and areas where the target density is large. Langevin dynamics provides a principled mechanism for doing this. It modifies random exploration by incorporating local gradient information, effectively steering the sampling process toward more likely regions. This approach helps the model reach equilibrium more efficiently by aligning its movement with the underlying structure of the distribution. Formally, Langevin dynamics is described by a stochastic differential

equation (SDE) that combines deterministic drift with stochastic diffusion:

$$dX_t = -\frac{1}{\gamma}\nabla U(X_t)\,dt + \sqrt{2D}\,dW_t,$$

where $U(x)$ is the potential function (typically the negative log of the target density), γ is a friction coefficient, D controls the diffusion strength, and W_t is standard Brownian motion. This equation models the time evolution of a particle in a potential field. The drift term represents velocity-like behavior, pulling the state toward regions of lower potential (i.e., higher probability), while the diffusion term introduces random perturbations (due to the Brownian motion) that reflect thermal noise or uncertainty, which prevents collapse to a single mode and encourages continued exploration. While this process acts on individual hypotheses or states, we can also view it as evolving a probability distribution over those hypotheses, allowing us to reason about convergence and equilibrium at the distributional level.

A central structural property of Langevin dynamics is that it defines a *reversible diffusion process*. Once equilibrium is reached, the system behaves identically when run forward or backward in time. This reversibility is formalized by the *detailed balance condition* defined in equation (1), which Langevin dynamics satisfies with respect to its stationary distribution $\pi(x) \propto \exp(-U(x))$ [15]. Reversibility ensures that π is a stationary distribution and supports sharp theoretical results on convergence and ergodicity.

One such result is an exponential convergence guarantee. The convergence behavior of Langevin dynamics closely parallels the operator-theoretic view we developed for MCMC (in section 2.1). The evolution of the system is governed by a transition mechanism, which is represented by the generator L in Langevin dynamics, whose spectral properties dictate how quickly the process forgets its initial state. Under mild regularity conditions, the probability distribution converges exponentially to its stationary distribution π at a rate determined by the spectral gap of L [1].

While Langevin dynamics has desirable theoretical properties in continuous time, it cannot be simulated directly in practice. Instead, we must discretize the dynamics into finite steps in order to implement it computationally. Thus, we should transition from the continuous-time stochastic process to a discrete-time sampling algorithm. The most basic discretization scheme is the *Euler–Maruyama method* [7], a stochastic analogue of Euler's method for ordinary differential equations. Applied to the Langevin SDE, it yields the update rule:

$$x_{k+1} = x_k - \frac{\eta}{\gamma}\nabla U(x_k) + \sqrt{2D\eta}\cdot\xi_k,$$

where $\eta > 0$ is the step size, and $\xi_k \sim \mathcal{N}(0, I)$ is standard Gaussian noise. This update forms the basis of the *Unadjusted Langevin Algorithm* (ULA), which is simple and scalable but introduces *discretization bias*—the samples do not exactly follow the target distribution π. To correct this bias, the *Metropolis-Adjusted Langevin Algorithm* (MALA) [16] adds a Metropolis–Hastings acceptance

step, ensuring that the resulting samples are consistent with π even at finite step sizes. However, this correction comes at a computational cost. Large steps may be frequently rejected, while small steps slow down exploration. This trade-off becomes especially pronounced in high dimensions, making step-size tuning critical for balancing accuracy and efficiency.

Another fundamental limitation lies in the fact that Langevin dynamics assumes we have prior access to the target distribution $\pi(x)$, or at least to its log-density $-\log \pi(x)$ and gradient $\nabla \log \pi(x)$. In traditional settings such as Bayesian inference or physical modeling, this quantity may be explicitly defined. But in modern generative modeling tasks, the distribution is often only specified implicitly via data. In particular, when $\pi(x)$ is not available in closed form these gradients cannot be computed directly. This raises a natural question: *can we learn these gradients from data? Can we model or estimate the score function* $\nabla \log \pi(x)$ *directly, and then use it to simulate Langevin-like dynamics?* This shift in perspective, from sampling with a known distribution to learning the score itself, leads us to the framework of *score-based generative models*.

