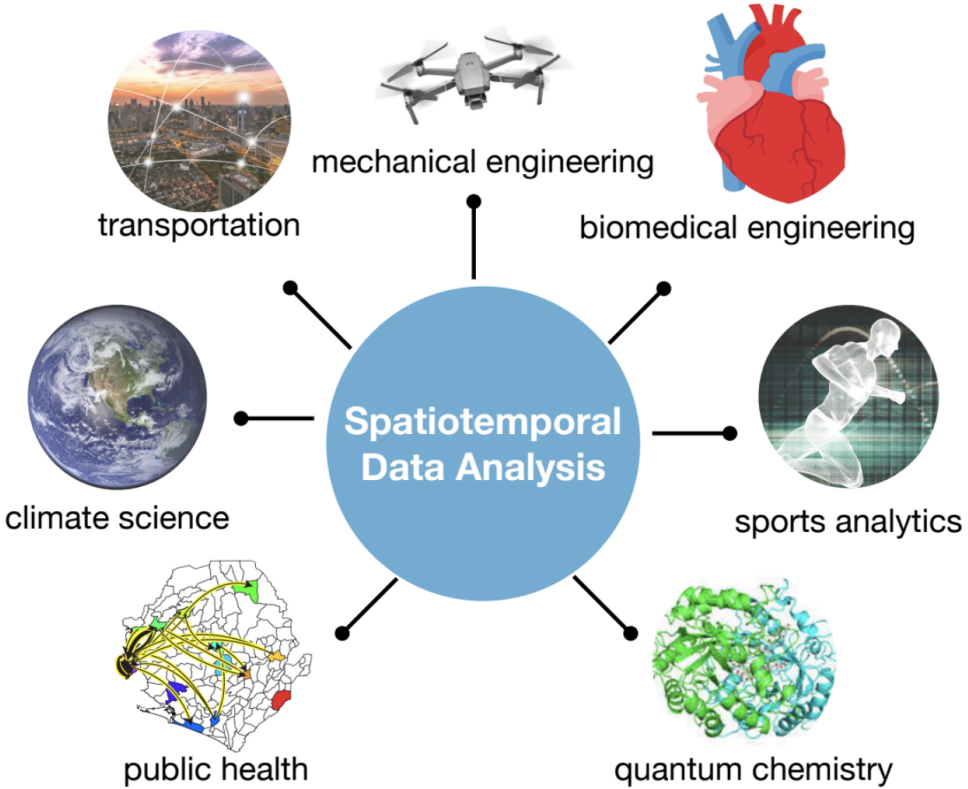
UCLA Samueli Computer Science

ScAi

Deep Learning for Reasoning Over Graph Structured Dynamic Data

Zijie Huang
Advisors: Prof. Yizhou Sun & Prof. Wei Wang

Large-Scale Spatio-Temporal Data



Research Opportunities:

- Uncover dynamic laws automatically from data
- Accelerate the simulation process for large-scale dynamical systems
- …

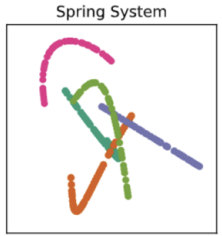
Reseach Tasks:

- Classification
- Clustering
- Forecasting
- Imputation
- …

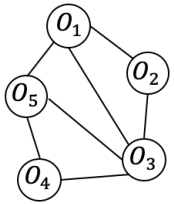
Figure from https://sites.google.com/view/kdd2021tutorial/home?authuser=0

An Example: Multi-Agent Dynamical System

- Reasoning over the spatio-temporal data is non-trivial due to the complex interplay of both spatial and temporal properties, compared with static graph modeling.



Observed trajectories



Interaction graph

Example: predicting trajectories of a 5-body spring system with fixed graph structure.

- Existing neural network models employ graph neural networks (GNN) to approximate pair-wise object interactions, which reveals how system changes from timestamp t, to timestamp t+1.

