Learning the graph structure of interaction networks for inference of complex object-centric and relational dynamics

Swapnil Pande (swapnilp), Saumya Saxena (saumyas), Alvin Shek (ashek), Kevin Wang (kwang2)
Course: Probabilistic Graphical Models

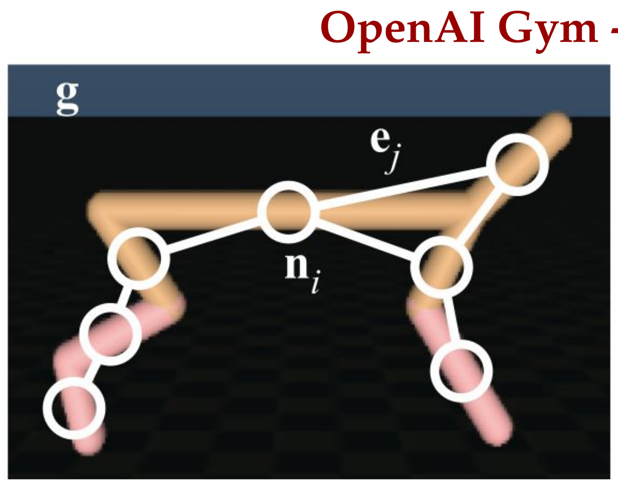
Motivation

- **Simulations are expensive to build and run**
  - Requires a lot of engineering effort and scaling up is computationally expensive
- **Difficult to model complex interactions**
  - Unknown physical parameters
  - Unknown dynamics equations and interactions
- **Prior knowledge of structure isn't available**
  - Applying wrong assumptions can cause model inaccuracies
    - Exploitation by Reinforcement Learning agents
- **Learned models enable model-based planning and control**
  - Learning dynamics requires only prior data

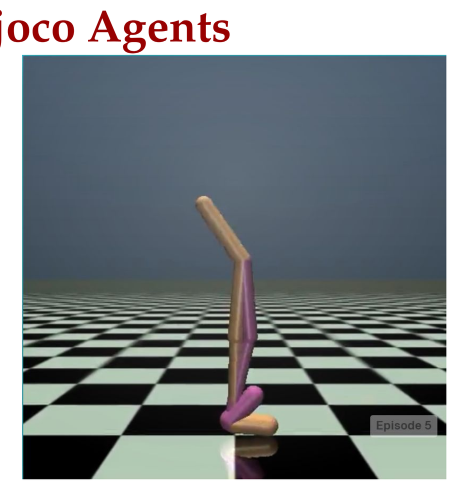
Dataset and Task

- **Input: Trajectory of node/joint states and actions**
  - Actions executed on Half Cheetah and Walker2D Mujoco models
- **Task: Infer graph structure, node and edge features**
  - Capture relationships, object properties and interactions
- **Output: Accurate dynamics model**
  - Future states given new actions
- **Metrics:**
  - **Quantitative: Mean Squared Error between predicted and true future states**
    - 1-step prediction
    - 5-step prediction
  - **Qualitative: Visualize graph structure on an agent**