3 The Evolution of Score-Based and Diffusion Models

In the previous section, we focused on classical sampling with physically motivated dynamics. These methods provide strong guarantees when the target density is known up to a normalizing constant. However, they rely on fixed, pre-defined transitions that are inefficient in high-dimensional or multimodal settings. Langevin partially addresses this by using gradients of the energy $U(x)$, yet still assumes full knowledge of the density and does not adapt to observed data. Score-based generative models address these limits by learning the *score function*—the gradient of the log-density—directly from data, enabling sampling from complex, implicitly defined distributions without an explicit likelihood or potential. This shift replaces pre-specified update rules with data-driven ones. The remainder of this section traces the conceptual and practical development of score-based and diffusion-based generative models, showing how they emerged from classical ideas, overcame key challenges, and continue to evolve toward more scalable, adaptive, and expressive sampling frameworks.

Score-based generative models build on the idea that one can learn to sample from complex distributions by directly estimating their *score function*, defined as the gradient of the log-density: $s(x) = \nabla_x \log p(x)$. As the logarithm of probability represents information content, where $-\log p(x)$ indicates the number of bits required to describe the event x, the score can be viewed as a local *information gradient*. Thus, the gradient points towards the direction where the description length decreases most rapidly, or in other words, where probability increases. In this sense, following the score moves a sample from uncertainty to certainty, locally reducing entropy. Crucially, because gradients of the log-density do not depend on the normalization constant, the score function is invariant to unnormalized scalings of the density.

Hyvärinen [5] formalized this with the *score matching* objective, which minimizes the squared difference between the model score and the data score. Although this avoids computing the normalizing constant, it still requires second-order derivatives and access to the true data score.

Vincent [26] addressed this by introducing *denoising score matching*, which trains a model to recover clean data from Gaussian-corrupted samples. This implicitly teaches the model the score of the smoothed distribution and provides a practical, scalable method for learning score functions from data. From a geometric perspective, denoising score matching trains the model to point toward regions of higher density in the data manifold, learning the direction of increasing likelihood without ever explicitly modeling the likelihood itself.

With the ability to learn score functions efficiently, the next challenge is using them to generate realistic samples. Instead of requiring an explicit density function, score-based models define a stochastic process that starts from pure noise and moves step by step toward regions of higher probability, guided entirely by the learned score field. This idea builds on classical Langevin dynamics but replaces the hand-crafted drift with a learned function that adapts to data. Song and Ermon [20] demonstrated the power of this approach with Noise-Conditional Score Networks (NCSNs), which estimate the score not just for clean data, but for data corrupted by noise at varying levels. Training across this range makes the score well-defined even in low-density regions and allows the model to guide samples from a simple Gaussian prior back to the data distribution. Generation proceeds through annealed Langevin dynamics, which is a gradual, noise-level-aware sampling process that refines the sample step by step, using the appropriate score at each stage. This multi-scale setup improves stability and robustness but is not without limitations. While the process is theoretically capable of reaching all regions of the distribution, doing so may require extremely small steps and a prohibitive number of iterations. Moreover, early methods struggled to scale to high-resolution data, where architectures and noise schedules that worked for 32×32 images often failed at higher resolutions [21], with sampling diverging or requiring heavy hyperparameter tuning. The need for many iterative updates per sample also makes the process slow. These challenges have driven ongoing efforts to make score-based sampling more efficient, scalable, and reliable.

