
Supplement

Anonymous Author(s)

Affiliation

Address

email

1 Introduction

2 This supplementary document provides extended explanations and additional results that support the
3 claims presented in the main paper. The content is organized as follows.

- 4 1. A comprehensive related work on fractional calculus is detailed in Section A.
- 5 2. Preliminaries on fractional calculus are presented in Section B.
- 6 3. Fractional Operators for $\alpha > 0$ are discussed in Section C.
- 7 4. All theoretical proofs in this paper are presented in Section D.
- 8 5. Nonsingular scenario for the kernel is shown in Section E.
- 9 6. KatFDE examples are presented in Section F.
- 10 7. Graph differential equation models are given in Section G.
- 11 8. Datasets and more experiments on the graph are available in Section H.
- 12 9. Datasets and more experiments for traffic forecasting are discussed in Section I.

13 A Related Work

14 Our work builds on research in fractional differential equations, neural integer-order and fractional-
15 order ODE models, and neural network attention mechanisms. We will present the related work from
16 the following aspects.

17 A.1 Fractional Differential Equations

18 Fractional Differential Equations (FDEs) generalize classical differential equations by allowing the
19 order of differentiation to be a non-integer, thereby providing a powerful framework for modeling
20 systems with memory and hereditary properties. The mathematical foundations of FDEs have been
21 rigorously studied, with seminal contributions from [1–3]. Among the most prominent formulations
22 are the Riemann–Liouville and Caputo derivatives [4], both of which use power-law kernels to encode
23 memory effects. However, such power-law kernels often impose restrictions when modeling systems
24 with heterogeneous or scale-dependent dynamics. To address these limitations, alternative definitions
25 have been proposed, such as the Caputo–Fabrizio derivative with an exponential decay kernel [2],
26 and the Atangana–Baleanu derivative with a Mittag-Leffler-based kernel [5]. These generalizations
27 retain the core non-local structure of fractional calculus while enhancing modeling flexibility. More
28 recently, variable-order fractional derivatives have attracted considerable attention due to their ability
29 to reflect more flexible and complex dynamic memory mechanism in real-world phenomena [6, 7]. In
30 these systems, the fractional order α is time-dependent, denoted as $\alpha(t)$, enabling a more precise
31 representation of evolving dynamic behaviors.

32 In addition to the rich theoretical results, FDEs have found broad applicability across a wide range of
33 fields. For instance, the authors offered a foundational overview of its practical uses in areas such as

signal processing, system modeling and automatic control [8]. The authors showed that fractional calculus has emerged as a powerful mathematical framework for modeling complex systems in traffic forecasting [9]. The authors highlighted its role in viscoelasticity, illustrating how fractional-order models effectively capture the memory and hereditary characteristics inherent in such materials [10]. Meanwhile, FDEs have also been widely utilized to improve the performance of graph neural networks [11–13]. Despite considerable progress in both theory and applications, the design of kernel functions for fractional calculus remains largely underexplored. Particularly in neural differential equations, adaptive kernel functions can learn to assign appropriate weights to historical states based on relevance, which is still in infancy.

A.2 Neural Differential Equations

Neural Differential Equations (NDEs) offer a unified framework that integrates neural networks with differential equations to model continuous-time dynamic systems, bridging the gap between deep learning and classical dynamical systems. Among the various types of NDEs, Neural Ordinary Differential Equations (NODEs) and Neural Fractional Differential Equations (NFDEs) are more closely related to our work. Introduced by [14], NODEs model the evolution of hidden states by parameterizing the right-hand side of an ODE using neural networks. This model allows for adaptive computation in continuous time, offering memory efficiency and interpretability. To enhance the expressiveness of NODEs, several variants have been proposed. Augmented NODEs [15] expand the hidden state space with additional dimensions to improve the representational power and alleviate topological constraints that limit standard NODEs. Neural Controlled Differential Equations (NCDEs) [16] further generalize the NODEs framework to handle controlled systems in modeling irregular time series data, such as in finance or healthcare. Graph NODEs [17] have demonstrated strong performance on graph-structured data by integrating graph convolutional operations with continuous dynamics. Spatio-temporal graph NCDE [18] achieves significant performance improvements in traffic forecasting by integrating two NCDEs for temporal and spatial processing.

Despite their flexibility, NODEs-based models are limited by integer-order calculus, restricting their ability to capture memory and long-range dependencies, thus motivating interest in fractional-order extensions. Inspired by Graph NODEs, the FROND framework introduces a generalized fractional-order continuous GNN model using Caputo derivatives to capture non-local, memory-dependent dynamics, offering improved performance and mitigating oversmoothing in graph learning tasks [12]. Then, the DRAGON framework is proposed, which shows a distributed-order fractional continuous GNN that learns a superposition of derivative orders, enabling flexible and non-Markovian feature updating dynamics [19]. Recently, the NvoFDE framework introduces variable-order fractional differential operators into neural networks, enabling learnable and adaptive derivative orders based on time and hidden features [13]. However, most existing fractional neural models use fixed kernels with predefined weights assigned to historical states, limiting their flexibility and adaptability. Our work addresses this gap by introducing adaptive kernel functions into the fractional differential equation framework for improved temporal representation.

A.3 Attention Mechanisms in Neural Networks

Attention mechanisms have emerged as a fundamental component of modern deep learning architectures, enabling models to dynamically prioritize informative parts of the input. The transformer architecture [20] introduces self-attention, which computes pairwise interactions between elements in a sequence, allowing for efficient modeling of long-range dependencies. This innovation has had transformative impacts across a variety of domains, including natural language processing, computer vision and time-series forecasting [21, 22]. In continuous-time systems, attention mechanisms have been integrated into neural differential equations to enhance the representational power. For example, Continuous Self-Attention Neural ODEs [23] extend Neural ODEs framework by integrating a lightweight self-attention mechanism, resulting in more flexible and interpretable dynamics. Similarly, attention-based Neural ODEs have been employed in spatio-temporal prediction tasks [24].

In Graph Neural Networks (GNNs), attention mechanisms have unlocked unprecedented flexibility in neighbor weighting and hierarchical feature propagation. Graph Attention Networks (GATs) [25] use self-attention to assign adaptive weights to neighboring nodes during aggregation, improving performance in scenarios with heterogeneous node importance. Extensions of GATs, such as multi-head and hierarchical attention models [26, 27], further enhance the model’s ability to capture

structural nuances. Despite these advances, the integration of attention with fractional-order models remains largely underexplored. Fractional calculus, known for its inherent memory and non-local properties, offers a natural framework to capture long-range dependencies. Combining it with attention mechanisms could lead to a new class of flexible and adaptive neural operators.

