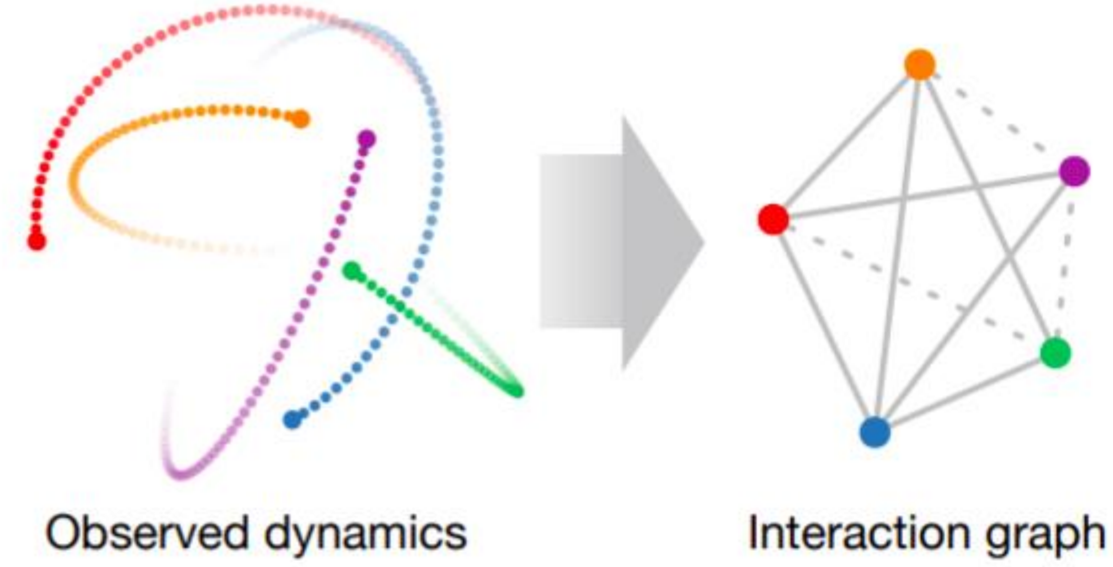
Neural Relational Inference for Interacting Systems

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, Richard Zemel

ICML, 2018

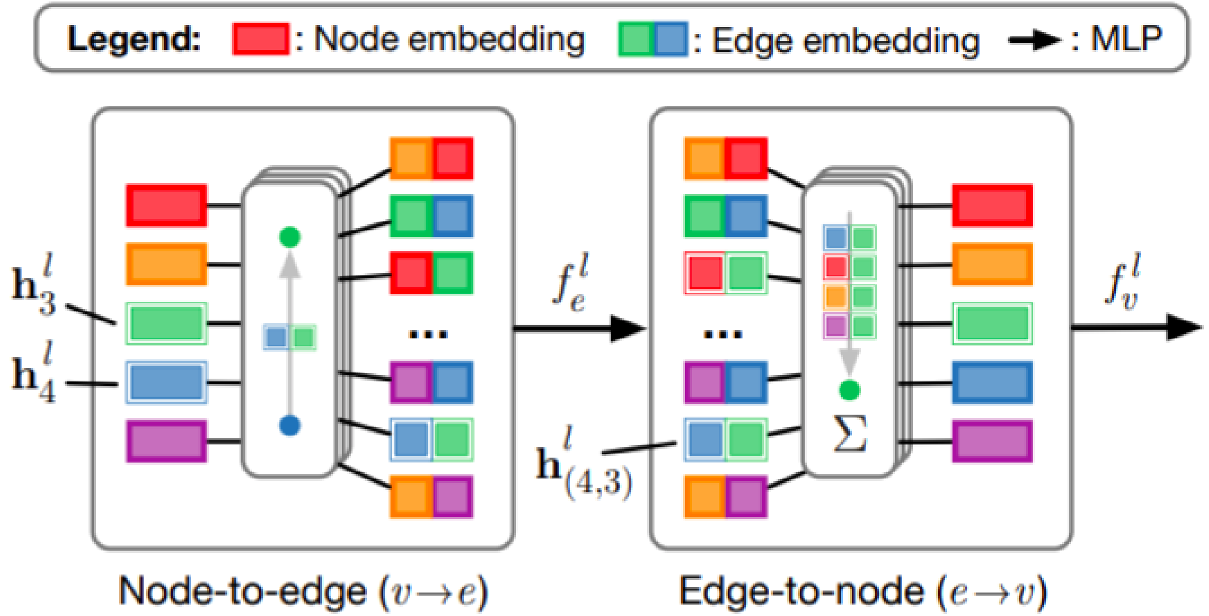
Task

- Given the trajectory of the particles, predict their future states using their implicit interactions
- Interactions are models as a graph where the particles are the vertices and edges represent K different type of interactions
- An auto-encoder is used to encode the interaction graph into a discrete random variable

