Rediscovering Chaos? Analysis of GPU Computing Effects in Graph-coupled NeuralODEs

Simon Heilig

M.Sc. Data Science, FAU Erl-Nbg, simon99.heilig@gmail.com

ellis

RUHR UNIVERSITÄT BOCHUM RUB

Take-Home Message

- Discrete NeuralODEs are (residual) Neural Networks
- Powerful formalism for analyzing theoretical properties of graph neural networks
- Exponential receptive field introduces numerical challenges
- GPU neighborhood aggregation is not deterministic and introduces significant noise
- Deterministic GPU operations help at the cost of increased computation time

Background

NeuralODEs for ResNets

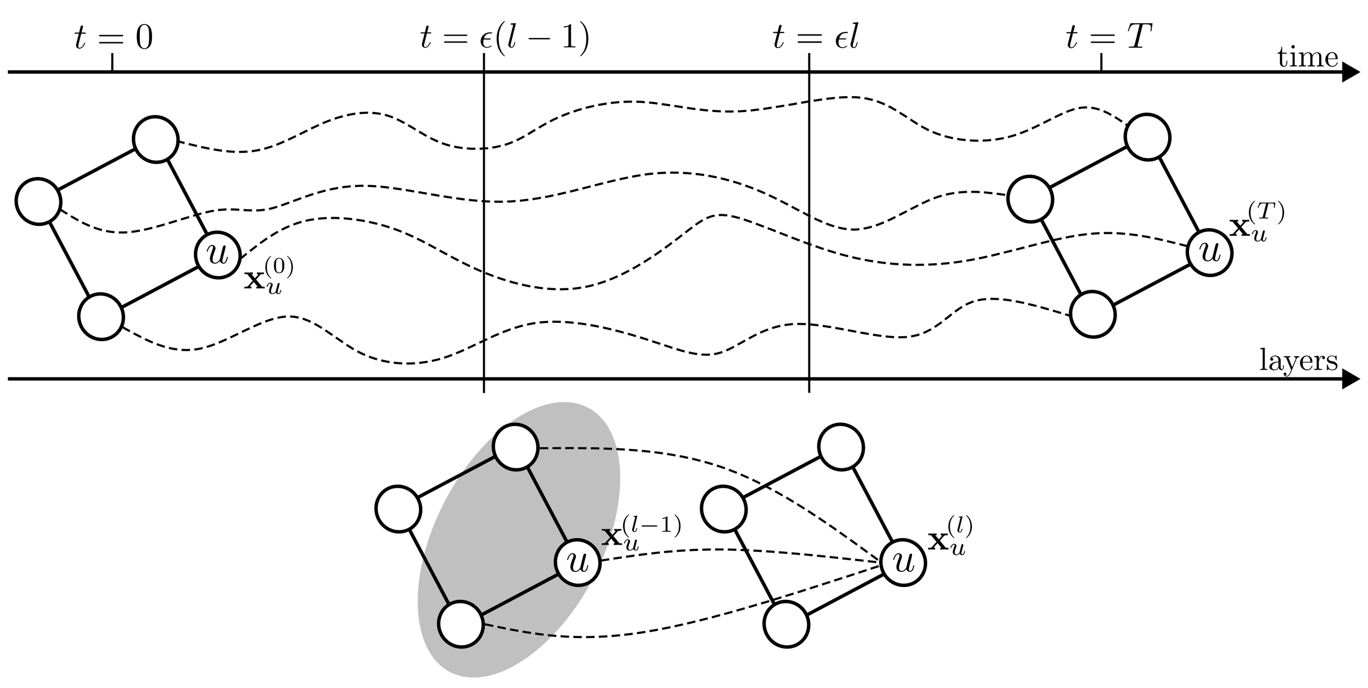
$$\frac{\partial \mathbf{x}(t)}{\partial t} = \sigma(\mathbf{W}(t)\mathbf{x}(t) + \mathbf{b}(t)) \quad , t \in [0, T] \tag{1}$$

discretized by forward Euler scheme ($\frac{\partial \mathbf{x}(t)}{\partial t} \approx \frac{\mathbf{x}(t+\epsilon) - \mathbf{x}(t)}{\epsilon}$)

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} + \epsilon\sigma(\mathbf{W}^{(n+1)}\mathbf{x}^{(n)} + \mathbf{b}^{(n+1)}) \tag{2}$$

Here, $\mathbf{W} \in \mathbb{R}^{d\times d}, \mathbf{b} \in \mathbb{R}^d$ are the parameters of the system with state $\mathbf{x} \in \mathbb{R}^d$ at time $t = \epsilon n \geq 0$ and σ is a nonlinear activation function, e.g., tanh or ReLU [3].

Graph-coupled Neural ODEs for GNNs



$$\frac{\partial \mathbf{X}(t)}{\partial t} = \sigma(\mathbf{X}(t)\mathbf{W}(t) + \mathbf{A}\mathbf{X}(t)\mathbf{V}(t) + \mathbf{b}(t)) \quad , t \in [0, T] \tag{3}$$
$$\mathbf{X}(0) = \mathbf{X}^0$$

Here, $\mathbf{X} \in \mathbb{R}^{n\times d}$ is the feature matrix of n nodes in an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with node set $\mathcal{V}$ and edge set $\mathcal{E}$ summarized in the adjacency matrix $\mathbf{A} \in \{0,1\}^{n\times n}$ [2].