B Preliminaries on Fractional Calculus

This section offers additional material on fractional calculus theory, with key details presented in the main text of Section 2. Different from [28], we present results in terms of ψ -fractional derivative that is quite general than previous work ($\psi(t) = t$). For more detailed information, please refer to [29, 30]. We begin with the basic definitions.

Definition 1 (ψ -Caputo Fractional Derivative). *Let $\alpha > 0$, $n \in \mathbb{N}$, and I be an interval such that $-\infty \leq a < b \leq \infty$. Let $f, \psi \in C^n(I)$ be two functions such that ψ is increasing and $\psi'(x) \neq 0$ for all $x \in I$. The left ψ -Caputo fractional derivative of f of order α is defined by*

$${}^C D_{a+}^{\alpha, \psi} f(x) := \frac{1}{\Gamma(n - \alpha)} \int_a^x \psi'(t) (\psi(x) - \psi(t))^{n - \alpha - 1} f_{\psi}^{[n]}(t) dt,$$

and the right ψ -Caputo fractional derivative is given by

$${}^C D_{b-}^{\alpha, \psi} f(x) := \frac{(-1)^n}{\Gamma(n - \alpha)} \int_x^b \psi'(t) (\psi(t) - \psi(x))^{n - \alpha - 1} f_{\psi}^{[n]}(t) dt,$$

where $f_{\psi}^{[n]}(t) := \left(\frac{1}{\psi'(t)} \frac{d}{dt} \right)^n f(t)$, $n = [\alpha] + 1$ if $\alpha \notin \mathbb{N}$, and $n = \alpha$ if $\alpha \in \mathbb{N}$.

For $\alpha \in (0, 1)$, the left and right ψ -Caputo fractional derivatives reduce to

$${}^C D_{a+}^{\alpha, \psi} f(x) = \frac{1}{\Gamma(1 - \alpha)} \int_a^x (\psi(x) - \psi(t))^{-\alpha} f'(t) dt,$$

and

$${}^C D_{b-}^{\alpha, \psi} f(x) = \frac{-1}{\Gamma(1 - \alpha)} \int_x^b (\psi(x) - \psi(t))^{-\alpha} f'(t) dt,$$

respectively. For specific choices of the function ψ , the ψ -Caputo fractional derivative reduces to several well-known operators [31]. Throughout this work, we focus on the left-sided fractional derivative. The corresponding results for the right-sided derivative can be obtained analogously with appropriate modifications.

To get some intuition, we provide a specific example below.

Lemma 1. *Given $\beta \in \mathbb{R}$ with $\beta > n$, consider the following function:*

$$f(x) = (\psi(x) - \psi(a))^{\beta - 1}, \quad g(x) = (\psi(b) - \psi(x))^{\beta - 1}.$$

For $\alpha > 0$, we have:

$$\begin{aligned} {}^C D_{a+}^{\alpha, \psi} f(x) &= \frac{\Gamma(\beta)}{\Gamma(\beta - \alpha)} (\psi(x) - \psi(a))^{\beta - \alpha - 1}, \\ {}^C D_{b-}^{\alpha, \psi} g(x) &= \frac{\Gamma(\beta)}{\Gamma(\beta - \alpha)} (\psi(b) - \psi(x))^{\beta - \alpha - 1}. \end{aligned}$$

Now we present the relation between fractional order derivative and integer order counterparts. This can be readily seen from the following theorem that is derived mainly using integration by parts.

Theorem B.1. *Suppose that $f, \psi \in C^{n+1}[a, b]$. Then, for all $\alpha > 0$,*

$${}^C D_{a+}^{\alpha, \psi} f(x) = \frac{(\psi(x) - \psi(a))^{n - \alpha}}{\Gamma(n + 1 - \alpha)} f_{\psi}^{[n]}(a) + \frac{1}{\Gamma(n + 1 - \alpha)} \int_a^x (\psi(x) - \psi(t))^{n - \alpha} \frac{d}{dt} f_{\psi}^{[n]}(t) dt,$$

and

$$\begin{aligned} {}^C D_{b-}^{\alpha, \psi} f(x) &= (-1)^n \frac{(\psi(b) - \psi(x))^{n - \alpha}}{\Gamma(n + 1 - \alpha)} f_{\psi}^{[n]}(b) \\ &\quad - \frac{1}{\Gamma(n + 1 - \alpha)} \int_x^b (\psi(t) - \psi(x))^{n - \alpha} (-1)^n \frac{d}{dt} f_{\psi}^{[n]}(t) dt. \end{aligned}$$

From this theorem, it is found that

$$\lim_{\alpha \rightarrow n^-} {}^C D_{a+}^{\alpha, \psi} f(x) = f_{\psi}^{[n]}(t).$$

We next present the relation between integration and differentiation of ψ -Caputo fractional function that is vital for the equivalent transformation between differential form and its integral form.

Theorem B.2. *Given a function $f \in C^n[a, b]$ and $\alpha > 0$, we have:*

$$I_{a+}^{\alpha, \psi} \left({}^C D_{a+}^{\alpha, \psi} f(x) \right) = f(x) - \sum_{k=0}^{n-1} \frac{f_{\psi}^{[k]}(a)}{k!} (\psi(x) - \psi(a))^k,$$

$$I_{b-}^{\alpha, \psi} \left({}^C D_{b-}^{\alpha, \psi} f(x) \right) = f(x) - \sum_{k=0}^{n-1} \frac{(-1)^k f_{\psi}^{[k]}(b)}{k!} (\psi(b) - \psi(x))^k.$$

Theorem B.3. *Given a function $f \in C^1[a, b]$ and $\alpha > 0$, we have*

$${}^C D_{a+}^{\alpha, \psi} I_{a+}^{\alpha, \psi} f(x) = f(x) \quad \text{and} \quad {}^C D_{b-}^{\alpha, \psi} I_{b-}^{\alpha, \psi} f(x) = f(x).$$

Obviously, one can simply apply $I_{a+}^{\alpha, \psi}$ on both sides of the differential equations to get its integral form. On the other hand, we can apply ${}^C D_{a+}^{\alpha, \psi}$ directly to integral equations in order to recover differential equations.

Lastly, we will show the semigroup law for the ψ -Caputo fractional derivative. Similar to classical fractional derivative [30], semigroup law does not hold in general for fractional derivative but it is indeed true for integrals. In what follows, we present a case that allows semigroup law.

