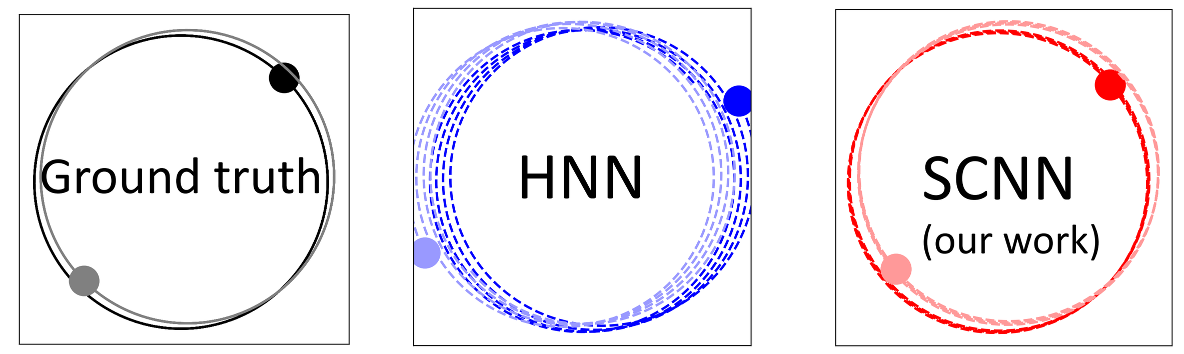
Improving Simulations with Symmetry Control Neural Networks

ICLR 2021 Workshop
Deep Learning for Simulation (simDL)

Marc Syväri, University of Munich (LMU) & MPP Munich Sven, Krippendorf, University of Munich (LMU)

The dynamics of physical systems is often constrained to lower dimensional sub-spaces due to the presence of conserved quantities. Here we propose a method to learn and exploit such symmetry constraints building upon Hamiltonian Neural Networks. By enforcing cyclic coordinates with appropriate loss functions, we find that we can achieve improved accuracy on simple classical dynamics tasks. By fitting analytic formulae to the latent variables in our network we recover that our networks are utilizing conserved quantities such as (angular) momentum.



Sample trajectories
Gravitational 2-body

Main idea: using and exploiting conserved quantities to constrain motion of particles

- Scheme: Predicting motion of particles using physical bias (HNN and additional conserved quantities)
- Motion governed by Hamiltonian
- Hamiltonian Neural Network restricts motions to hyperplanes (in phase space) of constant energy
- Additional constraints due to other (unknown) conserved quantities (e.g. angular momentum)
- Using constraints from conserved quantities to improve the Hamiltonian
- Side effect: reveal symmetries (see also [Krippendorf, Syväri 2020])

Theory of SCNN

- **Hamiltonian Neural Network approach** [Greydanus et al. 2019]: learn motion of particles by learning Hamiltonian; using Hamiltonian equations to compute time evolution (physics bias):

$$\frac{d\mathbf{q}}{dt} = \frac{\partial \mathcal{H}}{d\mathbf{p}} = \{\mathbf{q}, \mathcal{H}\}\,, \qquad \frac{d\mathbf{p}}{dt} = -\frac{\partial \mathcal{H}}{d\mathbf{q}} = \{\mathbf{p}, \mathcal{H}\}$$

- **Problem:** Rather unstable after longer time period, fails to conserve conserved quantities
- **Solution**: additional physics bias, use coordinate transformation to generalized coordinates $(\mathbf{P}, \mathbf{Q})$

$$T_\psi : (\mathbf{q}, \mathbf{p}) \mapsto (\mathbf{Q}(\mathbf{q}, \mathbf{p}), \mathbf{P}(\mathbf{q}, \mathbf{p}))$$
$$\mathcal{H}_\phi(\mathbf{p}, \mathbf{q}) = \tilde{\mathcal{H}}_\phi(\mathbf{P}(\mathbf{p}, \mathbf{q}), \mathbf{Q}(\mathbf{p}, \mathbf{q}))$$

T_ψ network for coordinate transformation, $\tilde{\mathcal{H}}_\phi$ for Hamiltonian

- T_ψ can be flexible (learning conserved quantities) or known conserved quantities can be enforced)