OpenAI Gym - Mujoco Agents

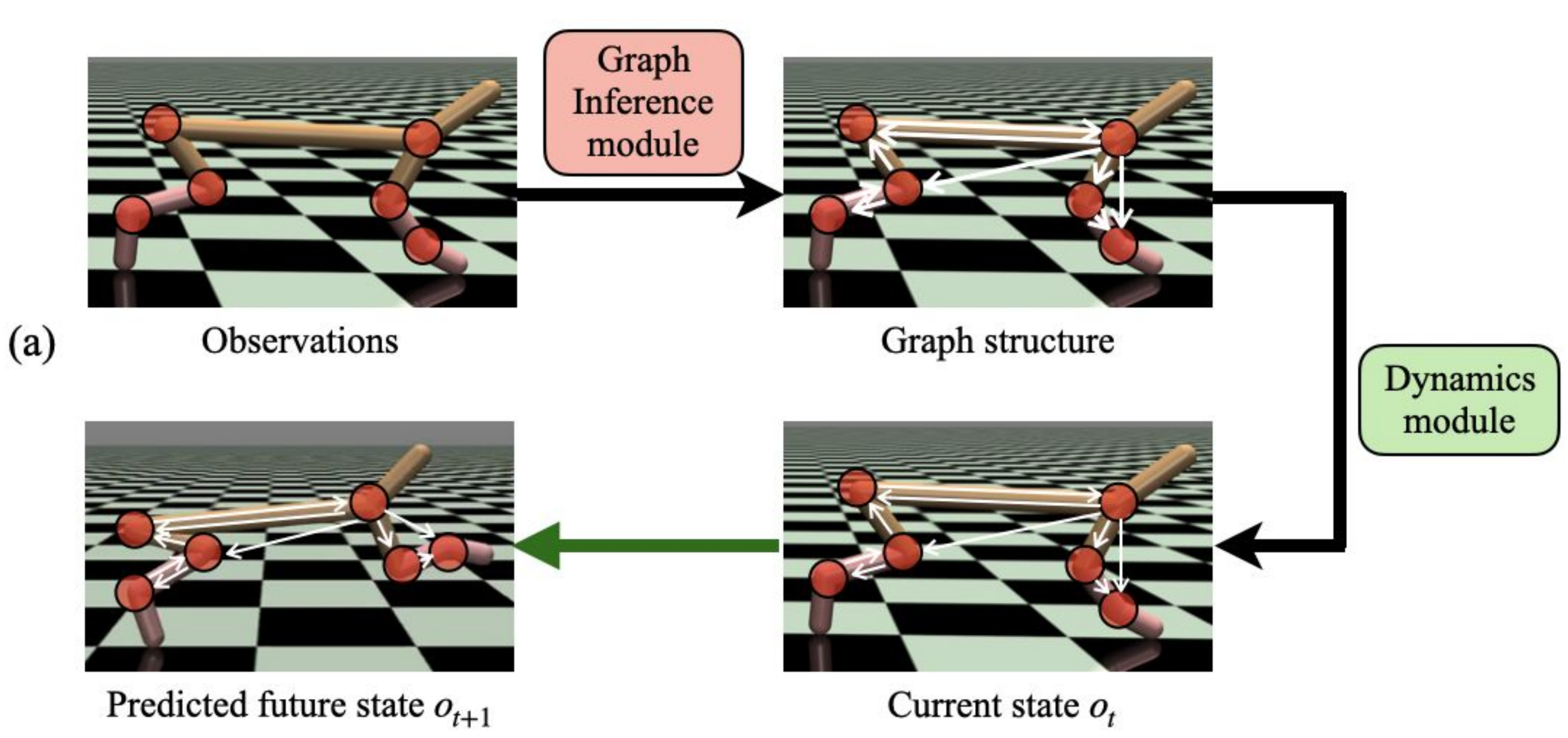


Agent 1: Half Cheetah

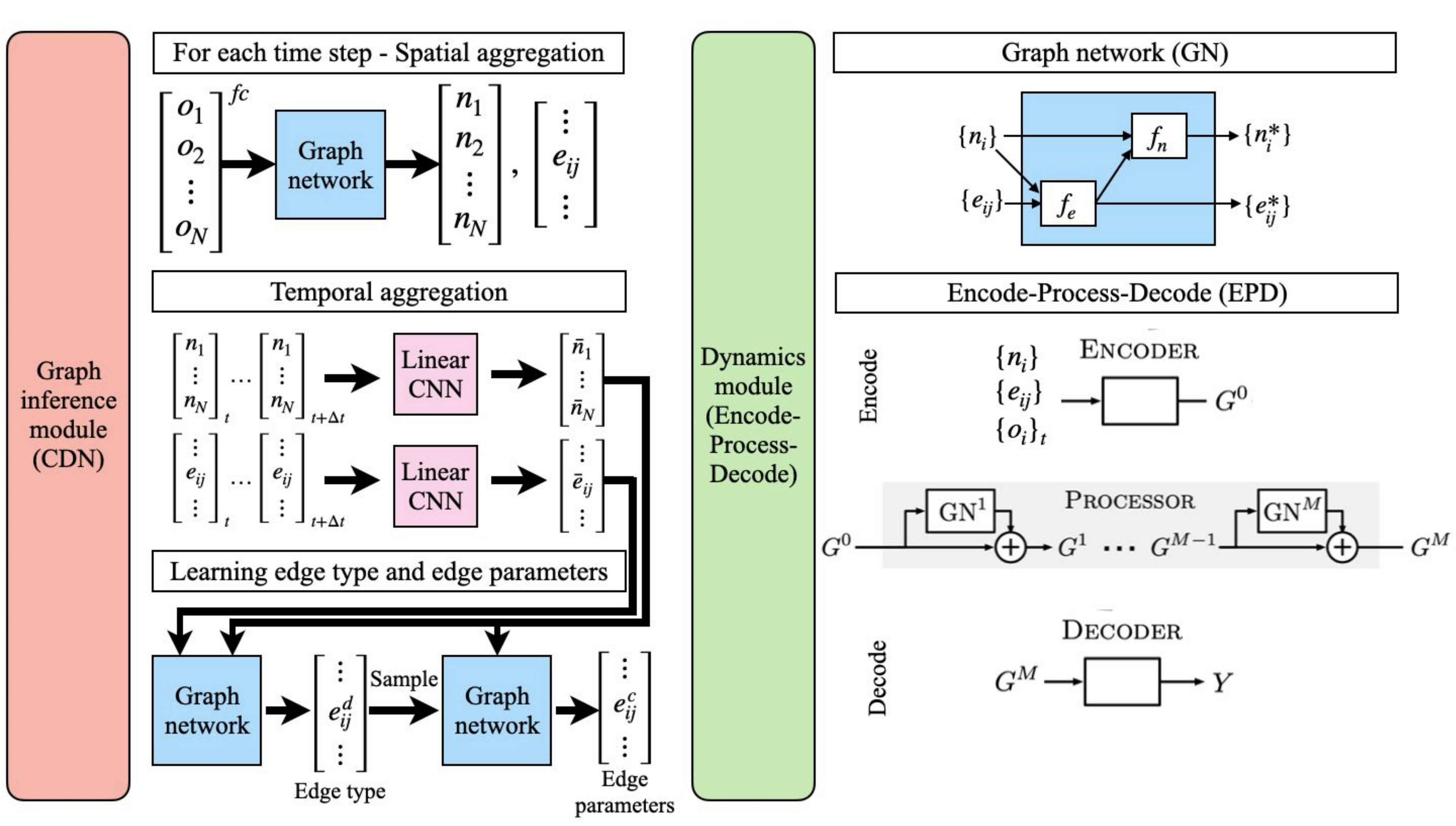


Agent 2: Walker2D

Approach