Theorem B.4. *If $f \in C^{m+n}[a, b]$ for some $m \in \mathbb{N}$ and $\alpha > 0$, then for all $k \in \mathbb{N}$ we have*

$$\left(I_{a+}^{\alpha, \psi} \right)^k \left({}^C D_{a+}^{\alpha, \psi} \right)^m f(x) = \left({}^C D_{a+}^{\alpha, \psi} \right)^m f(c) \cdot \frac{(\psi(x) - \psi(a))^{k\alpha}}{\Gamma(k\alpha + 1)},$$

$$\left(I_{b-}^{\alpha, \psi} \right)^k \left({}^C D_{b-}^{\alpha, \psi} \right)^m f(x) = \left({}^C D_{b-}^{\alpha, \psi} \right)^m f(d) \cdot \frac{(\psi(b) - \psi(x))^{k\alpha}}{\Gamma(k\alpha + 1)},$$

for some $c \in (a, x)$ and $d \in (x, b)$.

C Fractional Operators for $\alpha > 0$

A detailed discussion is provided in Section B.

D All theoretical proofs

Proof of Lemma 1. Since

$$\|I_K \mathbf{x}\| \leq C \int_a^t \frac{(\psi(t) - \psi(\tau))^{\alpha-1} \psi'(\tau)}{\Gamma(\alpha)} \|\mathbf{x}\| d\tau \leq C \frac{(\psi(t) - \psi(a))^\alpha}{\Gamma(1 + \alpha)} \|\mathbf{x}\|,$$

it is immediately seen that this operator is bounded.

Proof of Theorem 1. We shall prove the uniqueness as well as stability. To show the uniqueness, we define the operator as

$$T[\mathbf{x}](t) = \mathbf{x}(a) + \int_a^t \frac{(\psi(t) - \psi(\tau))^{\alpha-1} \psi'(\tau)}{\Gamma(\alpha)} \tilde{K}(\mathbf{x}(t), \mathbf{x}(\tau)) f_\theta(\tau, \mathbf{x}(\tau)) d\tau,$$

The Lipschitz properties of \tilde{K} and f lead to:

$$\|T[\mathbf{x}_1] - T[\mathbf{x}_2]\| \leq C \frac{(\psi(t) - \psi(a))^\alpha}{\Gamma(1 + \alpha)} \|\mathbf{x}_1 - \mathbf{x}_2\|,$$

Again, selecting suitable $\epsilon > 0$ and invoking the Banach fixed-point theorem ensures a unique solution. To prove the stability, from (Eq. (15).), we shall get

$$\|\mathbf{x}(t) - \tilde{\mathbf{x}}(t)\|_2 \leq \|\mathbf{x}(a) - \tilde{\mathbf{x}}(a)\|_2 + C \int_a^t \frac{(\psi(t) - \psi(\tau))^{\alpha-1} \psi'(\tau)}{\Gamma(\alpha)} \|\mathbf{x}(\tau) - \tilde{\mathbf{x}}(\tau)\|_2 d\tau.$$

138 Applying fractional Grönwall inequality [30, Lemma 6.19], we derive the stability result.

139 **• Observation and Motivation:** The preceding review of various Caputo fractional derivatives
 140 highlights their defining feature: the use of distinct weighting kernels, which can be static or designed
 141 to vary dynamically with time t . For high-dimensional states $\mathbf{x}(t)$, we can extend this idea by
 142 introducing a learnable vector $\psi(t) = (\psi_1(t), \dots, \psi_n(t))$, where each $\psi_i(t)$ defines a component-
 143 wise kernel in a ψ -Caputo framework, enabling adaptive, dimension-specific memory modeling.
 144 However, since these kernels depend only on t and τ and not on past states $\mathbf{x}(\tau)$ or the current
 145 state $\mathbf{x}(t)$, they cannot adjust their weighting based on the states correlation in the trajectory. *In*
 146 *this paper, we propose to overcome this limitation by developing a more generalizable learnable*
 147 *attention kernel that extends beyond the capabilities of the ψ -based approach. Our framework will*
 148 *incorporate mechanisms that can adapt memory weightings based on both temporal information and*
 149 *the contextual relationships between past and current states.*

150 D.1 Solving KatFDE

151 The integral equation Eq. (15). is a nonlinear equation which can be solved using linearized technique
 152 or iterative method. Here, we adopt the latter one. Taking $x = t_j$ and approximating the integral in
 153 Eq. (15). using the trapezoidal rule yields

$$\mathbf{x}(t_j) = \mathbf{x}(t_0) + \sum_{k=0}^{j-1} K(t_j, t_k, \mathbf{x}(t_j), \mathbf{x}(t_k)) f_{\theta}(t_k, \mathbf{x}(t_k)) h. \quad (\text{D1})$$

The iterative method for above nonlinear problems works as follows: taking the initial guess $\mathbf{x}(t_j^{(0)}) = \mathbf{x}(t_{j-1})$, then we conduct iterations based on

$$\mathbf{x}(t_j^{(L)}) = \mathbf{x}(t_0) + \sum_{k=0}^{j-1} K(t_j, t_k, \mathbf{x}(t_j^{(L-1)}), \mathbf{x}(t_k)) f_{\theta}(t_k, \mathbf{x}(t_k)) h, \quad L \geq 1.$$

154 This procedure will lead to a good approximation of $\mathbf{x}(t_j)$ after a few iterations. To address it, define

$$\mathbf{x}(t_j) = \phi(\mathbf{x}(t_j)), \quad \text{where} \quad \phi(\mathbf{x}(t_j)) = \mathbf{x}(t_0) + \sum_{k=0}^{j-1} K(t_j, t_k, \mathbf{x}(t_j), \mathbf{x}(t_k)) f_{\theta}(t_k, \mathbf{x}(t_k)) h,$$

155 for the sake of simplicity. The proposed iterative solution to the discretized equation can formulated
 156 as the Basic Iteration method.

157 **Basic Iteration Method.** Given an initial guess $\mathbf{x}^{(0)}(t_j) = \mathbf{x}(t_{j-1})$, we iteratively compute

$$\mathbf{x}^{(L)}(t_j) = \phi(\mathbf{x}^{(L-1)}(t_j)), \quad L \geq 1.$$

158 D.1.1 Convergence Criterion

159 The convergence of the above iterative method relies on the following condition: If $\phi : \mathbb{R}^d \rightarrow \mathbb{R}^d$
 160 satisfies a Lipschitz condition with a Lipschitz constant $C < 1$, namely,

$$\|\phi(\mathbf{x}) - \phi(\mathbf{y})\| \leq C \|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in [a, b]^d,$$

161 then the iterative methods converge for any initial guess $\mathbf{x}^{(0)} \in [a, b]^d$.