In parallel with the development of score-based models, researchers explored similar ideas from the perspective of diffusion processes. Ho et al. [4] introduced Denoising Diffusion Probabilistic Models (DDPMs), which use a fixed noising schedule and a neural network trained to predict the added noise. To speed up inference, Song et al. [19] introduced Denoising Diffusion Implicit Models (DDIMs), which replace the stochastic DDPM reverse process with a deterministic, non-Markovian update rule derived from the noise-prediction model, enabling substantially fewer sampling steps while approximately preserving DDPM's per-timestep marginals. This resulted in significantly faster generation while largely preserving quality, showing that diffusion-based sampling could be both practical and performant. While DDPMs and score-based models emerged from different starting points (one learning to denoise, the other learning gradients of the log-density) they were soon recognized as mathematically connected, both involving a forward corruption process and a learned reverse trajectory. However, both approaches in discrete time suffer from practical limitations. They rely on hand-designed noise schedules, require many iterative sampling steps, and lack a fully continuous formulation. To address these challenges, Song et al. [22] proposed a continuous-time

framework using stochastic differential equations (SDEs) to describe the noise process and its learned reversal. Instead of working with discrete steps, they modeled generation as simulating the reverse of a diffusion process driven by a learned time-dependent score function. These advancements led to the first generation of high-resolution images from score-based models. Yet, the original continuous and discrete models remained computationally expensive, often requiring hundreds or thousands of steps to generate a single sample.

The evolution of score-based models can also be understood through the same operator-theoretic lens we used for Langevin dynamics. These models can be seen as learning a family of operators that transform noise into data through gradual, score-driven updates. At training time, the model estimates the score function, and essentially learning a vector field that tells us how to locally push probability mass toward higher-density regions. At inference time, sampling is implemented by discretizing a reverse-time process iteratively applying updates that approximate the action of a differential operator over short time intervals to transport a sample from random noise to structured data. This means generation is no longer a one-shot transformation, but a long sequence of local moves guided by the learned score field. While this iterative structure enables stability and flexibility, it also imposes key limitations. Because each update is local and incremental, score-based samplers often struggle in low-density regions, where the score may be inaccurate, and find it difficult to make global, geometry-aware decisions. Moreover, as mentioned before, sampling is computationally expensive, typically requiring hundreds of iterative steps to generate a single output. These challenges raise another question: *can we design more global, expressive, and adaptive generative operators that compress long diffusion chains into fewer, more powerful transformations? Ones that can learn to act over the full geometry of the data, adapt their behavior to the current sample, and perform generation efficiently in both time and space.* This emerging direction points toward the next frontier in generative modeling.

4 Transformers and Attention Mechanisms

As discussed in the previous section, score-based diffusion models offer a powerful framework for generative modeling. Yet, they remain computationally intensive and inherently limited. To overcome these limitations, we seek a new class of generative operators that can perform nonlocal, data-adaptive transformations that are capable of mixing information across distant regions, adapting to the geometry of the sample, and executing meaningful transitions in just a few steps. The *attention mechanism*, originally introduced by Vaswani et al. [24], provides precisely this kind of content-aware computation. Rather than diffusing local signals over many steps, attention computes interactions between all positions in parallel, using learned relevance scores that adapt to the data. Each element in the input selects which others to focus on, dynamically aggregating information across the entire input space. This makes attention inherently adaptive, nonlocal, and globally aware, which are ideal traits. When layered with feedforward networks and residual connections, attention gives rise to the Transformer architecture [24], which has become foundational in modern deep learning across language, vision, and multimodal generation. Transformers effectively learn deep stacks

of global, attention-based transformations, enabling them to flexibly process structured inputs, model long-range dependencies, and offer powerful, efficient alternatives to diffusion-style generative processes.

Despite their success across domains, vanilla Transformers face key limitations when applied to data with inherent spatial, geometric, or physical structure. First, they lack spatial bias. The original attention mechanism treats all input positions as exchangeable and relies entirely on added positional encodings to provide a sense of order. There is no intrinsic notion of locality, distance, or adjacency, meaning that the model must learn spatial relationships from scratch. This can make learning inefficient and data-hungry, particularly in vision or scientific settings where nearby elements often share stronger correlations than distant ones. Second, vanilla attention is not inherently multiscale or frequency-aware. The dot-product similarity between query and key vectors is agnostic to the frequency content of features, and it does not distinguish between fine, high-frequency details and smooth, low-frequency trends. Consequently, Transformers lack the hierarchical or band-limited inductive biases that make convolutional architectures naturally sensitive to local edges and textures while preserving global coherence. Third, attention in its basic form allows unconstrained global interactions. Every token can, in principle, attend to every other with equal opportunity, regardless of spatial or semantic proximity. Without any built-in mechanism for range decay or locality, this unrestricted connectivity can lead to premature global mixing of information. In deep stacks, such indiscriminate aggregation can blur distinctions between tokens and reduce representation diversity, somewhat analogous to oversmoothing in graph neural networks. Together, these characteristics make vanilla attention powerful yet structurally ungrounded.