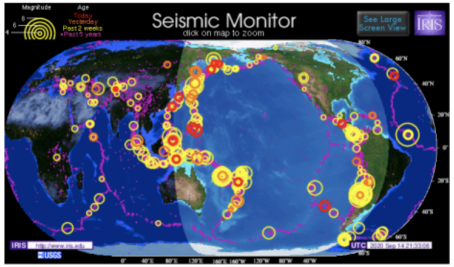
$$v \to e: \quad \tilde{\mathbf{h}}^t_{(i,j)} = \sum_k z_{ij,k} \tilde{f}^k_e([\mathbf{x}^t_i, \mathbf{x}^t_j])$$

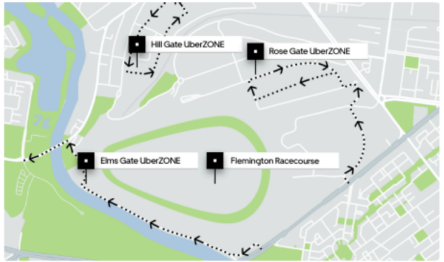
$$e \to v: \quad \boldsymbol{\mu}^{t+1}_j = \mathbf{x}^t_j + \tilde{f}_v(\textstyle\sum_{i \neq j} \tilde{\mathbf{h}}^t_{(i,j)})$$

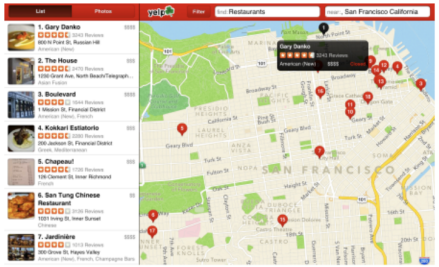
$$p(\mathbf{x}^{t+1}_j | \mathbf{x}^t, \mathbf{z}) = \mathcal{N}(\boldsymbol{\mu}^{t+1}_j, \sigma^2 \mathbf{I})$$

Challenges in Spatio-Temporal Modeling

- Learning under non-uniform data distributions.
  - Irregular space, irregular time: data collected at changing locations and irregular time intervals.