162 Given that the assumptions of Theorem 1 are satisfied with appropriate generic constant C , it can be
 163 directly verified that basic iteration method proposed here will converge. The essence is to make sure
 164 that the constant is less than 1.

165 D.1.2 Convergence Rate

166 Define the iteration error $e^{(L)} = \mathbf{x}^{(L)} - \mathbf{x}^{(L-1)}$. An iterative method has order of convergence p if
 167 there exists a nonzero constant C such that

$$\lim_{L \rightarrow \infty} \frac{\|e^{(L+1)}\|}{\|e^{(L)}\|^p} = C.$$

168 For the Basic Iteration method, applying the Lipschitz property of ϕ , we obtain

$$\frac{\|e^{(L)}\|}{\|e^{(L-1)}\|} = \frac{\|\phi(\mathbf{x}^{(L-1)}) - \phi(\mathbf{x}^{(L-2)})\|}{\|\mathbf{x}^{(L-1)} - \mathbf{x}^{(L-2)}\|} \leq C < 1,$$

169 indicating a linear (first-order) convergence rate.

170 E Nonsingular scenario for the Kernel

171 Suppose that the kernel K is bounded, Lipschitz continuous with respect to the last two variables. We
 172 find that the operator is bounded that is immediately seen from the boundedness of K . Besides, the
 173 integral equation is well-posed. The proof is as follows:

174 **Uniqueness.** Define operator:

$$T[\mathbf{x}](t) = \mathbf{x}(a) + \int_a^t K(t, \tau, \mathbf{x}(t), \mathbf{x}(\tau)) f_\theta(\tau, \mathbf{x}(\tau)) d\tau,$$

175 Using the Lipschitz assumptions, we obtain

$$\|T[\mathbf{x}_1] - T[\mathbf{x}_2]\| \leq C(t - a) \|\mathbf{x}_1 - \mathbf{x}_2\|.$$

176 Choosing $\epsilon > 0$ small enough that $C\epsilon < 1$ and applying the Banach fixed-point theorem [32] yields
 177 a unique solution.

Stability. From (Eq. (15).), using the Lipschitz assumptions again as well as boundedness of K , it is readily seen that

$$\|\mathbf{x}(t) - \tilde{\mathbf{x}}(t)\|_2 \leq \|\mathbf{x}(a) - \tilde{\mathbf{x}}(a)\|_2 + C \int_a^t \|\mathbf{x}(\tau) - \tilde{\mathbf{x}}(\tau)\|_2 d\tau,$$

178 which gives the desired result by classical Grönwall inequality [33, Lemma B.9].

179 As in singular case, one can also show that the basic iteration method works well for the nonsingular
 180 case, that is, it is convergent with first order.

181 F KatFDE Examples

182 Here shows multiple KatFDE variants based on graph learning tasks. Inspired by the models in
 183 [12], we develop two variants, including Kat-GRAND and Kat-CDE. Similar to [34], Kat-GRAND
 184 includes two versions. One is Kat-GRAND-nl:

$$\int_a^t K(t, \tau, \mathbf{Y}(t)) \left((\mathbf{A}(\mathbf{Y}(t)) - \mathbf{I}) \mathbf{Y}(t) \right) d\tau = \mathbf{Y}(t), \quad (\text{F2})$$

185 where $\mathbf{A}(\mathbf{Y}(t)) = (a_{i,j}(t))$ is given by a nonlinear attention mechanism. The other version is
 186 Kat-GRAND-l:

$$\int_a^t K(t, \tau, \mathbf{Y}(t)) (-\mathbf{L}\mathbf{Y}(t)) d\tau = \mathbf{Y}(t), \quad (\text{F3})$$

187 where \mathbf{L} is a time-invariant matrix, which is a linear FDE.

Furthermore, based on the CDE model [35], the Kat-CDE model has the following expression:

$$\int_a^t K(t, \tau, \mathbf{Y}(t)) \left(\mathbf{A}(\mathbf{Y}(t)) - \mathbf{I} \right) \mathbf{Y}(t) + \text{div}(\mathbf{V}(t) \circ \mathbf{Y}(t)) d\tau = \mathbf{Y}(t), \quad (\text{F4})$$

where the divergence operator $\text{div}(\cdot)$ is introduced by [36], and \circ stands for the element-wise product, also known as the Hadamard product. This model is crafted to handle heterophilic graphs, where connected nodes typically belong to different classes or exhibit distinct features.

G Graph Differential Equation Models

To better understand the baseline models, this section primarily introduces several dynamic comparison networks based on graph learning tasks, namely GRAND [37], CDE [35], FROND [12], DRAGON [28] and NvoFDE [13].

GRAND [37]: The Graph Neural Diffusion (GRAND) model is a graph neural network framework inspired by the heat diffusion process, where information spreads across graph nodes similarly to how heat diffuses through a medium. Its governing differential equation is given by:

$$\frac{d\mathbf{X}(t)}{dt} = (\mathbf{A}(\mathbf{X}(t)) - \mathbf{I})\mathbf{X}(t), \quad (\text{G5})$$

where $\mathbf{A}(\mathbf{X}(t))$ is a learnable attention-based adjacency matrix, and \mathbf{I} is the identity matrix. There are two variants. The update in (G5) defines the **GRAND-nl** model, where the adjacency matrix $\mathbf{A}(\mathbf{X}(t))$ is nonlinear. Let $d_i = \sum_{j=1}^n W_{ij}$ and define the diagonal matrix \mathbf{D} with $D_{ii} = d_i$. The random walk Laplacian is $\mathbf{L} = \mathbf{I} - \mathbf{W}\mathbf{D}^{-1}$. Thus in the simple case, we have the following **GRAND-I** model:

$$\frac{d\mathbf{X}(t)}{dt} = (\mathbf{W}\mathbf{D}^{-1} - \mathbf{I})\mathbf{X}(t) = -\mathbf{L}\mathbf{X}(t). \quad (\text{G6})$$

CDE [35]: In heterophilic graph, nodes often have diverse features, posing significant challenges for graph information processing. To address this issue, the authors introduced the convection-diffusion equations (CDE) into GNNs, and then proposed the Neural CDE model. This model adaptively regulates the rate of information propagation between nodes, enabling selective information sharing among dissimilar neighbors. The corresponding mathematical formulation is given by:

$$\frac{d\mathbf{X}(t)}{dt} = (\mathbf{A}(\mathbf{X}(t)) - \mathbf{I})\mathbf{X}(t) + \text{div}(\mathbf{V}(t) \circ \mathbf{X}(t)), \quad (\text{G7})$$

where $\mathbf{V}(t)$ denotes the velocity field, \circ indicates the element-wise product, and $\text{div}(\cdot)$ represents the divergence operator.