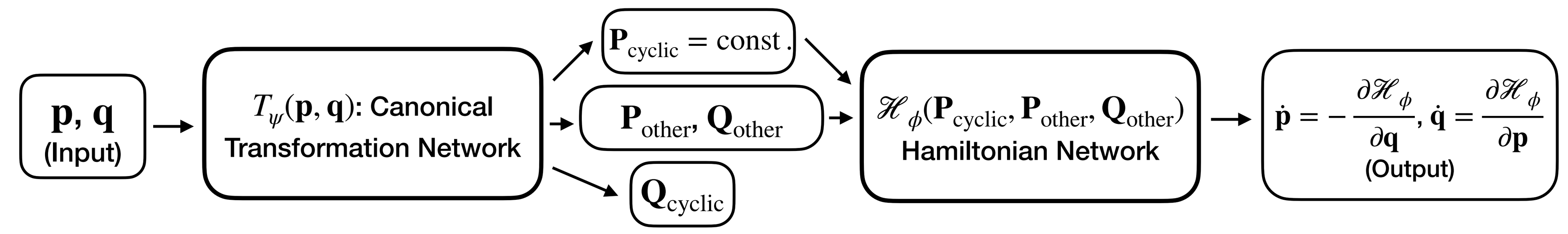
How do additional constraints on the Hamiltonian arise from this bias?

- Conservation of generalized momenta, constrains the structure of the Hamiltonian

$$0 = \dot{P}_i = -\frac{\partial \mathcal{H}}{\partial Q_i} = \{P_i, \mathcal{H}\}$$

- Different coordinates have to satisfy Poisson algebra

$$\{Q_i, P_j\} = \delta_{ij}, \{Q_i, Q_j\} = \{P_i, P_j\} = 0$$



Architecture of the neural network

- Both neural networks consists of two hidden layer with 200 neurons each and tanh-activation
- Adam optimizer with learning rate of 10^{-3}.

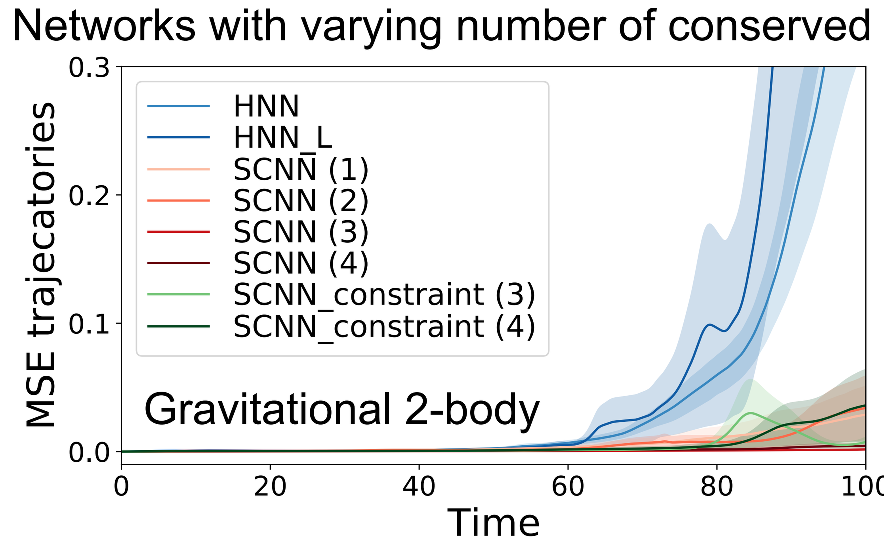
Experiments

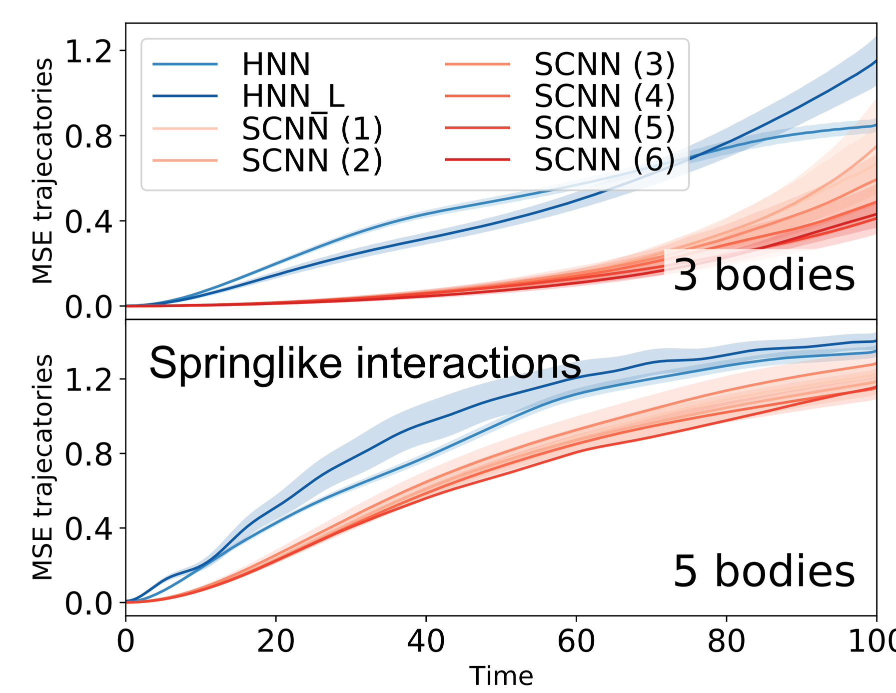
Does SCNN with and without domain knowledge perform better than HNN?