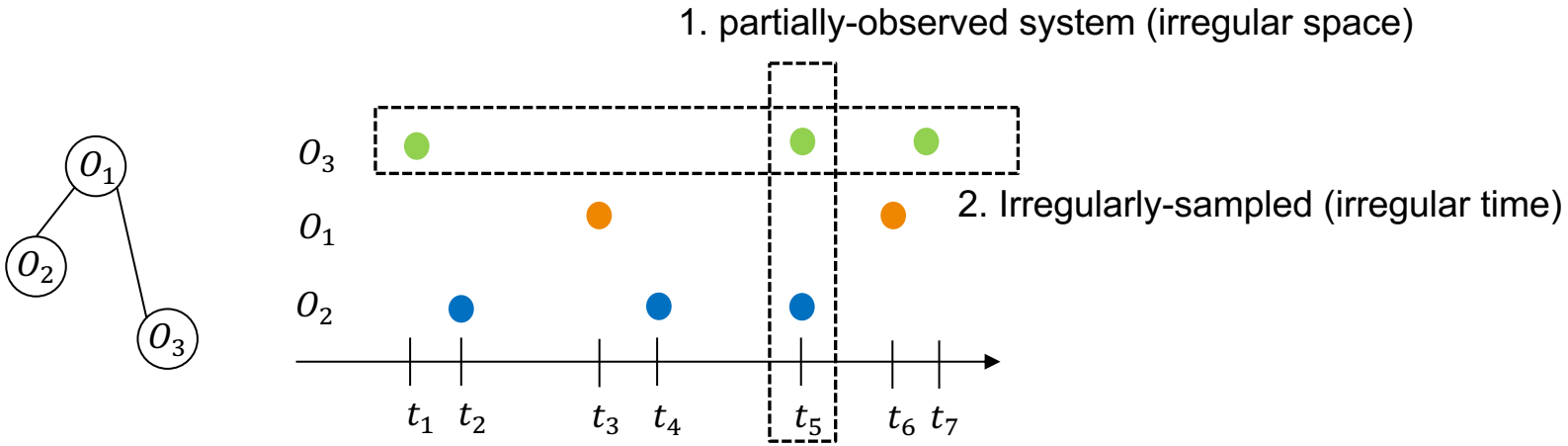




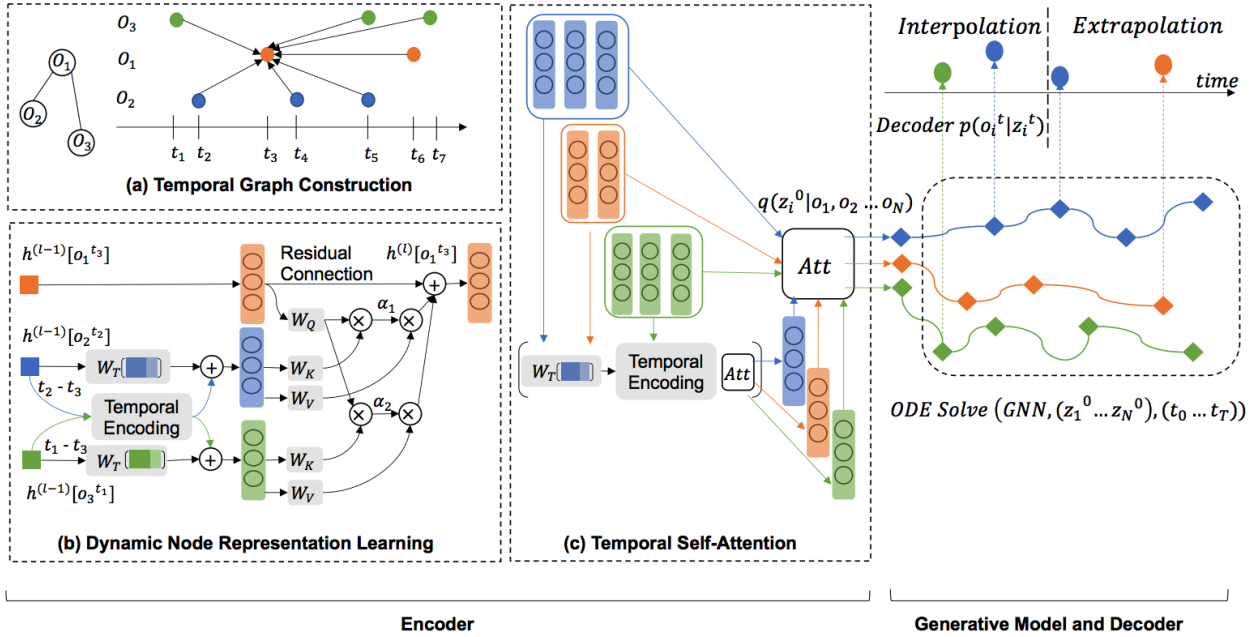
Challenges in Spatio-Temporal Modeling

- **Irregular space:** the observations for different agents are not temporally aligned , i.e. **partial** observable states.

- **Irregular time**: The observations of a specific object can happen at non-uniform intervals.



Our Solution : Latent Graph ODE (LG-ODE)



- Model latent dynamics using **neural ordinary differential equations** consider the mutual influence among agents.
- ODE function: GNN

$$\dot{\boldsymbol{z}}_i^t := \frac{d\boldsymbol{z}_i^t}{dt} = g_i(\boldsymbol{z}_1^t, \boldsymbol{z}_2^t \cdots \boldsymbol{z}_N^t)$$

- Infer the **initial states** for each agent **simultaneously** through a novel GNN module that can handle non-uniform data distribution.

$$\boldsymbol{z}_i^T = \boldsymbol{z}_i^0 + \int_{t=0}^{T} g_i(\boldsymbol{z}_1^t, \boldsymbol{z}_2^t \cdots \boldsymbol{z}_N^t) dt$$

VAE Loss

$$ELBO(\theta, \phi) = \mathbb{E}_{\boldsymbol{Z}^0 \sim q_\phi(\boldsymbol{Z}^0|o_1, \cdots o_N)}[\log p_\theta(o_1, \dots, o_N)] - \mathrm{KL}[q_\phi(\boldsymbol{Z}^0|o_1, \cdots, o_N) \| p(\boldsymbol{Z}^0)]$$

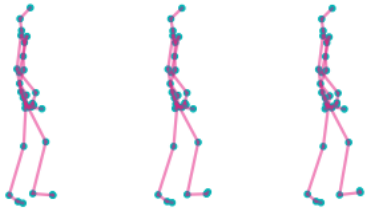
Learning Continuous System Dynamics from Irregularly-Sampled Partial Observations. Zijie Huang, Yizhou Sun, Wei Wang. Neurips 2020.