FROND [12]: The FROND framework extends traditional integer-order graph neural differential equations to fractional-order dynamics using the Caputo derivative:

$$D_t^\alpha \mathbf{X}(t) = \mathcal{F}(\mathbf{W}, \mathbf{X}(t)), \quad \alpha > 0, \quad (\text{G8})$$

where \mathcal{F} defines the graph dynamics. By leveraging the non-local nature of fractional calculus, FROND captures long-range dependencies in node features. Similar to (G5), (G6) and (G7), FROND has the following corresponding variants:

(1) **F-GROND-nl**

$$D_t^\alpha \mathbf{X}(t) = (\mathbf{A}(\mathbf{X}(t)) - \mathbf{I})\mathbf{X}(t), \quad 0 < \alpha \leq 1. \quad (\text{G9})$$

(2) **F-GROND-I**

$$D_t^\alpha \mathbf{X}(t) = -\mathbf{L}\mathbf{X}(t), \quad 0 < \alpha \leq 1. \quad (\text{G10})$$

(3) **F-CDE**

$$D_t^\alpha \mathbf{X}(t) = (\mathbf{A}(\mathbf{X}(t)) - \mathbf{I})\mathbf{X}(t) + \text{div}(\mathbf{V}(t) \circ \mathbf{X}(t)), \quad 0 < \alpha \leq 1. \quad (\text{G11})$$

DRAGON [28]: Unlike conventional continuous GNNs that rely on fixed integer or single fractional-order derivatives, DRAGON adopts a learnable distribution over derivative orders:

$$\int_a^b D^\alpha \mathbf{X}(t) d\mu(\alpha) = \mathcal{F}(\mathbf{W}, \mathbf{X}(t)), \quad (\text{G12})$$

where $[a, b]$ defines the domain of α , μ is a learnable distribution, and \mathcal{F} denotes the graph dynamics. Similar to (G5), (G6) and (G7), DRAGON has the following corresponding variants:

(1) **D-GRAND-nl**

$$\int_0^1 D^\alpha \mathbf{X}(t) d\mu(\alpha) = (\mathbf{A}(\mathbf{X}(t)) - \mathbf{I})\mathbf{X}(t). \quad (\text{G13})$$

(2) **D-GRAND-l**

$$\int_0^1 D^\alpha \mathbf{X}(t) d\mu(\alpha) = \mathbf{L}\mathbf{X}(t). \quad (\text{G14})$$

(3) **D-CDE**

$$\int_0^1 D^\alpha \mathbf{X}(t) d\mu(\alpha) = (\mathbf{A}(\mathbf{X}(t)) - \mathbf{I})\mathbf{X}(t) + \text{div}(\mathbf{V}(t) \circ \mathbf{X}(t)). \quad (\text{G15})$$

NvoFDE [13]: NvoFDE extends neural differential equation models by introducing a learnable variable-order derivative $\alpha(t, x(t))$ that dynamically adapts over time and feature space.

$$D_t^{\alpha(t, \mathbf{x}(t))} \mathbf{X}(t) = \mathcal{F}(\mathbf{W}, \mathbf{X}(t)), \quad 0 < \alpha(t, \mathbf{X}(t)) \leq 1. \quad (\text{G16})$$

where \mathcal{F} defines the graph dynamics. Similar to the above, there exist the following variants:

(1) **Nvo-GROND-nl**

$$D_t^{\alpha(t, \mathbf{x}(t))} \mathbf{X}(t) = (\mathbf{A}(\mathbf{X}(t)) - \mathbf{I})\mathbf{X}(t), \quad 0 < \alpha(t, \mathbf{X}(t)) \leq 1. \quad (\text{G17})$$

(2) **Nvo-GROND-l**

$$D_t^{\alpha(t, \mathbf{x}(t))} \mathbf{X}(t) = -\mathbf{L}\mathbf{X}(t), \quad 0 < \alpha(t, \mathbf{X}(t)) \leq 1. \quad (\text{G18})$$

(3) **Nvo-CDE**

$$D_t^{\alpha(t, \mathbf{x}(t))} \mathbf{X}(t) = (\mathbf{A}(\mathbf{X}(t)) - \mathbf{I})\mathbf{X}(t) + \text{div}(\mathbf{V}(t) \circ \mathbf{X}(t)), \quad 0 < \alpha(t, \mathbf{X}(t)) \leq 1. \quad (\text{G19})$$

H Datasets and More Experiments on Graph for KatFDE Model

H.1 Datasets and Setting

The datasets used in this paper are provided separately in Table H1 and Table H2.

Table H1: Dataset statistics used in Table 1 of the main text

Dataset	Type	Classes	Features	Nodes	Edges
Cora	citation	7	1433	2485	5069
Citeseer	citation	6	3703	2120	3679
PubMed	citation	3	500	19717	44324
Coauthor CS	co-author	15	6805	18333	81894
Computers	co-purchase	10	767	13381	245778
Photo	co-purchase	8	745	7487	119043
CoauthorPhy	co-author	5	8415	34493	247962
OGB-Arxiv	citation	40	128	169343	1166243
Airport	tree-like	4	4	3188	3188

The authors revealed critical limitations in the commonly used benchmark datasets for evaluating models on heterophilic graphs [38]. To address this, they introduced several new datasets, such as Roman-empire, Wiki-cooc, Questions, Workers and Amazon-ratings. These datasets, sourced from different fields, have low homophily scores and display a variety of structural properties. We follow the experimental setup specified in the CDE model [35]. For the Workers, and Questions datasets, we employ the ROC-AUC score as the evaluation metric, as these tasks involve binary classification. The performance of our KatFDE-CDE model, is evaluated against several well-known baselines, including TDE-GNN[39], GRAND [37], GraphBel [36], NSD [40], ACMP[41], CDE [35], F-CDE [12], D-CDE[28] and Nvo-CDE[13].