In response to these structural limitations of standard attention researchers have introduced a range of architectural enhancements. One major line of work modifies positional encodings to better reflect geometric relationships. Instead of relying on fixed absolute encodings as in the original Transformer [24], relative positional embeddings [18] allow the model to represent how far apart two tokens are, which effectively biases attention toward nearby tokens and allows the model to learn spatial adjacency priors. Rotary positional embeddings (RoPE) [23] further enhance this idea by introducing relative position dependence directly into the attention computation via complex rotations, enabling the model to implicitly capture relative distances.

Another set of improvements targets locality and frequency awareness. Local attention schemes, such as in the Swin Transformer [9], restrict attention to fixed-size non-overlapping windows and build global context hierarchically, improving inductive bias and efficiency in structured domains like images. At the same time, frequency-aware models replace or augment attention with spectral components that operate in the Fourier domain. GFNet [14] replaces the self-attention sub-layer with a global filter layer that mixes tokens via Fourier Transform and element-wise multiplication with learnable frequency-domain filters. The learned filters can emphasize different frequency bands (the final layer often focuses on low frequencies). On the other hand, Spectformer [13] mixes spectral (frequency-domain) token-mixing layers with standard multi-head self-attention to model long-range

coupling and global smoothness. These advancements ground attention-based models in the spatial and spectral structure of real-world data, making them more efficient and effective across domains.

While architectural variants like positional encodings and windowed attention introduce valuable structure, they largely act as external augmentations without altering the internal behavior of the attention operator itself. As a result, core attention mechanisms still permit pathological behaviors: tokens can attend uniformly or excessively to others within the allowed set, and nothing inherently enforces frequency awareness, smoothness, operator-norm boundedness, or energy decay. In deep networks, such instabilities can accumulate, leading to noise amplification, global propagation of discontinuities, or collapse of internal representations. Recognizing this, we have turned to operator theory and physical dynamics to redesign attention itself, and embedding principles like locality, smoothness, and energy control directly into the attention mechanism. Rather than relying on architectural patches, these approaches aim to build attention layers that are inherently stable, structured, and geometry-aware, making them more suitable for deep, complex generative models.

5 From GNNs to Graph Transformers

In the previous sections, we saw how generative models could be framed as operators. We now turn to graphs, a domain where these tradeoffs become especially acute. Graphs arise across scientific, relational, and physical domains, representing data with irregular geometry and sparse, discrete structure. Unlike sequences or images, graphs lack a natural coordinate system, making operator design more challenging. In this section, we revisit both Graph Neural Networks (GNNs) and Graph Transformers, examining their strengths, limitations, and how they handle structure, depth, and expressiveness in graph domains.

Graph Neural Networks (GNNs) are a broad class of models designed to process data defined on graphs. These models aim to learn representations that capture both the features of individual nodes and the topology of the graph. The core idea behind most GNNs is *neural message passing*, where each node iteratively aggregates information from its neighbors to update its own representation. Early models such as Graph Convolutional Networks (GCNs) [6] demonstrated the effectiveness of this approach for tasks like semi-supervised node classification. The common structure across these models is a form of discrete diffusion where each layer performs a localized smoothing operation, gradually mixing node features across the graph. While this is beneficial for capturing local dependencies, it also introduces a fundamental limitation, known as *oversmoothing*. As the number of layers increases, node representations tend to become indistinguishable from one another, eventually collapsing toward a constant vector across the graph. This phenomenon has been studied both empirically and theoretically. For instance, Oono and Suzuki [12] showed that for typical GNN architectures, expressive power decays exponentially with depth due to the repeated averaging of neighboring features. As a result, standard GNNs struggle to scale to deeper architectures, since deeper layers lead to loss of discriminability and degraded performance on tasks requiring fine-grained node-level representations.