(a) Overall learning architecture of our proposed method



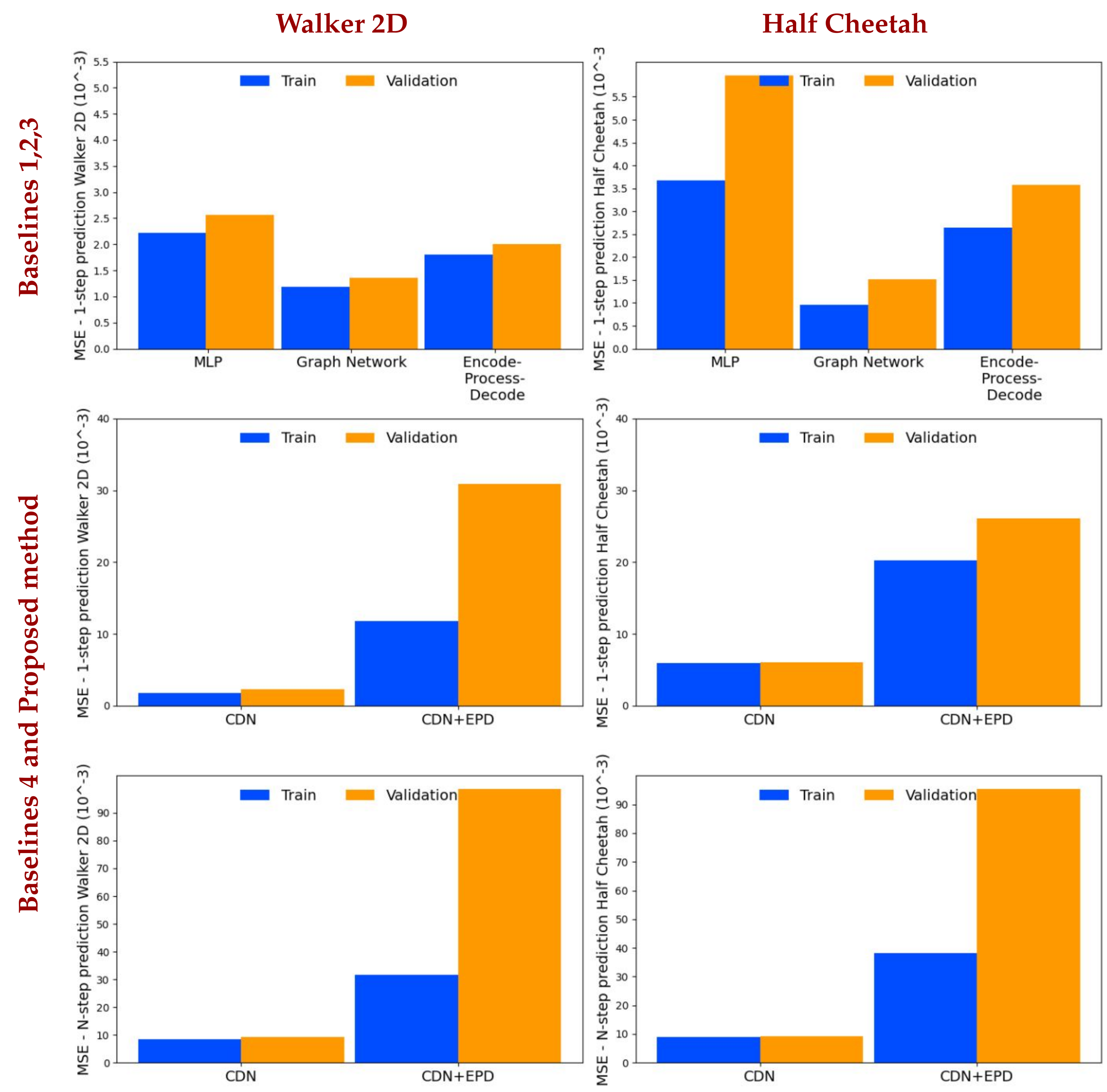
(b) The inference and dynamic modules that form the key elements of the learning architecture