Table H2: Dataset statistics used in Table H3

Dataset	Nodes	Edges	Classes	Node Features
Roman-empire	22662	32927	18	300
Wiki-cooc	10000	2243042	5	100
Minesweeper	10000	39402	2	7
Questions	48921	153540	2	301
Workers	11758	519000	2	10
Amaon-ratings	24492	93050	5	300

H.2 Node Classification on Heterophilic Graph

Performance and Analysis: In Table H3, we present the experimental results on heterophilic graph datasets. It is evident that our model KatFDE-CDE achieves competitive or better performance, demonstrating its effectiveness. On the Workers and Amazon-ratings datasets, KatFDE-CDE achieves the best performance among all compared models, outperforming the Nvo-CDE model by approximately 0.7% on both datasets. This advantage stems from the framework’s flexibility in kernel function design, enabling it to capture more complex feature-updating dynamics.

Table H3: Node classification results(%). The best and the second-best result for each criterion are highlighted in red and blue, respectively.

Model	Roman-empire	Wiki-cooc	Questions	Workers	Amazon-ratings
TDE-GNN[39]	64.29±0.58	84.95±0.78	68.94±1.69	75.13±0.81	40.33±1.37
GRAND-I[37]	69.24±0.53	91.58±0.37	68.54±1.07	75.59±0.86	48.99±0.35
GRAND-nl[37]	71.60±0.58	92.03±0.46	70.67±1.28	75.33±0.84	45.05±0.65
GraphBel[36]	69.47±0.37	90.30±0.50	70.79±0.99	73.02±0.92	43.63±0.42
NSD[40]	77.50±0.67	92.06±0.40	69.25±1.15	79.81±0.99	37.96±0.20
ACMP[41]	71.27±0.59	92.68±0.37	71.18±1.03	75.03±0.92	44.76±0.52
CDE[35]	91.64±0.28	97.99±0.38	75.17±0.99	80.70±1.04	47.63±0.43
F-CDE[12]	93.06±0.55	98.73±0.68	75.17±0.99	82.68±0.86	49.01±0.56
D-CDE[28]	93.87±0.41	98.58±0.12	75.53±0.98	83.02±0.86	49.43±1.26
Nvo-CDE[13]	93.42±0.22	99.32±0.28	74.87±0.23	83.33±0.65	50.09±0.40
KatFDE-CDE (ours)	93.46±0.32	98.94±0.12	75.10±0.11	84.02±0.38	50.75±0.45

I Datasets and More Experiments for Traffic Forecasting

I.1 Datasets and Setting

We evaluate the effectiveness of STDDE using six real-world traffic datasets: PeMSD7(M), PeMSD7(L), PeMS03, PeMS04, PeMS07, and PeMS08. These datasets are sourced from the Caltrans Performance Measurement System [42], which collects traffic flow data every 30 seconds. For analysis, the data is aggregated into 5-minute intervals, resulting in 288 time steps one day. A summary of the dataset statistics can be found in Table I4. These datasets are pre-divided into training, validation, and testing sets using a 6:2:2 ratio. The training procedure and hyperparameter settings are kept consistent with those reported in [18]. For instance, the model is trained for 200 epochs using the Adam optimizer.

I.2 Experimental Results

From I5, our model STG-KatFDE achieves generally the best performance on PeMSD7 dataset. Compared to neural differential equation-based models, it demonstrates a stronger ability to capture complex dynamics. Figure I1 illustrates the predicted traffic flow from STG-KatFDE in comparison with STG-NCDE and the ground truth on PeMSD4 and PeMSD8 datasets. The horizontal axis denotes the time steps (5-minute intervals), and the vertical axis represents the traffic flow. A total of 288 time steps are selected, covering an entire 24-hour period.

Table I4: Datasets for Traffic Forecasting

Datasets	Sensors	Edges	Time Steps
PeMS04	307	340	16992
PeMS07	883	866	28224
PeMS08	170	295	17856
PeMS07(M)	228	1132	12672
PeMS07(L)	1026	10150	12672

Table I5: Forecasting error on PeMSD7

Model	PeMSD7		
	MAE	RMSE	MAPE
HA[43]	45.12	65.64	24.51%
VAR[43]	50.22	75.63	32.22%
TCN[44]	32.72	42.23	14.26%
DSANet[45]	31.36	49.11	14.43%
AGCRN[46]	22.37	36.55	9.12%
STFGNN[47]	23.46	36.60	9.21%
Z-GCNETs[48]	21.77	35.17	9.25%
STGODE[49]	22.59	37.54	10.14%
STG-NCDE[18]	20.53	33.84	8.80%
STG-KatFDE (ours)	20.46	33.70	8.94%

Each subfigure corresponds to a specific node and is annotated with a zoomed-in region to highlight prediction differences in more dynamic or complex traffic periods. Overall, both models demonstrate a strong ability to follow the ground truth trends. However, the proposed STG-KatFDE consistently achieves closer alignment with the ground truth, especially in rapidly changing regions.

Particularly in Figure I1:

- Node 211 and Node 111 in PeMSD4 (top row) show that STG-KatFDE better captures sudden increases and local peaks, maintaining smoother yet accurate transitions.
- Node 167 and Node 123 in PeMSD8 (bottom row) further validate STG-KatFDE superiority, with visibly reduced error margins in congested and fluctuating segments, as shown in the zoom-in windows.

These results support the quantitative findings discussed in the main text and demonstrate the robustness and generalization capacity of STG-KatFDE across different traffic environments.

I.3 Parameter Analysis

Hidden Dimension Analysis: Figure I2 presents the performance of STG-KatFDE on the PeMSD8 dataset with varying input feature dimensions: 16, 32, 64, and 128. It can be observed that as the feature dimension increases, the model’s performance improves consistently across all three metrics. In particular, the lowest RMSE and MAE are achieved at dimension 128, indicating that higher-dimensional representations help capture more complex spatiotemporal patterns in traffic data. Notably, the improvement becomes more pronounced when increasing the dimension from 32 to 64, and then stabilizes between 64 and 128. These findings suggest that while increasing feature dimensionality benefits performance, the marginal gain diminishes beyond a certain point.

Step Size Analysis: Table I6 reports the forecasting error metrics of STG-KatFDE on the PeMSD4 and PeMSD8 datasets, evaluated with varying step sizes. As the step size increases, MAE increases from 19.48 to 19.88, and RMSE from 31.34 to 31.83 on PeMSD4; while on PeMSD8, MAE changes marginally from 17.45 to 17.10 and RMSE slightly decreases from 27.55 to 26.83. The experiments suggest that a moderately larger step size contributes to improved performance on PeMSD8, while it has the opposite effect on PeMSD4.