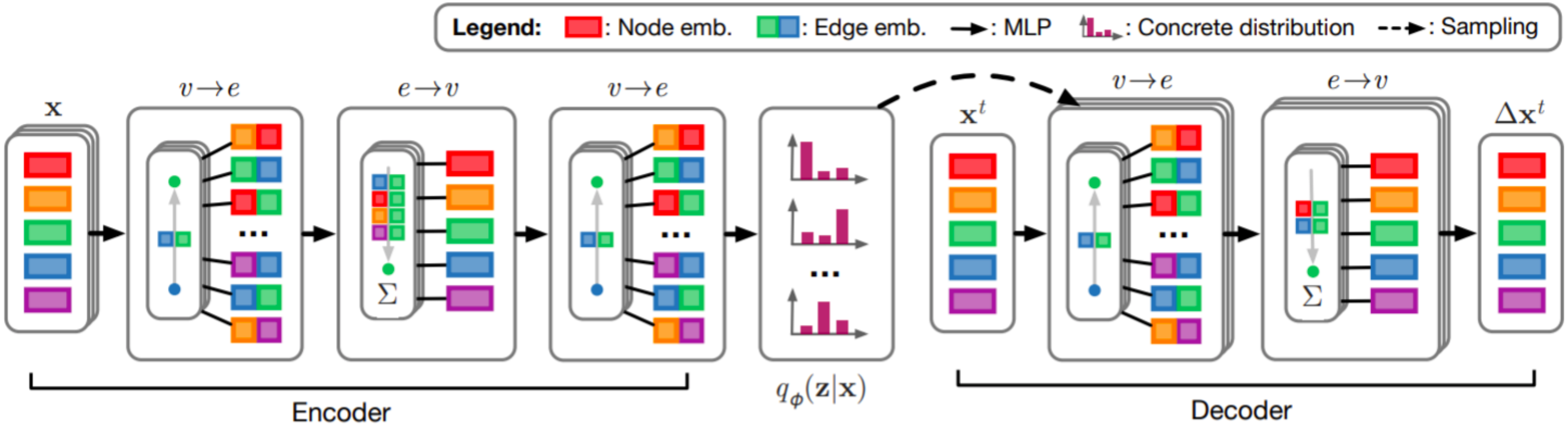


Background: Graph Neural Networks

$$v{\rightarrow}e: \quad \mathbf{h}^l_{(i,j)} = f^l_e([\mathbf{h}^l_i, \mathbf{h}^l_j, \mathbf{x}_{(i,j)}])$$
$$e{\rightarrow}v: \quad \mathbf{h}^{l+1}_j = f^l_v([\textstyle\sum_{i\in\mathcal{N}_j} \mathbf{h}^l_{(i,j)}, \mathbf{x}_j])$$



Model



$$\mathcal{L} = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathrm{KL}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})]$$

$$p_\theta(\mathbf{x}|\mathbf{z}) = \prod_{t=1}^{T} p_\theta(\mathbf{x}^{t+1}|\mathbf{x}^t, \ldots, \mathbf{x}^1, \mathbf{z})$$

$$p_\theta(\mathbf{z}) = \prod_{i \neq j} p_\theta(\mathbf{z}_{ij})$$

Encoder

- Encoding:

$$
\begin{aligned}
& \mathbf{h}_j^1 = f_{\mathrm{emb}}(\mathbf{x}_j) \\
v \to e: \quad & \mathbf{h}_{(i,j)}^1 = f_e^1([\mathbf{h}_i^1, \mathbf{h}_j^1]) \\
e \to v: \quad & \mathbf{h}_j^2 = f_v^1(\textstyle\sum_{i \neq j} \mathbf{h}_{(i,j)}^1) \\
v \to e: \quad & \mathbf{h}_{(i,j)}^2 = f_e^2([\mathbf{h}_i^2, \mathbf{h}_j^2])
\end{aligned}
$$

- Sampling:

$$
\mathbf{z}_{ij} = \mathrm{softmax}((\mathbf{h}_{(i,j)}^2 + \mathbf{g})/\tau)
$$