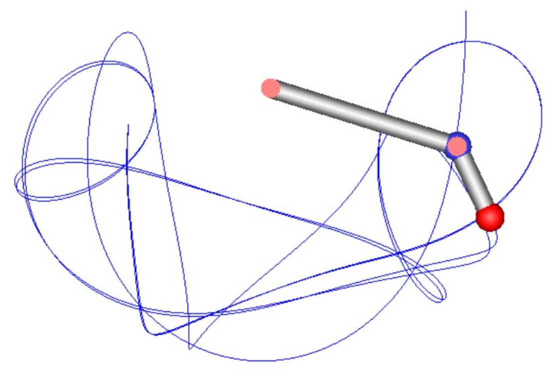
Scatter Operation

Atomic operation on GPU by Pytorch Geometric [1]:

$$\text{out}_i = \text{out}_i + \sum_j \text{src}_j \tag{4}$$

Chaos in Deterministic Dynamical Systems

Nonlinear coupled systems can show chaotic behavior [4]. That means, high sensitivity to the initial condition, numerical errors, and finite precision, leading to non-reproducible trajectories. E.g. Double Pendulum [5]:



$\mathbf{x}'(0) = \mathbf{x}(0) + \delta\mathbf{x}(0)$ and λ Lyapunov exponent of $\dot{\mathbf{x}}(t) = f(\mathbf{x})$

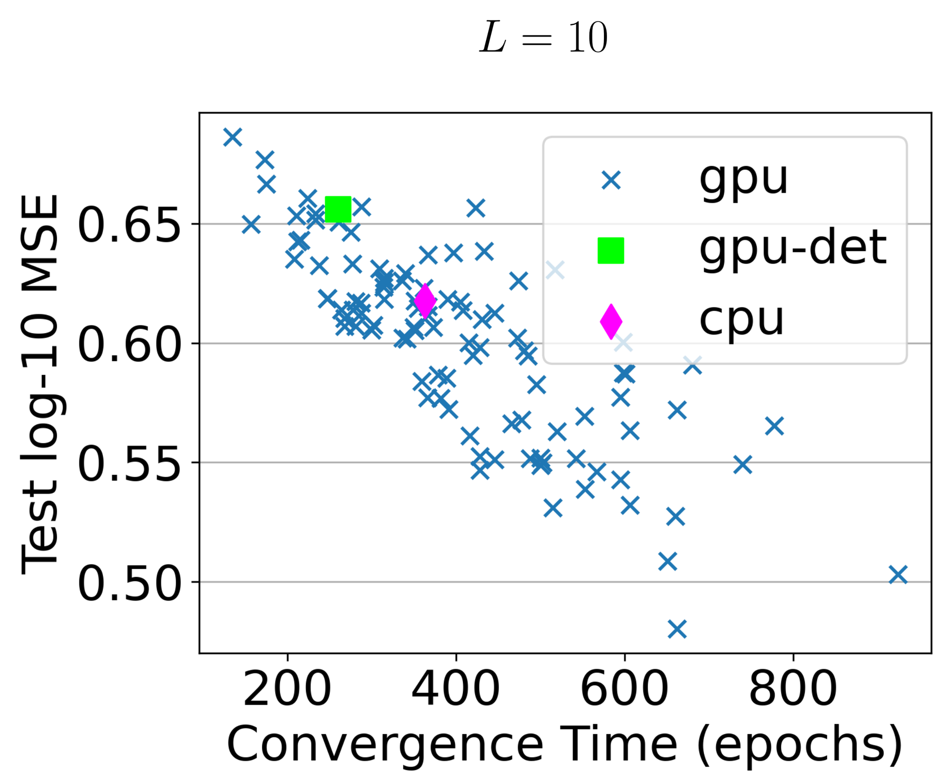
$$\|\mathbf{x}(t) - \mathbf{x}'(t)\| \approx \|\mathbf{x}(0) - \mathbf{x}'(0)\| e^{\lambda t}$$

Experiments

Setup

Predicting the Eccentricity of each node requires the approximation of shortest paths [2]. Repeat 100 isolated training and evaluation runs controlled by fixed seed on GPU, CPU, and GPU with deterministic scatter. Model selection: $L \in \{1, 5, 10, 15, 20\}$ with $\epsilon = 1.0$.

Results



L	Ranked #1	CPU Rank
1	0	5
5	0	4
10	74	1
15	12	3
20	14	2

GPU	GPU-det	CPU
0.29 s	0.49 s	4.54 s

Contact Information

Simon Heilig

Email: simon99.heilig@gmail.com

ELLIS Ph.D. Track, Asja Fischer RUB

Link to Poster

References

[1] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[2] Alessio Gravina, Davide Bacciu, and Claudio Gallicchio. Anti-Symmetric DGN: a stable architecture for Deep Graph Networks. In *The Eleventh International Conference on Learning Representations*, 2023.

[3] E. Haber and L. Ruthotto. Stable architectures for deep neural networks. *Inverse Problems*, 34(1), 2017.

[4] Heinz Georg Schuster and Wolfram Just. *Deterministic chaos: an introduction*. John Wiley & Sons, 2006.

[5] Dirk Zimmer. A planar mechanical library for teaching modelica. In *Proceedings of the 9th International MODELICA Conference*, pages 681–690. Linköping University Press, 2012.

International Workshop on Self-Organizing Maps and Learning Vector Quantization, Clustering and Data Visualization

WSOM+ MiWoCI 2024