Modeling Complex Systems with Graph Neural ODEs

Moritz Laber

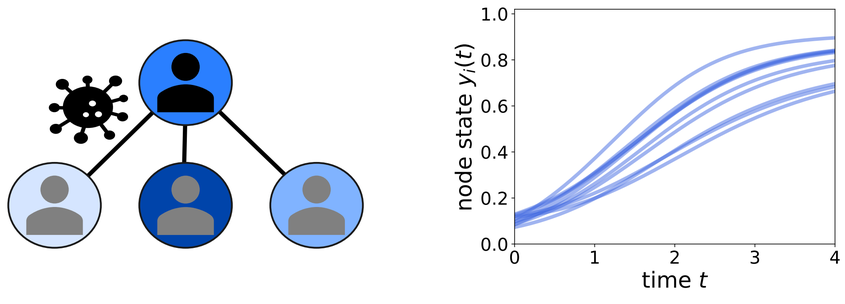
Network Science Institute, Northeastern University, Boston MA 02115,
✉ laber.m@northeastern.edu

Abstract

Combining neural differential equations (NDEs) with relational inductive bias from an underlying graph is a promising modelling approach for dynamical systems from various domains. However, the robustness of this approach across dynamics, graphs, and data quality remains poorly understood. Here we take a first step in this direction by assessing the ability of graph NDEs to generalize to unseen initial conditions on the training graph and new graphs from the same random graph ensemble.

Introduction & Background

- **Complex Systems** from various domains can be modelled by a system of ordinary differential equations (ODEs), coupled via an underlying graph. Examples include neural activity, epidemic outbreaks, chemical reactions, and spreading of rumors. [1]



(a) (b)

Figure 1. **(a)**: Schematic depiction of a dynamical systems on a graph: The infection status of a node (color coded) in the Susceptible-Infected-Susceptible (SIS) model depends on the infection status of its neighbors. **(b)**: Simulation of the SIS model (recovery rate $\mu = 0.2$, infection rate, $\beta = 0.4$) on random network of $n = 10$ nodes.

- **Neural Differential Equations** (NDEs) are a family of models that use neural networks to define the vector field of an (unkown) differential equation.
  - *Neural Ordinary Differential Equations* find applications in time series modelling, classification, and as part of normalizing flows. [2, 3].
  - *Graph Neural Differential Equations* combine NDEs and message passing to approximate dynamical systems on graphs or as continuous models for graph representation learning. [5, 7]
  - *Robustness Checks* for (graph) NDEs across a wide range of dynamical systems are rare. The recent work of Vasiliauskaite & Antulov-Fantulin [6] is closest to ours.

Dynamics on Graphs

- **Dynamics:** We consider a graph with node set V and adjacency matrix A. The state $y_i(t)$ of node i evolves according to the ODE,

$$\frac{dy_i}{dt} = f(y_i) + \sum_{i \in V} A_{ij} q(y_i, y_j), \tag{1}$$

  where $f : \mathbb{R} \to \mathbb{R}$ is called self-dynamics and the interaction $q : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is assumed to factorize, $q(y_i, y_j) = q^{(1)}(y_i)q^{(2)}(y_j)$. We use five dynamical systems from epidemiology, bio-chemistry, population dynamics [4].

model	$f(y)$	$q^{(1)}(x)$	$q^{(2)}(y)$
Suscept. Infect. Suscept. (SIS)	$-\mu y$	$\beta(1-y)$	y
Mass-Action Kinetics (MAK)	$\nu - \mu y$	$-\beta y$	y
Michaelis-Menten Dynamics (MM)	$-\mu y$	β	$\frac{y^\alpha}{1+y^\alpha}$
Birth-Death Process (BD)	$-\mu y^{\alpha_1}$	β	y^{α_2}
Neural Dynamics (ND)	$-\mu y + \nu \tanh(y)$	β	$\tanh(y)$

Table 1. Functional form of various dynamical systems according to [4]. The partitioning of the interaction term into $q(y_i, y_j) = q^{(1)}(y_i)q^{(2)}(y_j)$ is not unique, and chosen arbitrarily.

- **Graphs:** We use the non-linear preferential attachment model to generate random graphs. In this growing graph model nodes arrive one after the other with a fixed number m of links attached to them and node i connects to an existing node j with probability $p_{ij} \propto k_j^\rho$, where k_j is node js degree and $\rho \in [0, \infty)$. We set $n = 100$, $m = 2$, and $\rho = 0$ (Erdős-Rényi (ER) model) or $\rho = 1$ (Barabási-Albert (BA) model).

Model

- **Model:** We obtain a neural version of eq. (1) by replacing $f, q^{(1)}, q^{(2)}$ by neural networks $f_\theta, q_\theta^{(1)}, q_\theta^{(2)}$ with parameters θ,

$$\frac{dy_i}{dt} = \Phi_\theta(y_i, \{y_j\}_{j \in \mathcal{N}_i}) = f_\theta(y_i) + \sum_{i \in V} A_{ij} q_\theta^{(1)}(y_i) q_\theta^{(2)}(y_j) \ , \tag{2}$$

  Equation (2) can be integrated with standard numerical solvers. We use MLPs with 2 layers and 16 hidden units for $f_\theta, q_\theta^{(1)}, q_\theta^{(2)}$ and integrate with the `DOPRI` method. We compare this factorized NDE to a NDE parametrized by a graph convolutional network (GCN), to a NDE parameterized by a MLP, and to GRU.