- **Baselines**

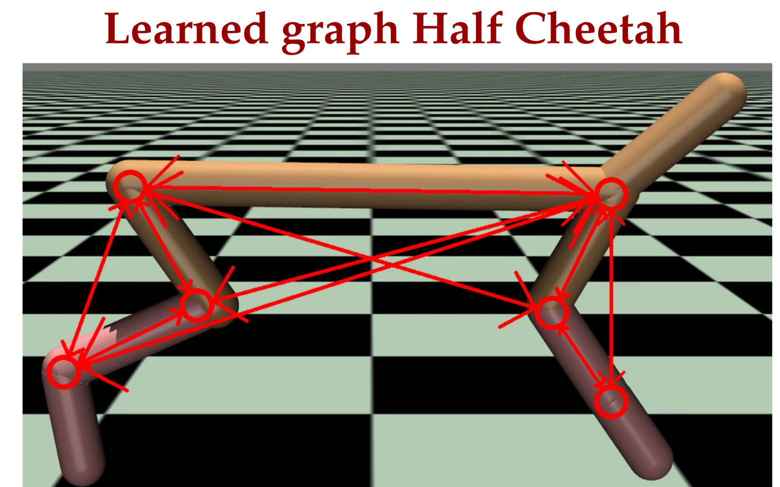
	Inference module	Dynamics module
Baseline 1	Graph known	MLP
Baseline 2	Graph known	Graph Network [1]
Baseline 3	Graph known	Encode-Process-Decode [3]
Baseline 4	Causal Discovery Network [4]	Graph Network [1]
Proposed method	Causal Discovery Network [4]	Encode-Process-Decode [3]

Experiments and Results

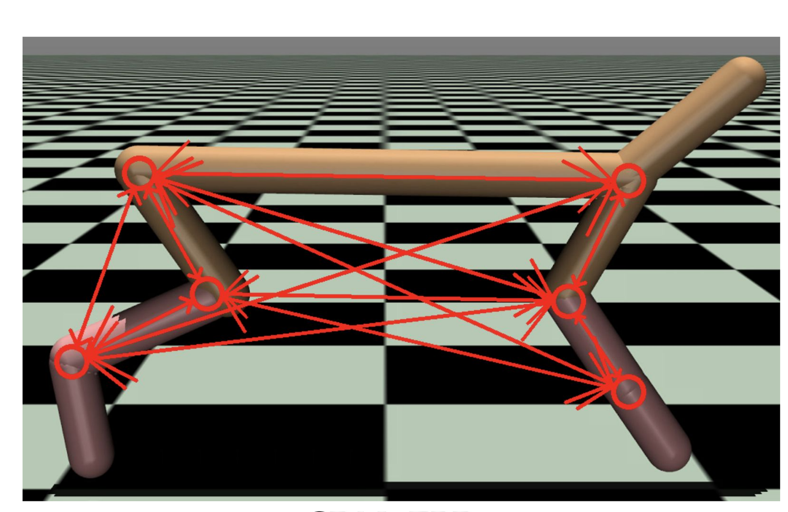
- **Comparison of baselines with proposed method**
  - **Environments:** Half-Cheetah and Walker2D MuJoCo environments
  - **Metrics -** 1-step dynamics prediction, N-step dynamics prediction
  - **Learned graph** using the inference module