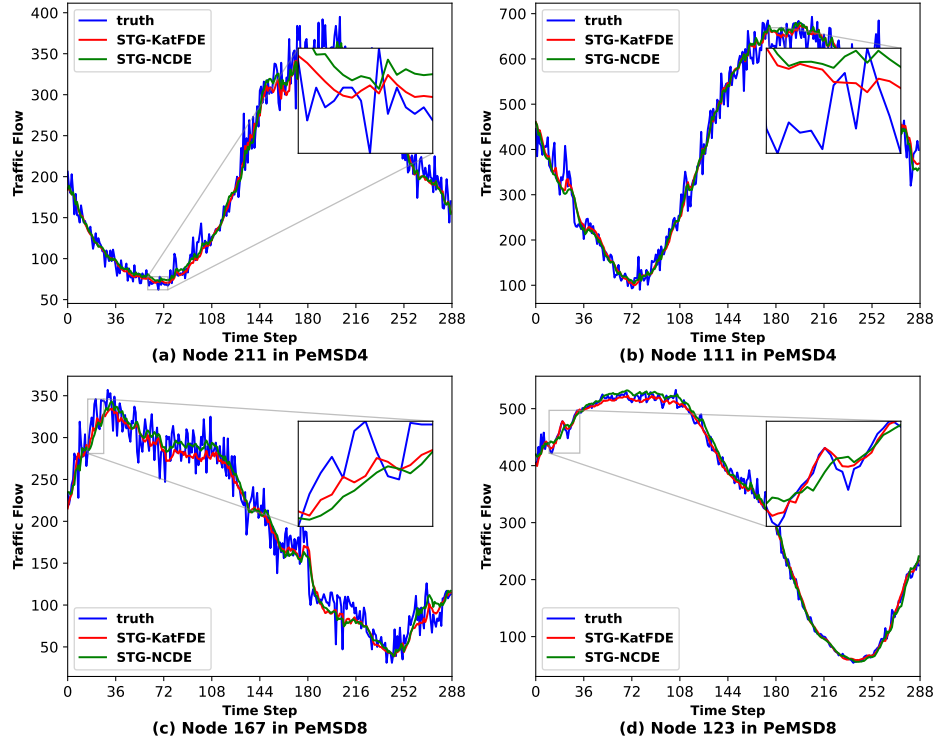


Figure I1: Traffic forecasting visualization in PeMSD4 and PeMSD8

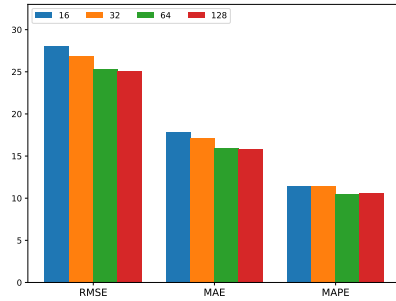


Figure I2: Performance comparison of STG-KatFDE with varying feature dimensions on the PeMSD8 dataset.

Table I6: Forecasting error metrics (MAE, RMSE, MAPE) for different step sizes on PeMSD4 and PeMSD8 datasets.

Step size	PeMSD4			PeMSD8		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE
0.5	19.48	31.34	12.94%	17.45	27.55	11.07%
1.0	19.88	31.83	13.14%	17.10	26.83	10.66%

References

- [A-1] I. Podlubny, *Fractional Differential Equations*. Academic Press, 1999.
- [A-2] M. Caputo and M. Fabrizio, “A new definition of fractional derivative without singular kernel,” *Progress in Fractional Differentiation & Applications*, vol. 1, no. 2, pp. 73–85, 2015.
- [A-3] K. Diethelm and N. J. Ford, “Analysis of fractional differential equations,” *J. Math. Anal. Appl.*, vol. 265, no. 2, pp. 229–248, 2002.
- [A-4] R. Hilfer, *Applications of fractional calculus in physics*. World scientific, 2000.
- [A-5] A. Atangana and D. Baleanu, “New fractional derivatives with nonlocal and non-singular kernel: theory and application to heat transfer model,” *arXiv preprint arXiv:1602.03408*, 2016.
- [A-6] S. G. Samko and B. Ross, “Integration and differentiation to a variable fractional order,” *Integral transforms and special functions*, vol. 1, no. 4, pp. 277–300, 1993.
- [A-7] C. F. Coimbra, “Mechanics with variable-order differential operators,” *Annalen der Physik*, vol. 515, no. 11-12, pp. 692–703, 2003.
- [A-8] I. Podlubny, “An introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications,” *Math. Sci. Eng.*, vol. 198, no. 340, pp. 0924–34 008, 1999.
- [A-9] Y. Kang, S. Mao, and Y. Zhang, “Fractional time-varying grey traffic flow model based on viscoelastic fluid and its application,” *Transportation research part B: methodological*, vol. 157, pp. 149–174, 2022.
- [A-10] F. Mainardi, *Fractional calculus and waves in linear viscoelasticity: an introduction to mathematical models*. World Scientific, 2022.
- [A-11] H. Antil, R. Khatri, R. Löhner, and D. Verma, “Fractional deep neural network via constrained optimization,” *Mach. Learn.: Sci. Technol.*, vol. 2, no. 1, p. 015003, 2020.
- [A-12] Q. Kang, K. Zhao, Q. Ding, F. Ji, X. Li, W. Liang, Y. Song, and W. P. Tay, “Unleashing the potential of fractional calculus in graph neural networks with FROND,” in *Proc. International Conference on Learning Representations*, 2024.
- [A-13] W. Cui, Q. Kang, X. Li, K. Zhao, W. P. Tay, W. Deng, and Y. Li, “Neural variable-order fractional differential equation networks,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 15, 2025, pp. 16 109–16 117.
- [A-14] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, “Neural ordinary differential equations,” in *Advances Neural Inf. Process. Syst.*, 2018.
- [A-15] E. Dupont, A. Doucet, and Y. W. Teh, “Augmented neural odes,” in *Advances Neural Inf. Process. Syst.*, 2019, pp. 1–11.
- [A-16] P. Kidger, J. Morrill, J. Foster, and T. Lyons, “Neural controlled differential equations for irregular time series,” *Advances in neural information processing systems*, vol. 33, pp. 6696–6707, 2020.
- [A-17] M. Poli, S. Massaroli, J. Park, A. Yamashita, H. Asama, and J. Park, “Graph neural ordinary differential equations,” *arXiv preprint arXiv:1911.07532*, 2019.
- [A-18] J. Choi, H. Choi, J. Hwang, and N. Park, “Graph neural controlled differential equations for traffic forecasting,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 36, no. 6, 2022, pp. 6367–6374.
- [A-19] K. Zhao, Q. Kang, F. Ji, X. Li, Q. Ding, Y. Zhao, W. Liang, and W. P. Tay, “Distributed-order fractional graph operating network,” in *Advances Neural Inf. Process. Syst.*, Dec. 2024.

