*Proceedings of the 33rd European Safety and Reliability Conference (ESREL 2023)*
*Edited by* Mário P. Brito, Terje Aven, Piero Baraldi, Marko Čepin and Enrico Zio
 *Published by* Research Publishing, Singapore.
doi: 10.3850/978-981-18-8071-1_P508-cd

# Learning dynamics of spring-mass models with physics-informed graph neural networks

Vinay Sharma
*Laboratory of Intelligent Maintenance and Operation systems, Switzerland. E-mail: vinay.sharma@epfl.ch*

Manav Manav
*Computational Mechanics Group. E-mail: mmanav@ethz.ch*

Laura De Lorenzis
*Computational Mechanics Group. E-mail: ldelorenzis@ethz.ch*

Olga Fink
*Laboratory of Intelligent Maintenance and Operation systems, Switzerland. E-mail: olga.fink@epfl.ch*

We propose a physics-informed message-passing graph neural network (GNN) for learning the dynamics of spring-mass systems. The proposed method embeds the underlying physics directly into the message-passing scheme of the GNN. We compare the new scheme with conventional message passing and demonstrate the generalization capability of the method. Additionally, we infer the learned parameters of the edges and show that these parameters serve as explainable metrics for the learned physics. The numerical results indicate that the proposed method accurately learns the physics of the spring-mass systems.

*Keywords*: Graph networks, Physics simulations, coupled-spring mass systems.

## 1. Introduction

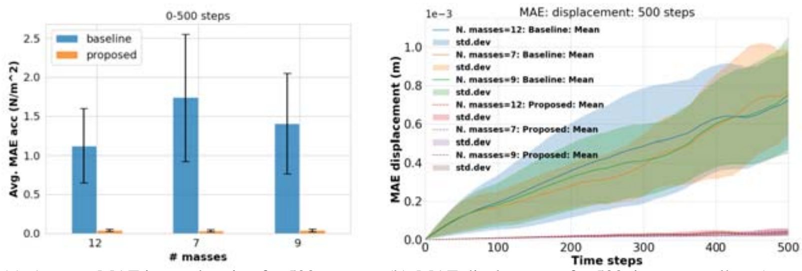
Previous research has shown that GNNs can efficiently predict pairwise interactive dynamics between discrete masses when represented as graphs Battaglia and et. al. [2018] Pfaff and et al. [2021]. This is due to the inductive bias of GNNs, which models pairwise interactions as messages passed between the edges of the graph. Recently, a new type of GNNs called E(n) - Equivariant Graph Neural Networks - have been proposed Satorras and et al. [2021]. The goal of these networks is to enhance the inductive bias by preserving the rotation and translation symmetries in Euclidean space. These networks operate on Euclidean graphs where nodes and edges are assigned positions and relative distance vectors, respectively, in addition to their corresponding features. Building upon the aforementioned research, our study presents a novel message-passing scheme that integrates physical inductive biases to simulate the dynamics of spring-mass systems. Similar to the prior work, our method operates on graphs defined in Euclidean space.

## 2. Physics-informed GNN

We propose a GNN framework for spring-mass systems, where learned edge messages operate on embeddings of connected nodes.

### 2.1. *Physics-informed message passing*

Our message passing is based on the following ingredients. **(1)** Edge message $m_{ij}$ is constructed as $m_{ij} = \phi_e(||\vec{x}_{ij}||^2)$, where $\phi_e$ is a learned function and $\vec{x}_{ij} = \vec{x}_i - \vec{x}_j$ is the edge vector. Our formulation results in $m_{ij} = m_{ji}$. We consider $m_{ij}$ to be a hidden representation of the magnitude of the spring force. **(2)** At each node incoming edge vectors are weighted and summed to estimate a latent internal force on nodes, where weights are determined by $\phi_w$ with $m_{ij}$ as inputs i.e. $\vec{f}_i^l = \Sigma_{in}\vec{x}_{ij}\phi_w(m_{ij})$. This formulation ensures that the force from node $i$ to $j$ is equal and opposite to that from $j$ to $i$. Then, two functions $\phi_{n\to s}$ and $\phi_{n\to v}$ transform the node embeddings to a scalar $\alpha$ and a vector $\vec{f}^l_{ext}$, respectively. Next, the latent node acceleration is calculated as $\ddot{\vec{x}}^l = \alpha\vec{f}_i^l + \vec{f}^l_{ext}$.




(a) Average MAE in acceleration for 500 steps

(b) MAE displacement for 500 time-step rollout (averaged over all the nodes)

**Fig. 1.:** Comparison with baseline GNN

**(3)** The updated edge vector $\vec{x}_{ij}^{upd}$ is calculated by adding the parallel and orthogonal projections of $\ddot{\vec{x}}^{l}$ onto $\vec{x}_{ij}$. Next, the edge message is updated as $m_{ij}^{upd} = \phi_{upd}(||\vec{x}_{ij}^{upd}||, m_{ij})$. **(4)** Finally all the latent node accelerations are aggregated after $n$ message passing steps to predict the acceleration on the node.