Evaluation

- Experiment on motion capture, spring system, and charged particle datasets on **interpolation** and **extrapolation** tasks over various observation ratios.

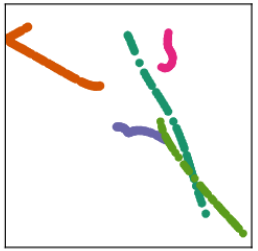
$$\mathbf{F}_{ij} = -C(\mathbf{x}_i - \mathbf{x}_j)$$

$$\mathbf{F}_{ij} = C \cdot \mathrm{sign}(q_i \cdot q_j) \frac{\mathbf{x}_i - \mathbf{x}_j}{||\mathbf{x}_i - \mathbf{x}_j||^3}$$



Motion Capture

Spring System

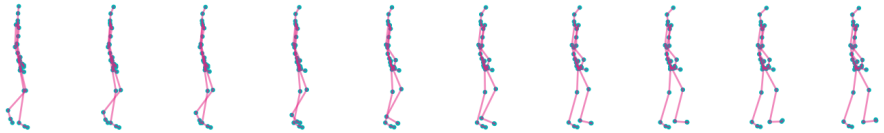


Charged Particles

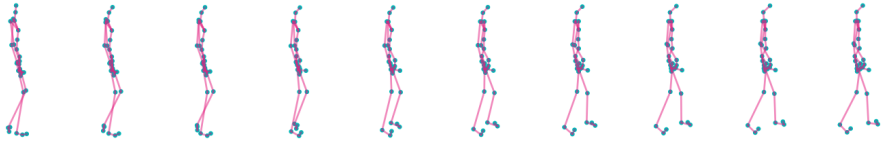
Evaluation - Interpolation Results

Table 1: Mean Squared Error(MSE) $\times 10^{-2}$ on Interpolation task.

	Springs			Charged			Motion		
Observed ratio	40%	60%	80%	40%	60%	80%	40%	60%	80%
Latent-ODE	0.5454	0.5036	0.4290	1.1799	1.1198	0.8332	0.7709	0.4826	0.3603
Weight-Decay	1.1634	1.1377	1.6217	2.8419	2.2547	1.5390	1.9007	2.0023	1.6894
Edge-GNN	1.3370	1.2786	0.8188	1.5795	1.5618	1.1420	2.7670	2.6582	1.8485
NRI + RNN	0.5225	0.4049	0.3548	1.3913	1.1659	1.0344	0.5845	0.5395	0.5204
LG-ODE	**0.3350**	**0.3170**	**0.2641**	**0.9234**	**0.8277**	**0.8046**	0.4515	**0.2870**	**0.3414**
LG-ODE-first	1.3017	1.1918	1.0796	2.5105	2.6714	2.3208	1.4904	1.3702	1.2107
LG-ODE-mean	0.3896	0.3901	0.3268	1.1246	1.0050	0.9133	0.6415	0.5834	0.5549
LG-ODE-no att	0.5145	0.4198	0.4510	0.9372	0.9503	0.9752	0.6991	0.6998	0.7452
LG-ODE-no PE	0.4431	0.4278	0.3879	1.0450	1.0350	0.9621	0.4677	0.4808	0.4799
LG-ODE-fixed PE	0.4285	0.4445	0.4083	0.9838	0.9775	0.9524	**0.4215**	0.4371	0.4313



(a) Groundtruth.



(b) Predictions with 0.6 observation ratio.

Evaluation – Extrapolation Results

Table 2: Mean Squared Error(MSE) $\times 10^{-2}$ on Extrapolation task.