Decoder

- Decoding:

$$v \to e: \quad \tilde{\mathbf{h}}_{(i,j)}^t = \sum_k z_{ij,k} \tilde{f}_e^k([\mathbf{x}_i^t, \mathbf{x}_j^t])$$

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

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

- Multiple time steps:

$$\begin{aligned}
\boldsymbol{\mu}_j^2 &= f_{\text{dec}}(\mathbf{x}_j^1) \\
\boldsymbol{\mu}_j^{t+1} &= f_{\text{dec}}(\boldsymbol{\mu}_j^t) && t = 2, \ldots, M \\
\boldsymbol{\mu}_j^{M+2} &= f_{\text{dec}}(\mathbf{x}_j^{M+1}) \\
\boldsymbol{\mu}_j^{t+1} &= f_{\text{dec}}(\boldsymbol{\mu}_j^t) && t = M+2, \ldots, 2M \\
&\cdots
\end{aligned}$$

Training

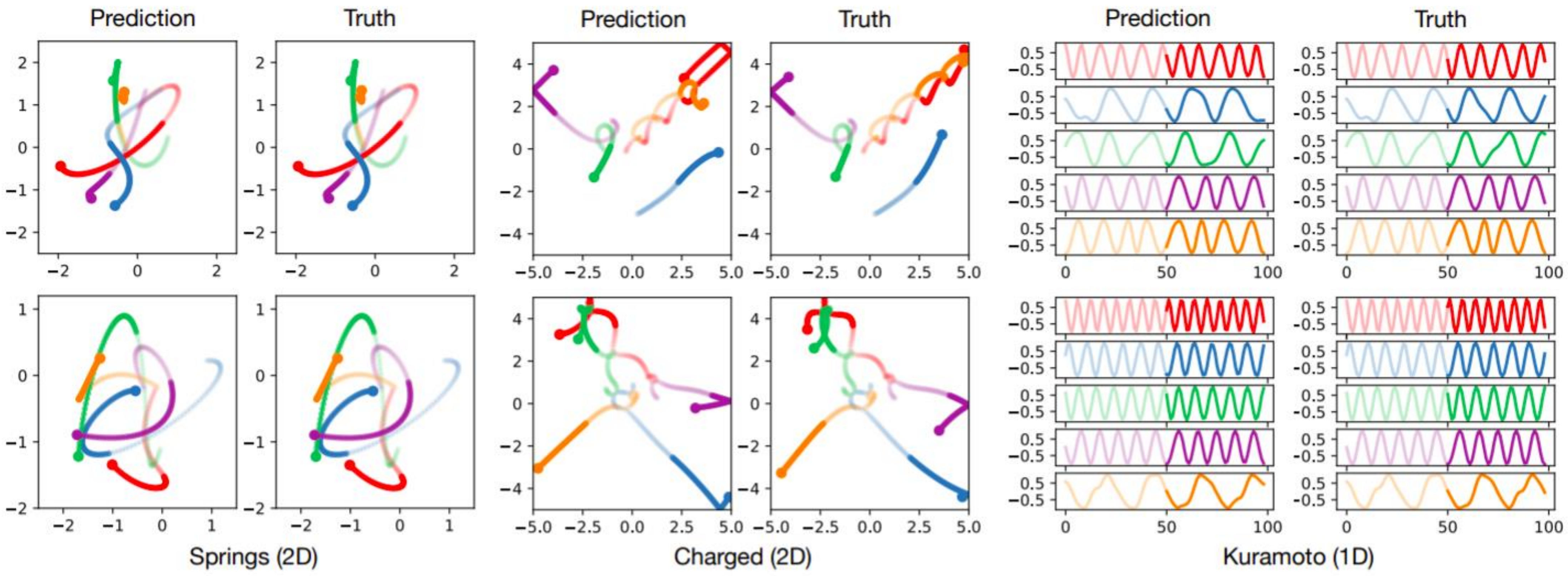
- Reconstruction:

$$-\sum_{j}\sum_{t=2}^{T}\frac{||\mathbf{x}_j^t-\boldsymbol{\mu}_j^t||^2}{2\sigma^2}+\text{const}$$

- KL divergence:

$$\sum_{i\neq j}H(q_\phi(\mathbf{z}_{ij}|\mathbf{x}))+\text{const.}$$

Results



Prediction
Truth
Prediction
Truth
Prediction
Truth
Springs (2D)
Charged (2D)
Kuramoto (1D)

Thanks