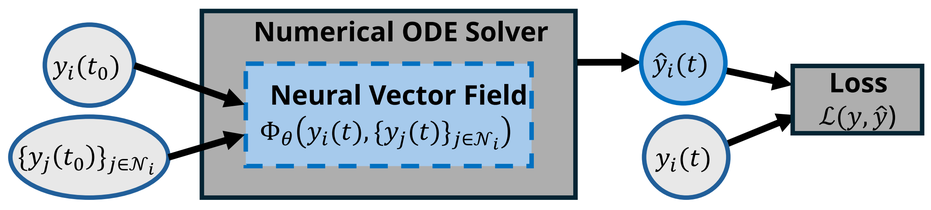


Figure 2. Schematic representation of a node-wise, neural ordinary differential equation: A neural vector field Φ_θ is numerically integrated starting from the focal node i's initial state $y_i(0)$ and the initial states $\{y_j(0)\}_{j \in \mathcal{N}_i}$ of its neighbors to obtain a predicted trajectory $\hat{y}_i(t)$. The loss is computed as the mean square error between the observed and predicted node states at all times. Subsequently weights are updated using backpropagation.

Results

We find that taking into account the graph structure substantially improves predictive performance, especially in the inductive setting. Tayloring the NDE even closer to the underlying system by factorizing the interaction term leads to additional improvements. The results are robust across types of graphs and dynamical systems.

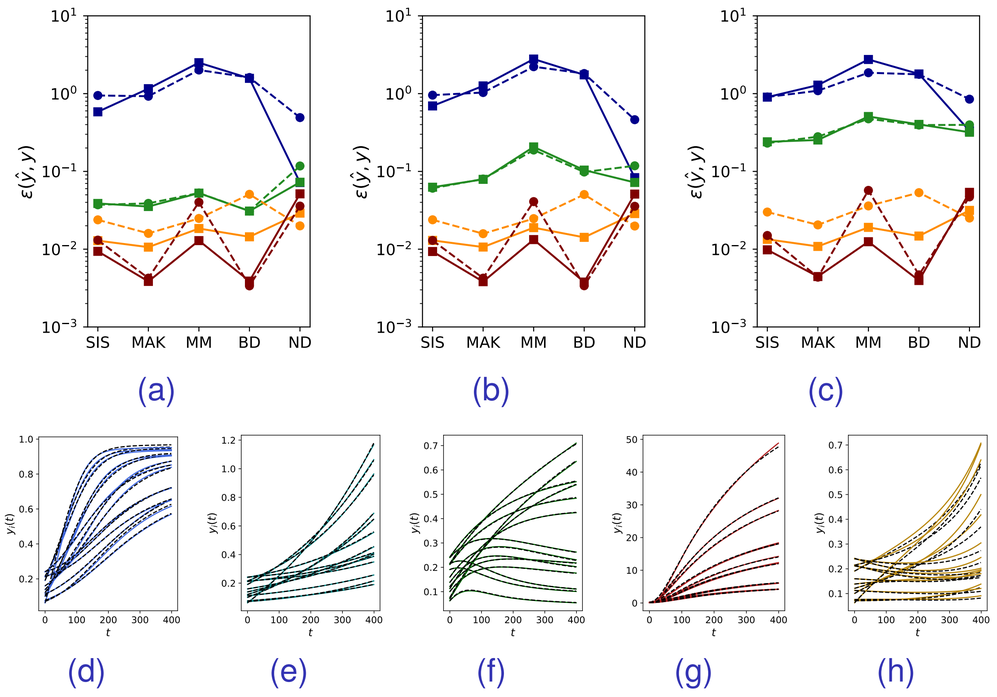


Figure 3. The relative error $\epsilon(\hat{y}, y)$ averaged over time and nodes for the GRU (blue), MLP NDE (green), convolutional NDE (orange), and factorized NDE (dark red) for five different dynamical systems on ER-graphs ($\rho = 0$, squares, solid lines) and BA-graphs ($\rho = 1$, circles, dashed lines) on (a) the training set, (b) the transductive test set, (c) the inductive test set decreases with the amount inductive bias incorporated into the architecture. The comparison of fourteen example predicted trajectories (solid, colored) lines on degree heterogeneous graphs compared to the numerical solution (dashed, black) shows that the quality of prediction depends on the dynamical system ((d) SIS, (e) BD, (f) MAK (g) ND (h) MM).

Discussion & Future Work

Graph NDEs are able to learn dynamical systems from various domains and generalize to new initial conditions on the same graph as well as to new graphs from the same ensemble. Extending this analysis to assess their ability to generalize across graph ensembles, basins of attraction, and in the presence of noise are important next steps.

References

[1] A. Barrat, M. Barthélemy, and A. Vespignani. *Dynamical Processes on Complex Networks*. Cambridge University Press, Cambridge, 2008. ISBN 978-0-521-87950-7.

[2] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

[3] P. Kidger. *On Neural Differential Equations*. PhD thesis, University of Oxford, Feb. 2022.

[4] C. Meena, C. Hens, S. Acharyya, S. Haber, S. Boccaletti, and B. Barzel. Emergent stability in complex network dynamics. *Nature Physics*, 19(7): 1033--1042, 2023.

[5] M. Poli, S. Massaroli, J. Park, A. Yamashita, H. Asama, and J. Park. Graph Neural Ordinary Differential Equations. In *AAAI*, 2022.

[6] V. Vasiliauskaite and N. Antulov-Fantulin. How accurate are neural approximations of complex network dynamics? *arXiv preprint arXiv:2301.04900*, 2023.

[7] C. Zang and F. Wang. Neural dynamics on complex networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 892--902, 2020.