Extrapolation	Springs			Charged			Motion		
Observed ratio	40%	60%	80%	40%	60%	80%	40%	60%	80%
Latent-ODE	6.6923	4.2478	4.3192	13.5852	12.7874	20.5501	2.4186	2.9061	2.6590
Weight-Decay	6.1559	5.7416	5.3712	9.4764	9.1008	9.0886	16.8031	13.6696	13.6796
Edge-GNN	6.0417	4.9220	3.2281	9.2124	9.1410	8.8341	13.2991	13.9676	9.8669
NRI + RNN	2.6638	2.4003	2.5550	7.1776	6.9882	6.6736	3.5380	3.0119	2.6006
LG-ODE	**1.7839**	1.8084	**1.7139**	6.5320	**6.4338**	**6.2448**	**1.2843**	**1.2435**	1.2010
LG-ODE-first	6.5742	6.3243	5.7788	9.3782	9.2107	8.4765	3.8864	3.2849	3.0001
LG-ODE-mean	2.2499	2.1165	2.2516	9.1355	8.7820	8.4422	1.3169	1.3008	1.2534
LG-ODE-no att	2.3847	2.1216	1.9634	7.2958	7.3609	6.7026	3.4510	3.2178	3.9917
LG-ODE-no PE	1.7943	1.8172	1.7332	6.9961	6.7208	6.5852	1.5054	1.2997	1.2029
LG-ODE-fixed PE	1.7905	**1.7634**	1.7545	**6.4520**	6.4706	6.3543	1.4624	1.2517	**1.1992**

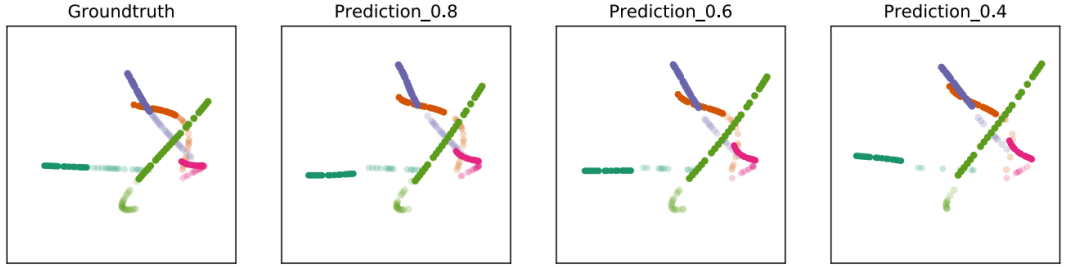
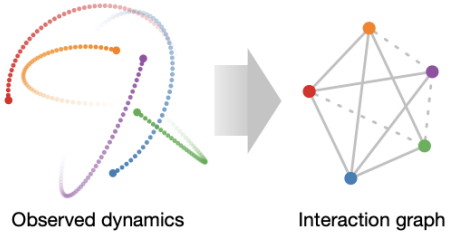


Figure 4: Visualization of extrapolation results for spring system. Semi-transparent paths denote observations from first-half of time, from which the latent initial states are estimated. Solid paths denote model predictions.

Challenges in Spatio-Temporal Modeling

- Model the complex interplay of spatial and temporal properties.
  - Graph structure may evolve over time along with node attributes and they will pose significant mutual influence to each other.
  - Learning with observable dynamic graph structure
  - Learning with unknown structural info – Infer the underlying graph structure first and then conduct reasoning over the co-evolution of nodes and edges [1].



[1] Thomas Kipf, etc. Neural Relational Inference for Interacting Systems. (ICML 2018)

Motivation Example

- Example: The spread of COVID-19 in the U.S.
  - Nodes: 50 states with dynamic daily confirmed cases, deaths, etc
  - Edges: Population flow among states.