- [A-20] 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 Neural Inf. Process. Syst.*, 2017, pp. 1–11.
- [A-21] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *International Conference on Learning Representations*, 2020.
- [A-22] B. Lim, S. Ö. Arık, N. Loeff, and T. Pfister, “Temporal fusion transformers for interpretable multi-horizon time series forecasting,” *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748–1764, 2021.
- [A-23] J. Zhang, P. Zhang, B. Kong, J. Wei, and X. Jiang, “Continuous self-attention models with neural ode networks,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 16, 2021, pp. 14 393–14 401.
- [A-24] P. Wang, T. Zhang, H. Zhang, S. Cheng, and W. Wang, “Adding attention to the neural ordinary differential equation for spatio-temporal prediction,” *International Journal of Geographical Information Science*, vol. 38, no. 1, pp. 156–181, 2024.
- [A-25] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–12.
- [A-26] H. Kim, A. Mnih, J. Schwarz, M. Garnelo, A. Eslami, D. Rosenbaum, O. Vinyals, and Y. W. Teh, “Attentive neural processes,” *arXiv preprint arXiv:1901.05761*, 2019.
- [A-27] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, “Heterogeneous graph attention network,” in *The world wide web conference*, 2019, pp. 2022–2032.
- [A-28] K. Zhao, X. Li, Q. Kang, F. Ji, Q. Ding, Y. Zhao, W. Liang, and W. P. Tay, “Distributed-order fractional graph operating network,” in *Advances Neural Inf. Process. Syst.*, 2024, pp. 1–14.
- [A-29] R. Almeida, “A caputo fractional derivative of a function with respect to another function,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 44, pp. 460–481, 2017.
- [A-30] K. Diethelm, *The analysis of fractional differential equations: an application-oriented exposition using differential operators of Caputo type.* Springer, 2010, vol. 2004.
- [A-31] F. Jarad, T. Abdeljawad, and D. Baleanu, “Caputo-type modification of the hadamard fractional derivatives,” *Advances in Difference Equations*, vol. 2012, pp. 1–8, 2012.
- [A-32] E. Zeidler, *Applied Functional Analysis*, ser. Applied Mathematical Sciences, J. E. Marsden and L. Sirovich, Eds. New York, NY: Springer New York, 1995, vol. 108.
- [A-33] J. Shen, T. Tang, and L.-L. Wang, *Spectral Methods: Algorithms, Analysis and Applications*, ser. Springer Series in Computational Mathematics. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, vol. 41.
- [A-34] B. P. Chamberlain, J. Rowbottom, M. Goronova, S. Webb, E. Rossi, and M. M. Bronstein, “Grand: Graph neural diffusion,” in *Proc. Int. Conf. Mach. Learn.*, 2021.
- [A-35] K. Zhao, Q. Kang, Y. Song, R. She, S. Wang, and W. P. Tay, “Graph neural convection-diffusion with heterophily,” in *Proc. Inter. Joint Conf. Artificial Intell.*, China, 2023.
- [A-36] Y. Song, Q. Kang, S. Wang, K. Zhao, and W. P. Tay, “On the robustness of graph neural diffusion to topology perturbations,” in *Advances Neural Inf. Process. Syst.*, 2022.
- [A-37] B. Chamberlain, J. Rowbottom, M. I. Goronova, M. Bronstein, S. Webb, and E. Rossi, “Grand: Graph neural diffusion,” in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 1407–1418.
- [A-38] O. Platonov, D. Kuznedelev, M. Diskin, A. Babenko, and L. Prokhorenkova, “A critical look at the evaluation of gnns under heterophily: Are we really making progress?” *arXiv preprint arXiv:2302.11640*, 2023.

- 389 [A-39] M. Eliasof, E. Haber, E. Treister, and C.-B. B. Schönlieb, “On the temporal domain of differ-
390 ential equation inspired graph neural networks,” in *International Conference on Artificial*
391 *Intelligence and Statistics*. PMLR, 2024, pp. 1792–1800.
- 392 [A-40] C. Bodnar, F. D. Giovanni, B. P. Chamberlain, P. Liò, and M. M. Bronstein, “Neural sheaf dif-
393 fusion: A topological perspective on heterophily and oversmoothing in GNNs,” in *Advances*
394 *Neural Inf. Process. Syst.*, 2022.
- 395 [A-41] Y. Wang, K. Yi, X. Liu, Y. G. Wang, and S. Jin, “Acmp: Allen-cahn message passing
396 with attractive and repulsive forces for graph neural networks,” in *Proc. Int. Conf. Learn.*
397 *Representations*, 2022.
- 398 [A-42] C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, “Freeway performance measurement
399 system: mining loop detector data,” *Transportation research record*, vol. 1748, no. 1, pp.
400 96–102, 2001.
- 401 [A-43] J. D. Hamilton, *Time series analysis*. Princeton university press, 2020.
- 402 [A-44] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and
403 recurrent networks for sequence modeling,” *arXiv preprint arXiv:1803.01271*, 2018.
- 404 [A-45] S. Huang, D. Wang, X. Wu, and A. Tang, “Dsanet: Dual self-attention network for multi-
405 variate time series forecasting,” in *Proceedings of the 28th ACM international conference on*
406 *information and knowledge management*, 2019, pp. 2129–2132.
- 407 [A-46] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, “Adaptive graph convolutional recurrent
408 network for traffic forecasting,” *Advances in neural information processing systems*, vol. 33,
409 pp. 17 804–17 815, 2020.
- 410 [A-47] M. Li and Z. Zhu, “Spatial-temporal fusion graph neural networks for traffic flow forecasting,”
411 in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 5, 2021, pp.
412 4189–4196.
- 413 [A-48] Y. Chen, I. Segovia, and Y. R. Gel, “Z-gcnets: Time zigzags at graph convolutional networks
414 for time series forecasting,” in *International Conference on Machine Learning*. PMLR,
415 2021, pp. 1684–1694.
- 416 [A-49] Z. Fang, Q. Long, G. Song, and K. Xie, “Spatial-temporal graph ode networks for traffic flow
417 forecasting,” in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery*
418 *& data mining*, 2021, pp. 364–373.