1) SCNN (k): SCNN with k constrained generalized coordinates
2) SCNN_constraint (k): SCNN with k known conserved quantities (using domain knowledge)

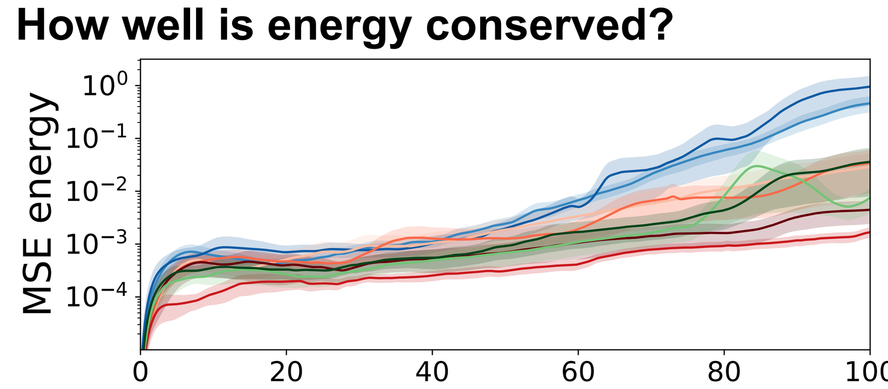
How good are the trajectories?

Networks with varying number of conserved quantities

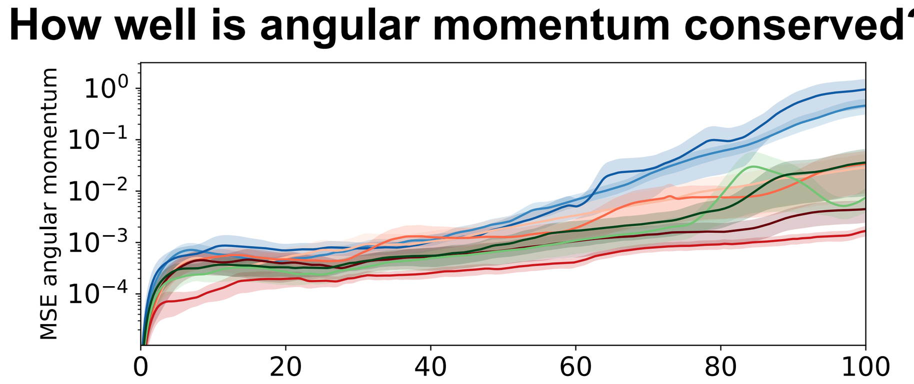




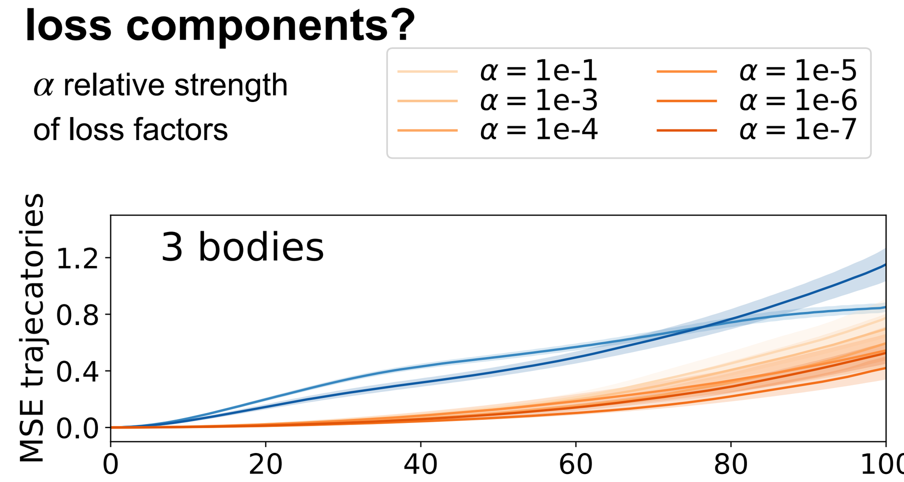
How well is energy conserved?