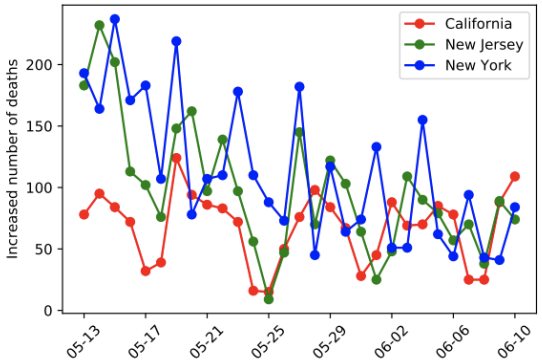
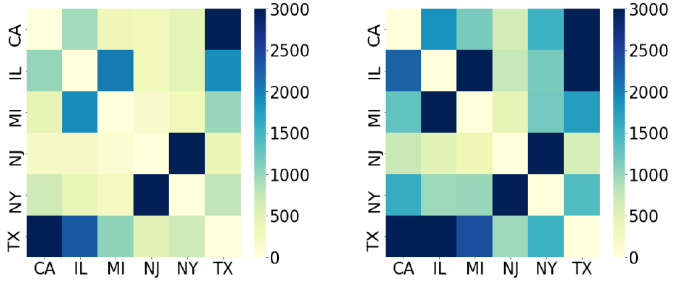


Figure 1: COVID-19 death count time series of three states in U.S. Correlation is higher between two states that have higher population flow.



(a) Population flow on May.11 (b) Population flow on Aug.8

Figure 2: Population flow in May and August with self-loop flow excluded (Diagonal entries). May has less population flow due to the "close border" policies in many states.

Coupled Graph ODE for Learning Interacting System Dynamics. **Zijie Huang**, Yizhou Sun, Wei Wang. KDD 2021.

Our Solution – Coupled Graph ODE (CG-ODE)

- Build a model that considers the **co-evolution** of nodes and edges in a **continuous** manner.
  - Why continuous? Many dynamical systems are **continuous** in nature. Existing discrete GNN-based methods may not well-capture the underlying dynamics for such continuous systems.
  - How to capture the continuous mutual influence between nodes and edges? -- Coupled ODE fucntions

Generative Model: Coupled-ODE

- **Key observation 1**: three driven factors for nodes from an epidemic modeling perspective

- **Key observation 2**: Edge evolution is both node-driven and self-driven.

Coupled-ODE for edges:

$$\frac{dz_{i\to j}^{t}}{dt} = f_e(z_i^t || z_j^t) + f_{self}(z_{i\to j}^t)$$

$$A_{ij}^t = f_{edge2value}(z_{i\to j}^t) \qquad \tilde{A}^t = D^{-1}A^t$$

Z^t ↑ ↓ $\tilde{A}^t$

Coupled-ODE for nodes:

$$\frac{dZ^t}{dt} = \sigma(\tilde{A}^t Z^t W) - Z^t + Z^0$$

Neighbor Aggregation — Natural Recovery — Natural Physique

Generative Model

Experiment Results for Prediction Performance

- Prediction at different lengths
- Evaluation metric: MAPE
- Baselines: discrete neural network-based methods; traditional statistical models

Table 1: Mean Absolute Percentage Error (MAPE) for Cumulative Deaths

Step Length	Pred Date	UCLA-SuEIR	UT-Mobility	Columbia	IHME	LSTM	NRI	VGRNN	CG-ODE
1-week -ahead	Nov.29-Dec.05	0.03297	0.02707	**0.02001**	-	0.08094	0.07784	0.06807	0.02144
	Dec.07-Dec.12	0.02283	0.03736	0.02455	0.02458	0.08363	0.07448	0.06086	0.02653
	Dec.14-Dec.19	0.01946	0.04178	**0.01443**	-	0.07144	0.06462	0.06102	0.01997
	Dec.21-Dec.26	0.01851	0.05460	0.02595	-	0.04912	0.04616	0.04297	**0.01849**
	Average	0.02344	0.04020	**0.02124**	0.02458	0.07128	0.06578	0.05823	0.02161
2-weeks -ahead	Nov.29-Dec.12	0.11036	0.07119	0.08194	-	0.15922	0.15004	0.13791	**0.04341**
	Dec.07-Dec.19	0.07951	0.05830	0.09248	0.06252	0.14873	0.13782	0.12812	**0.04702**
	Dec.14-Dec.26	0.06356	0.04112	0.05174	-	0.13012	0.11423	0.10712	**0.03709**
	Average	0.08448	0.05687	0.07539	0.06252	0.14602	0.13403	0.12438	**0.04251**
3-weeks -ahead	Nov.29-Dec.19	0.17361	0.13255	0.13721	-	0.11793	0.10752	0.10624	**0.04513**
	Dec.06-Dec.26	0.13116	0.09570	0.14445	0.10671	0.19561	0.18088	0.17322	**0.09832**
	Average	0.15239	0.11413	0.14083	0.10671	0.15677	0.14420	0.13973	**0.07173**