## 2.2. *Inference of edge parameters*

Edge parameters are inferred by calculating the eigenvalues of the Jacobian matrix of node acceleration $\ddot{\vec{x}}_i$ with respect to the edge vector $\vec{x}_{ij}$. For a spring with stiffness $K_{ij}$ and rest length $\ell_{0ij}$, the eigenvalues are given by:

$$eig_1 = \frac{K_{ij}(||\vec{x}_{ij}|| - \ell_{0ij})}{||\vec{x}_{ij}||} \quad \text{and} \quad eig_2 = K_{ij}.$$

## 3. Results

To evaluate our method, we predict the trajectory roll-out of various configurations of 1kg masses connected by springs with 50N/m stiffness and 4m rest length. The GNN model proposed in Pfaff and et al. [2021] is used as baseline. Our model is trained on simulated 1000 time-step noisy trajectories of configurations with 4, 5, 6 and 8 masses. The performance of trained model was evaluated on the configurations of 7, 9 and 12 masses. Notably, the latter two configurations represent extrapolation scenarios beyond the training data. Our model achieves excellent performance in predicting the dynamics of the spring-mass systems, as well as inferring the underlying physics. The proposed method generates stable roll-outs, as shown in Figure 1 and Table 1. Table 2 shows the inferred $K$ and $l_0$ parameters for each edge in a configuration of 9 masses at the $100^{th}$ and $250^{th}$ steps.

**Table 1.:** Avg MAE acceleration 0-500

| mean ± std. | **baseline** | **proposed** |
|---|---|---|
| **m:12 $\ell_0$:4 K:50** | 1.286 ± 0.207 | 0.125 ± 0.096 |
| **m:3 $\ell_0$:4 K:50** | 1.193 ± 0.637 | 0.057 ± 0.037 |
| **m:7 $\ell_0$:4 K:50** | 1.445 ± 0.272 | 0.146 ± 0.124 |
| **m:9 $\ell_0$:4 K:50** | 1.590 ± 0.222 | 0.137 ± 0.111 |

**Table 2.:** Inferred spring stiffness and rest length for 9-mass configuration

| Edge | Mean ± Std.dev. | | | |
|---|---|---|---|---|
| | **step = 100** | | **step = 250** | |
| | K | $\ell_0$ | K | $\ell_0$ |
| 0→1 | 48.64 ± 0.74 | 4 ± 0 | 50.32 ± 0.60 | 4.4 ± 0.01 |
| 2→1 | 48.52 ± 0.62 | 4 ± 0.01 | 49.90 ± 0.54 | 4.2 ± 0 |
| 1→2 | 50.14 ± 2.35 | 4 ± 0 | 49.81 ± 0.40 | 4.1 ± 0 |
| 3→2 | 48.46 ± 0.69 | 4 ± 0 | 48.76 ± 0.98 | 4 ± 0.01 |
| 2→3 | 50.22 ± 2.41 | 4 ± 0 | 48.71 ± 0.96 | 4 ± 0.01 |
| 4→3 | 50.16 ± 2.31 | 4 ± 0 | 49.72 ± 0.33 | 4.1 ± 0.01 |
| 3→4 | 48.4 ± 0.64 | 4 ± 0.01 | 49.84 ± 0.42 | 4.2 ± 0 |
| 5→4 | 48.41 ± 0.67 | 4 ± 0.01 | 49.84 ± 0.44 | 4.2 ± 0 |
| 4→5 | 50.13 ± 2.35 | 4 ± 0 | 49.7 ± 0.41 | 4.1 ± 0 |
| 6→5 | 48.46 ± 0.69 | 4 ± 0 | 48.69 ± 0.96 | 4 ± 0.01 |
| 5→6 | 50.17 ± 2.32 | 4 ± 0 | 48.78 ± 0.98 | 4 ± 0.01 |
| 7→6 | 50.21 ± 2.40 | 4 ± 0 | 49.82 ± 0.34 | 4.1± 0.01 |
| 6→7 | 48.52 ± 0.60 | 4 ± 0.01 | 49.9 ± 0.53 | 4.2 ± 0 |
| 8→7 | 48.57 ± 0.71 | 4 ± 0.01 | 50.26 ± 0.50 | 4.4 ± 0.01 |

## References

Peter W. Battaglia and et. al. Relational inductive biases, deep learning, and graph networks, 2018.

Tobias Pfaff and et al. Learning mesh-based simulation with graph networks. In *International Conference on Learning Representations*, 2021.

Satorras and et al. E (n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.