How well is angular momentum conserved?



What is the dependence on the additional loss components?



Increased performance of SCNN also on further physical systems: charged particle in magnetic field, spherical pendulum and double pendulum

Experiment	SCNN	SCNN-constraint	HNN
Magnetic field	0.164 ± 0.025	$\mathbf{0.033 \pm 0.010}$	0.083 ± 0.019
Spherical pendulum	0.092 ± 0.33	$\mathbf{0.055 \pm 0.005}$	0.288 ± 0.007
Double pendulum	$\mathbf{0.014 \pm 0.004}$	–	0.117 ± 0.012

Neural Network and loss components

- Hamiltonian loss:

$$\mathcal{L}_{\text{HNN}} = \sum_{i=1}^{N\cdot d} \left\| \frac{\partial \mathcal{H}_\phi(\mathbf{P}, \mathbf{Q})}{\partial p_i} - \frac{dq_i}{dt} \right\|_2 + \left\| \frac{\partial \mathcal{H}_\phi(\mathbf{P}, \mathbf{Q})}{\partial q_i} + \frac{dp_i}{dt} \right\|_2$$

- Poisson algebra loss:

$$\mathcal{L}_{\text{Poisson}} = \sum_{i,j=1}^{n\cdot d} \left\| \{Q_i, P_j\} - \delta_{ij} \right\|_2 + \sum_{i,j>i}^{n\cdot d} \left\| \{P_i, P_j\} \right\|_2 + \left\| \{Q_i, Q_j\} \right\|_2$$

- Loss on cyclic coordinates:

$$\mathcal{L}^{(n)}_{\text{HQP}} = \sum_{i=1}^{n} \left\| \frac{dP_i}{dt} \right\|_2 + \left\| \frac{dQ_i}{dt} - \frac{\partial \mathcal{H}_\phi(\mathbf{P}, \mathbf{Q})}{\partial P_i} \right\|_2$$

- Additional loss components force the neural network to find conserved quantities and use them to constrain Hamiltonian
- Complexity: Inference is comparable to HNN

Analytical formulae for conserved quantities

- Fitting of generalized coordinates (and specially conserved quantities) possible
- Improvement in prediction precision and computing efficiency
- Conserved quantities in the gravitional two body system (example):

$$\begin{aligned} P_{c_1} &= -4.2\, p_{x_1} - 4.2\, p_{x_2} - 1.3\, p_{y_1} - 1.3\, p_{y_2}\,, \\ P_{c_2} &= -0.9\, p_{x_1} - 0.9\, p_{x_2} - 3.2\, p_{y_1} - 3.2\, p_{y_2}, \\ L &= 1.0\, q_{x_1}p_{y_1} + 0.9\, q_{x_1}p_{y_2} + 0.9\, q_{x_2}p_{y_1} - 1.0\, q_{x_2}p_{y_2} \\ &\quad + 1.0\, q_{y_1}p_{x_1} - 0.9\, q_{y_1}p_{x_2} - 0.9\, q_{y_2}p_{x_1} + 1.0\, q_{y_2}p_{x_2}. \end{aligned}$$

Outlook and related work

- In the domain of the work of [Battaglia et al.]
- As for HNN, natural extensions to Graph Neural Network [Sanchez-Gonzalez et al., 2019] and in flows [Toth et al., 2019]
- Applications within molecular dynamics and astrophysical simulations

References

- Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. Neurips 2016.
- Sam Greydanus, Misko Dzamba, and Jason Yosinski. Hamiltonian neural networks. Neurips 2019, arXiv:1906.01563.
- S. Krippendorf and M. Syväri: Detecting Symmetries with Neural Networks, Machine Learning: Science and Technology 2 (2020) 015010, arXiv:2003.13679.
- Alvaro Sanchez-Gonzalez, Victor Bapst, Kyle Cranmer, and Peter Battaglia: Hamiltonian graph networks with ode integrators. Neurips workshop Machine Learning and the Physical Sciences 2019, arXiv:1909.12790.
- Peter Toth, Danilo J. Rezende, Andrew Jaegle, Sebastien Racaniere, Aleksandar Botev, and Irina Higgins: Hamiltonian generative networks. ICLR 2020.

LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN
MAX-PLANCK-INSTITUT FÜR PHYSIK