Case Study for the COVID-19 Dataset

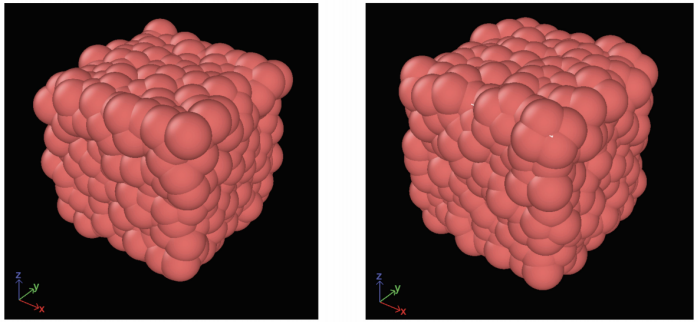
- Case study on mobility intervention
  - We set the duration of each intervention as 2 weeks and study the effect of adding the same intervention at different time.
  - Intervention 1: adds 20% reduction to all in-state population flows
  - Intervention 2: adds 20% reduction to all between-state population flows
  - Intervention 3: removes the same amount of population flow as in the second intervention, but in descending order of states' original population outflow

Table 3: Number of Deaths Reduced on Dec.26

	1-wk-ahead	2-wk-ahead	3-wk-ahead
In-state flow deduction (20%)	-8973	- 7084	-6824
Between-state flow deduction (20%)	-2465	-2215	-2197
Flow deducted from core states	-3854	-3625	-3517

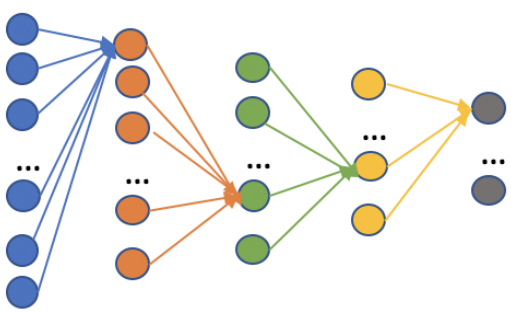
Future Work

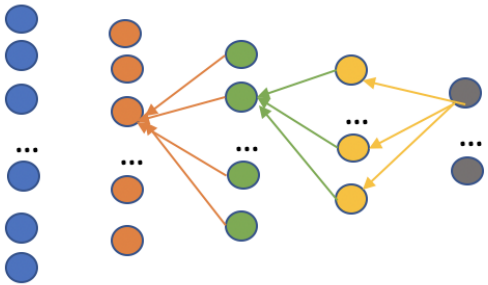
- Deep Learning for Accelerating Simulation
  - Explore weather we can build **faster** neural graph operations to approximate the “ground-truth” dynamics in order to predict the **large-scale** system evolution over **extremely large number of timestamps**, and can even be generalized to unseen system conditions.
  - Design a dynamic graph structure learning module to infer the underlying graph structure at each timestamp in an efficient way.



Future Work

- Accelerating Neural Network Training from a Dynamical System View
  - By viewing neuro as nodes and weight among neuros as edges, we can construct a dynamic graph where both nodes and edges features changes over each training iteration/epochs via forward and backward propagation respectively.
  - Design efficient spatio-temporal model that takes only the first several epochs of training data as input, to fast predict what would be the final results if the model converges





UCLA Samueli Computer Science

ScAi

Thanks for Listening!