Graph Attention Networks (GATs) introduce attention mechanisms into graph neural networks by allowing each node to attend to its neighbors with learned, data-dependent weights instead of aggregating them uniformly [25]. This yields a flexible message-passing scheme in which the model can emphasize the most relevant neighbors while down-weighting less informative ones. In practice, GAT have shown strong performance on node and graph classification benchmarks, particularly when sufficient labeled data is available. One might hope that such learned, non-uniform aggregation helps mitigate oversmoothing, since layers can selectively preserve important distinctions rather than averaging all neighbors equally. However, recent theoretical work suggests otherwise. Even attention-based GNNs can suffer from exponential decay in expressiveness with depth under certain conditions [27]. Without structural biases or regularization, attention-based aggregation alone does not guarantee discriminability, highlighting the need for better operator design even in attention-based graph models.

The limitations observed in both traditional GNNs and Graph Transformers point to a deeper issue: the lack of physically grounded dynamics in the design of their message-passing operators. Whether through oversmoothing in GNNs or unstructured aggregation in Transformers, both architectures struggle to maintain rich, discriminative node representations as depth increases. What's missing is a principled mechanism for regulating how information flows over the graph, one that accounts for inertia, dissipation, and structural constraints. From this perspective, introducing ideas from physical systems offers a promising way forward. We can design graph operators that integrate local propagation with mechanisms for stability and long-range interaction. Recent works have begun to explore this direction: GraphCON [17] models message passing as a system of coupled oscillators with inertial terms, while models like GRAND [2] incorporate diffusion dynamics to guide learning. These approaches suggest that embedding physical priors into graph architectures can help reconcile expressiveness and stability, providing a foundation for deeper, more reliable generative models on graphs.

6 Bridging Dynamics and Attention

Having walked through the landscapes of Markov chains, Langevin dynamics, score-based diffusion, and attention-based models, we now return to a central question that has quietly threaded through each of these developments: how should we design operators that are expressive, geometry-aware, and stable at scale?

Our guiding hypothesis is that diffusion and attention are not opposing paradigms, but can be unified into a single operator framework. As a case study, we explore how attention can be reinterpreted as a dynamic process. Rather than treating attention and dynamics as separate modules, we embed physically inspired principles directly into the attention kernel, turning it into a structured operator that fuses the interpretability of diffusion with the expressiveness of Transformers. In this formulation, the attention kernel will be a diffusion-regularized spectral operator. This kernel interpolates between local smoothing and global coupling, while an explicit damping mechanism suppresses high-frequency noise and mitigates the pathological behaviors seen in both deep GNNs and Transformers. Unlike

classical GNNs and graph Transformers this operator defines a middle ground which is a geometry-aware message-passing operator embedded within the attention mechanism itself. Damping ensures that attention scores remain bounded across layers, preventing amplification or collapse, while the harmonic component promotes spectral coherence and stabilizes long-range interactions. The resulting attention mechanism can be analyzed as an operator, which is empirically stable, its spectrum is controlled to avoid signal explosion or vanishing, and it maintains discriminability across depth without requiring external reweighting tricks.

What distinguishes our operator from prior spectral or physics-inspired architectures is its emphasis on internal regulation. While many existing approaches wrap attention in structural components—adding Fourier layers, positional encodings, or external filters to stabilize behavior—we embed structure directly into the attention kernel itself. Here, the weights are not just learned but shaped by a diffusion prior, making the attention mechanism an operator that naturally reflects the geometry of the graph and the dynamics of information propagation. This enables attention maps that are globally expressive yet locally grounded, promoting stable and meaningful message passing. Importantly, this is not a fixed architecture but a conceptual framework defined by three core principles: diffusion-inspired propagation, which anchors interactions in neighborhood structure; a low frequency spectral bias, which enhances coherence and suppresses noise; and damping over depth, which prevents signal amplification and maintains expressiveness in deep networks. Together, these elements create a scaffold for attention that respects topology, stabilizes dynamics, and supports long-range dependencies.