Learned graph Half Cheetah

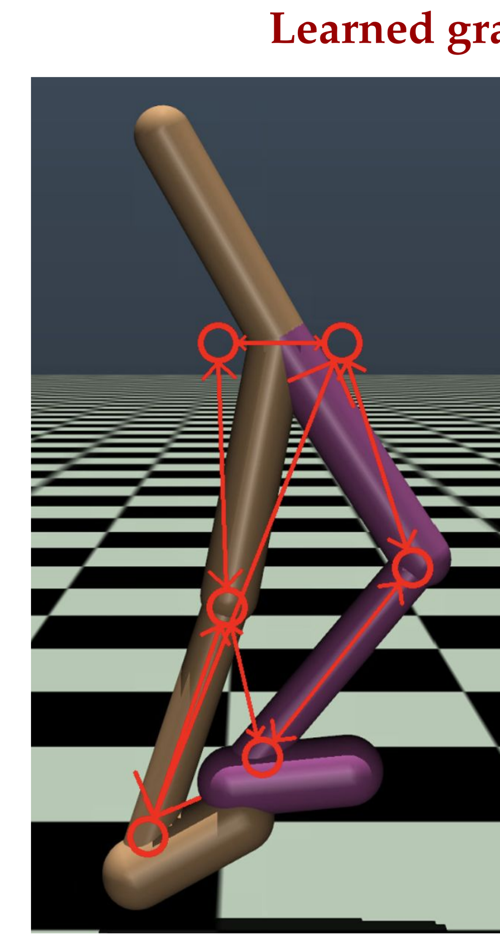


Baseline 4

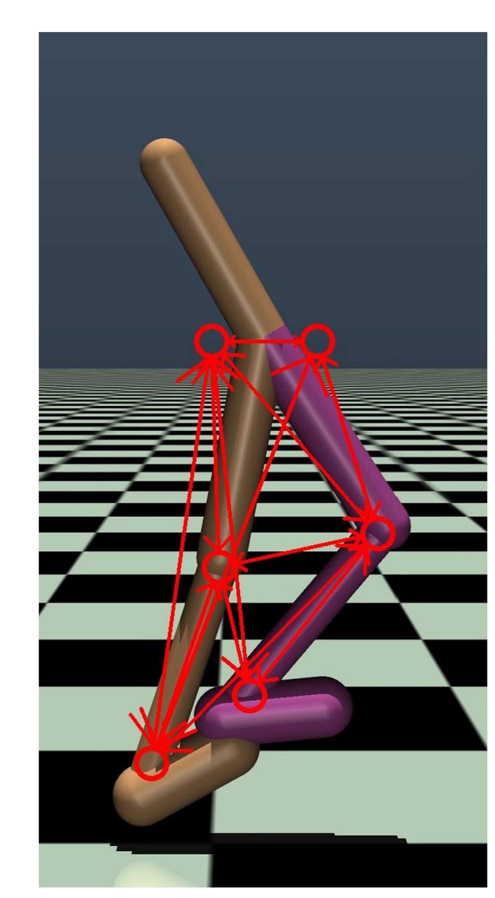


CDN+EPD

Learned graph Walker 2D



Baseline 4



CDN+EPD

- **Discussion of above plots**
  - **Baselines 1,2,3 -** Results with known graph architectures
    - **Interaction Networks perform the best** with the given graph architecture
    - EPD does not provide us with the improvement in performance that we expected
      - More complicated model. Many hyperparameters to tune.
        - Number of message passing steps
        - Depth of encoder and decoder NN
  - **Baseline 4 and Proposed methods**
    - Baseline 4 performs better than our proposed architecture
    - The graphs learned using baseline 4 are sparser than the graphs learned using CDN+EPD

Related Work

1. **Graph Networks:** Sanchez-Gonzalez, A., Heess, N., Springenberg, J.T., Merel, J., Riedmiller, M., Hadsell, R. and Battaglia, P., 2018, July. Graph networks as learnable physics engines for inference and control.
   - Applies graph networks on MuJoCo environments for dynamics predictions.
2. **Interaction Networks:** Battaglia, P.W., Pascanu, R., Lai, M., Rezende, D. and Kavukcuoglu, K., 2016. Interaction networks for learning about objects, relations and physics.
   - Used interaction networks to predict future states and estimate potential energy in n-body systems (i.e. balls bouncing in a box)
3. **Encode-Process-Decode:** Sanchez-Gonzalez, A., Godwin, J., Pfaff, T., Ying, R., Leskovec, J. and Battaglia, P., 2020, November. Learning to simulate complex physics with graph networks
   - Uses graph networks to simulate complex particle-based systems
4. **V-CDN:** Li, Y., Torralba, A., Anandkumar, A., Fox, D. and Garg, A., 2020. Causal discovery in physical systems from videos.
   - Proposed the Visual Causal Discovery Network (V-CDN) to estimate causal structures in a system based on keypoints