Graph Neural Ordinary Differential Equations

Michael Poli[1*], Stefano Massaroli[2*], Junyoung Park[1*]
Atsushi Yamashita[2], Hajime Asama[2], Jinkyoo Park[1]

[1]Korea Advanced Institute of Technology (KAIST),
[2]The University of Tokyo,
[*]Equal contribution authors

The First International Workshop on Deep Learning on Graphs: Methodologies and Applications (DLGMA'20)

2020-02–08

Graph Neural ODEs (GDEs)

Objective: develop the **continuous–depth** paradigm for deep learning.

Graph Neural Networks

$$\begin{cases} \mathbf{H}_{s+1} = \mathbf{H}_s + \mathbf{F}_{\mathcal{G}}\left(s, \mathbf{H}_s, \boldsymbol{\Theta}_s\right) \\ \mathbf{H}_0 = \mathbf{X}_e \end{cases}, \quad s \in \mathbb{N}$$

where $\mathbf{F}$ is a matrix–valued nonlinear function conditioned on graph $\mathcal{G}$ and $\boldsymbol{\Theta}_s$ is the tensor of **trainable parameters** of the s-th layer.

Graph Neural ODEs (GDEs) [Proposed]

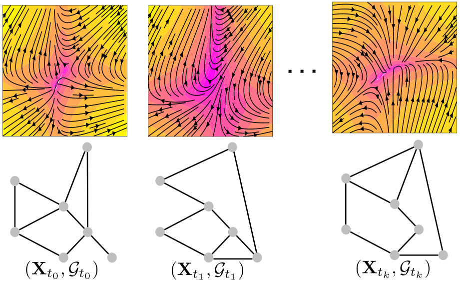
$$\begin{cases} \dot{\mathbf{H}}_s = \mathbf{F}_{\mathcal{G}}\left(s, \mathbf{H}_s, \boldsymbol{\Theta}\right) \\ \mathbf{H}_0 = \mathbf{X}_e \end{cases}, \quad s \in \mathcal{S} \subset \mathbb{R} \tag{1}$$

where $\mathbf{F} : \mathcal{S} \times \mathbb{R}^{n \times d} \times \mathbb{R}^p \to \mathbb{R}^{n \times d}$ is a **depth–varying vector field** defined on $\mathcal{G}$.

GDEs blend **discrete topological structures** and **differential equations**.

- **Static settings:** computational advantages by incorporation of **numerical methods** in the forward pass.
- **Dynamic settings:** exploitation of the **geometry of the underlying dynamics** and flexibility with respect to **irregular observations**.

$$\dot{\mathbf{H}}_t = \mathbf{F}(t, \mathbf{H}_t, \boldsymbol{\Theta})$$



Hybrid system perspective

Autoregressive GDEs handle **sequences of graphs** (dynamic topology – jumps).