7 What's Next?

The central premise of this work is to rethink attention-based architectures through the lens of operator design. This will open a wide landscape of future directions. Consider molecular systems. Atoms do not simply pass messages; they interact through forces, repel at short range, attract at intermediate distances, and settle into stable energy configurations. These interactions define the structure of molecular dynamics, and that dynamics can be embedded directly into the attention kernel. Rather than relying on learned weights to "discover" that distant atoms interact weakly or that certain motifs matter, we can design attention mechanisms that reflect known physical laws. Moreover, in settings where system behavior follows a particular physical model, attention constructed alongside the governing equation can naturally inherit the appropriate inductive biases. For example, in settings where behavior follows reaction–diffusion dynamics, attention shaped by the reaction–diffusion equation can naturally combine local propagation with transformation, allowing signals to diffuse across the graph where helpful, decay or amplify in place where needed, and form coherent global patterns from local interactions. Likewise, in systems governed by telegraph dynamics, where information propagates at finite speeds and is damped over time, attention shaped by the telegraph equation can enforce communication patterns that respect both spatial proximity and temporal structure.

What all these examples share is the belief that structured dynamics, whether domain-specific or

abstract, can be *internalized* within the attention mechanism. This leads to three core benefits. First, it provides a source of stability. Deep networks remain well-behaved because information spreads in a controlled, physically meaningful way. Second, it enforces geometry-awareness. Thus, who communicates with whom, and how strongly, is shaped by the structure of the domain—be it a molecule, a graph, or a spatial mesh. Third, it allows for meaningful selectivity. So, the model doesn't simply average features but learns which signals to preserve, which to dampen, and how to modulate influence over time and space.

In the long term, this perspective points toward a class of architectures that are expressive like Transformers, but grounded in the structure of the systems they model. If attention can carry dynamics, then this approach extends far beyond physics and chemistry. Any domain where influence has geometry, speed, or cost, such as social networks, communication graphs, epidemics, transportation systems, stands to benefit from attention operators that reflect how information should actually flow. Rather than asking attention to learn everything from scratch, we let it inherit the rules of the world it operates in.

A practical caveat to embedding physically inspired dynamics into attention is computational cost. Many of the most principled operators, especially those rooted in diffusion, require global computations, such as solving linear systems or computing eigendecompositions of the graph Laplacian. These operations scale poorly with graph size and can bottleneck training and inference. This tension opens a valuable path for fundamental research of developing fast, structure-preserving approximations that retain the stability and geometric fidelity of the underlying dynamics. One promising direction lies in Chebyshev polynomial approximations, which approximate spectral filters through a recursive sequence of sparse matrix-vector products. Folding such approximations into the attention kernel offers a path toward scalable, stable, and physically grounded architectures, and invites deeper exploration into the intersection of numerical methods and neural design.

8 Conclusion

In this work, we explored a common thread running through many modern generative models, which is how information moves and changes over time. From the early ideas of Markov chains and Langevin dynamics, to score-based diffusion models, to the global mixing of Transformers, each method can be seen as using a kind of operator.

Diffusion models are grounded in physical ideas and give stable, local updates that respect the shape and structure of data, but they can be slow and require many steps. Transformers are fast and powerful, using attention to let every part of the input talk to every other part. However, they often ignore important structure, like distance or geometry. Graph neural networks sit somewhere in the middle, but they too face problems when scaled to deeper architectures, especially as they tend to smooth out important differences between nodes.

The main idea we've put forward is that attention doesn't have to be separate from dynamics. It can actually host them. Instead of layering dynamics and attention on top of one another, we can design

attention to behave like a dynamic system itself, one that is stable, aware of structure, and tuned to the signals that matter. This leads to models that are both expressive and grounded, capable of handling complex data in a principled way.

More broadly, this perspective invites us to think differently about model design. If attention can carry the dynamics of the system we care about—whether it's molecules, waves, or information networks—then we can build models that better reflect the problems they're solving. Not by adding more layers or tricks, but by shaping the core operators that define how information flows.

References

[1] D. Bakry, I. Gentil, and M. Ledoux. *Analysis and geometry of Markov diffusion operators*. Vol. 348. Springer Science & Business Media, 2013.

[2] B. Chamberlain, J. Rowbottom, M. I. Gorinova, M. Bronstein, S. Webb, and E. Rossi. "Grand: Graph neural diffusion". In: *International conference on machine learning*. PMLR. 2021, pp. 1407–1418.

[3] W. K. Hastings. "Monte Carlo sampling methods using Markov chains and their applications". In: (1970).

[4] J. Ho, A. Jain, and P. Abbeel. "Denoising diffusion probabilistic models". In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[5] A. Hyvärinen and P. Dayan. "Estimation of non-normalized statistical models by score matching." In: *Journal of Machine Learning Research* 6.4 (2005).

[6] T. N. Kipf and M. Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *International Conference on Learning Representations*. 2017. URL: https://openreview.net/forum?id=SJU4ayYgl.

[7] P. E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Vol. 23. Applications of Mathematics. Springer, 1992.

[8] D. A. Levin and Y. Peres. *Markov chains and mixing times*. Vol. 107. American Mathematical Soc., 2017.

[9] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. "Swin transformer: Hierarchical vision transformer using shifted windows". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 10012–10022.

[10] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. "Equation of state calculations by fast computing machines". In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092.

[11] S. P. Meyn and R. L. Tweedie. *Markov chains and stochastic stability*. Springer Science & Business Media, 2012.

[12] K. Oono and T. Suzuki. "Graph Neural Networks Exponentially Lose Expressive Power for Node Classification". In: *International Conference on Learning Representations*. 2020. URL: https://openreview.net/forum?id=S1ldO2EFPr.

[13] B. N. Patro, V. P. Namboodiri, and V. S. Agneeswaran. "Spectformer: Frequency and attention is what you need in a vision transformer". In: *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2025, pp. 9543–9554.

[14] Y. Rao, W. Zhao, Z. Zhu, J. Lu, and J. Zhou. "Global filter networks for image classification". In: *Advances in neural information processing systems* 34 (2021), pp. 980–993.

[15] H. Risken. *The Fokker-Planck equation: methods of solution and applications*. Springer Science & Business Media, 1989.

[16] G. O. Roberts and R. L. Tweedie. "Exponential convergence of Langevin distributions and their discrete approximations". In: *Bernoulli* 2.4 (1996), pp. 341–363. DOI: 10.2307/3318418.

[17] T. K. Rusch, B. Chamberlain, J. Rowbottom, S. Mishra, and M. Bronstein. "Graph-coupled oscillator networks". In: *International Conference on Machine Learning*. PMLR. 2022, pp. 18888–18909.

[18] P. Shaw, J. Uszkoreit, and A. Vaswani. "Self-attention with relative position representations". In: *arXiv preprint arXiv:1803.02155* (2018).

[19] J. Song, C. Meng, and S. Ermon. "Denoising diffusion implicit models". In: *arXiv preprint arXiv:2010.02502* (2020).

[20] Y. Song and S. Ermon. "Generative modeling by estimating gradients of the data distribution". In: *Advances in neural information processing systems* 32 (2019).

[21] Y. Song and S. Ermon. "Improved techniques for training score-based generative models". In: *Advances in neural information processing systems* 33 (2020), pp. 12438–12448.

[22] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. "Score-based generative modeling through stochastic differential equations". In: *arXiv preprint arXiv:2011.13456* (2020).

[23] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. "Roformer: Enhanced transformer with rotary position embedding". In: *Neurocomputing* 568 (2024), p. 127063.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[25] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. "Graph attention networks". In: *arXiv preprint arXiv:1710.10903* (2017).

[26] P. Vincent. "A connection between score matching and denoising autoencoders". In: *Neural computation* 23.7 (2011), pp. 1661–1674.

[27] X. Wu, A. Ajorlou, Z. Wu, and A. Jadbabaie. "Demystifying oversmoothing in attention-based graph neural networks". In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 35084–35106.