Deep Discrete-Time Lagrangian Mechanics

Takehiro Aoshima[1], Takashi Matsubara[1], Takaharu Yaguchi[2]

[1]Graduate School of Engineering Science, Osaka University

[2]Graduate School of System Informatics, Kobe University

ICLR 2021 Workshop
Deep Learning for Simulation (simDL)

Introduction

Goal: a deep neural network that

- learns the physical dynamics only from the position.
- ensures the energy conservation laws strictly in discrete time.

Background

- Recent studies demonstrated that neural networks learn the physical dynamics associated with the conservation law of energy in continuous time [1].
- The energy is no longer conserved after numerical integrators discretize the time for computer simulations.
- Symplectic integrators conserve a modified energy and a discrete gradient method conserves energy strictly in discrete time [2; 3].
- They need velocity or momentum but measuring an accurate velocity is troublesome.
- Without an accurate velocity, a learned dynamics may be greatly different from the teacher system.
- The Verlet method is a symplectic integrator and depends only on the position.
- No method that conserves energy strictly in discrete time is available when only the position is available.

Proposed approach

- models discrete-time Lagrangian mechanics, which is expressed only with the position.
- ensures the energy conservation laws strictly in discrete time by the *automatic discrete differential (AAD) algorithm* [3].

Method: Theory

Target system

- The potential energy V is expressed as a function of the position $\boldsymbol{q}$.
- The kinetic energy T is expressed as a function of the velocity $\dot{\boldsymbol{q}}$ or momentum $\boldsymbol{p}$.

Hamiltonian mechanics

A system

- has a state $\boldsymbol{u} = (\boldsymbol{q}, \boldsymbol{p})$.
- has a Hamiltonian $\mathcal{H} = T + V$.

Hamilton's equation

$\frac{\mathrm{d}\mathbf{q}}{\mathrm{d}t} = \nabla_{\boldsymbol{p}} T, \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = \nabla_{\boldsymbol{q}} V$ ensures the conservation law of energy.

Lagrangian mechanics

A system

- has a state $\boldsymbol{u} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$.
- has a Lagrangian $\mathcal{L} = T - V$.

Euler-Lagrangian equation

$\frac{\mathrm{d}}{\mathrm{d}t} \nabla_{\dot{q}} T = \nabla_{\boldsymbol{q}} V$ ensures the conservation law of energy.

For conserving energy strictly in discrete time, a discrete gradient has been employed [4].

Hamiltonian mechanics in continuous time

$$\frac{\mathrm{d}\mathbf{q}}{\mathrm{d}t} = \nabla_{\boldsymbol{p}} T$$
$$\frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = \nabla_{\boldsymbol{q}} V$$

Hamiltonian mechanics in discrete time with a discrete gradient

$$\frac{\boldsymbol{q}^{(n+1)} - \boldsymbol{q}^{(n)}}{\Delta t} = \overline{\nabla}_{\boldsymbol{p}} T(\boldsymbol{p}^{(n+1)}, \boldsymbol{p}^{(n)})$$
$$\frac{\boldsymbol{p}^{(n+1)} - \boldsymbol{p}^{(n)}}{\Delta t} = \overline{\nabla}_{\boldsymbol{q}} V(\boldsymbol{q}^{(n+1)}, \boldsymbol{q}^{(n)})$$

$\overline{\nabla} T$ is a discrete gradient of T.

Method: Proposed Approach

Method

- Eliminating the momentum from Hamiltonian mechanics in discrete time.

Hamiltonian mechanics in discrete time with a discrete gradient

$$\frac{\boldsymbol{q}^{(n+1)} - \boldsymbol{q}^{(n)}}{\Delta t} = \overline{\nabla}_{\boldsymbol{p}} T(\boldsymbol{p}^{(n+1)}, \boldsymbol{p}^{(n)}), \frac{\boldsymbol{p}^{(n+1)} - \boldsymbol{p}^{(n)}}{\Delta t} = \overline{\nabla}_{\boldsymbol{q}} V(\boldsymbol{q}^{(n+1)}, \boldsymbol{q}^{(n)})$$

Proposed: discrete-time Lagrangian mechanics

$$\boldsymbol{M} \frac{\boldsymbol{q}^{(n+1)} - 2\boldsymbol{q}^{(n)} + \boldsymbol{q}^{(n-1)}}{(\Delta t)^2} = -\frac{1}{2}\left(\overline{\nabla}_{\boldsymbol{q}} V(\boldsymbol{q}^{(n+1)}, \boldsymbol{q}^{(n)}) + \overline{\nabla}_{\boldsymbol{q}} V(\boldsymbol{q}^{(n)}, \boldsymbol{q}^{(n-1)})\right) \quad (1)$$

- The potential energy V is modeled by a neural network.
- A discrete gradient $\overline{\nabla}_q V$ is obtained by the *ADD algorithm* [3].
- M is a mass matrix.

Since the proposed equation (1) converges to the Euler-Lagrangian equation, it is considered as a discrete-time Lagrangian mechanics.

Proposed approach

- learns the physical dynamics only from the position by modeling a discrete-time Lagrangian mechanics.
- ensures the conservation law of energy strictly in discrete time using a discrete gradient.

Training

- Minimizing mean squared error between left- and right-hand sides of Eq. (1).

Prediction

- A next state is estimated by implicitly solving Eq. (1).

Experiments and Results

Datasets

- mass-spring system, pendulum system, and 2-body system

Comparative methods

- Euler method, symplectic Euler method, and leapfrog integrator
- "leapfrog + proposed" use leapfrog integrator for training and the proposed approach for prediction.

Results

- The proposed approach and the leapfrog integrator predict the state at a similar level.
- For the mass-spring and pendulum systems, the proposed approach conserves energy accurately, symplectic Euler method and leapfrog integrator conserve a modified energy only from the position but they do not conserve energy strictly.
- For the 2-body system, "leapfrog + proposed" conserves energy accurately.
- The proposed approach conserves modeled energy.

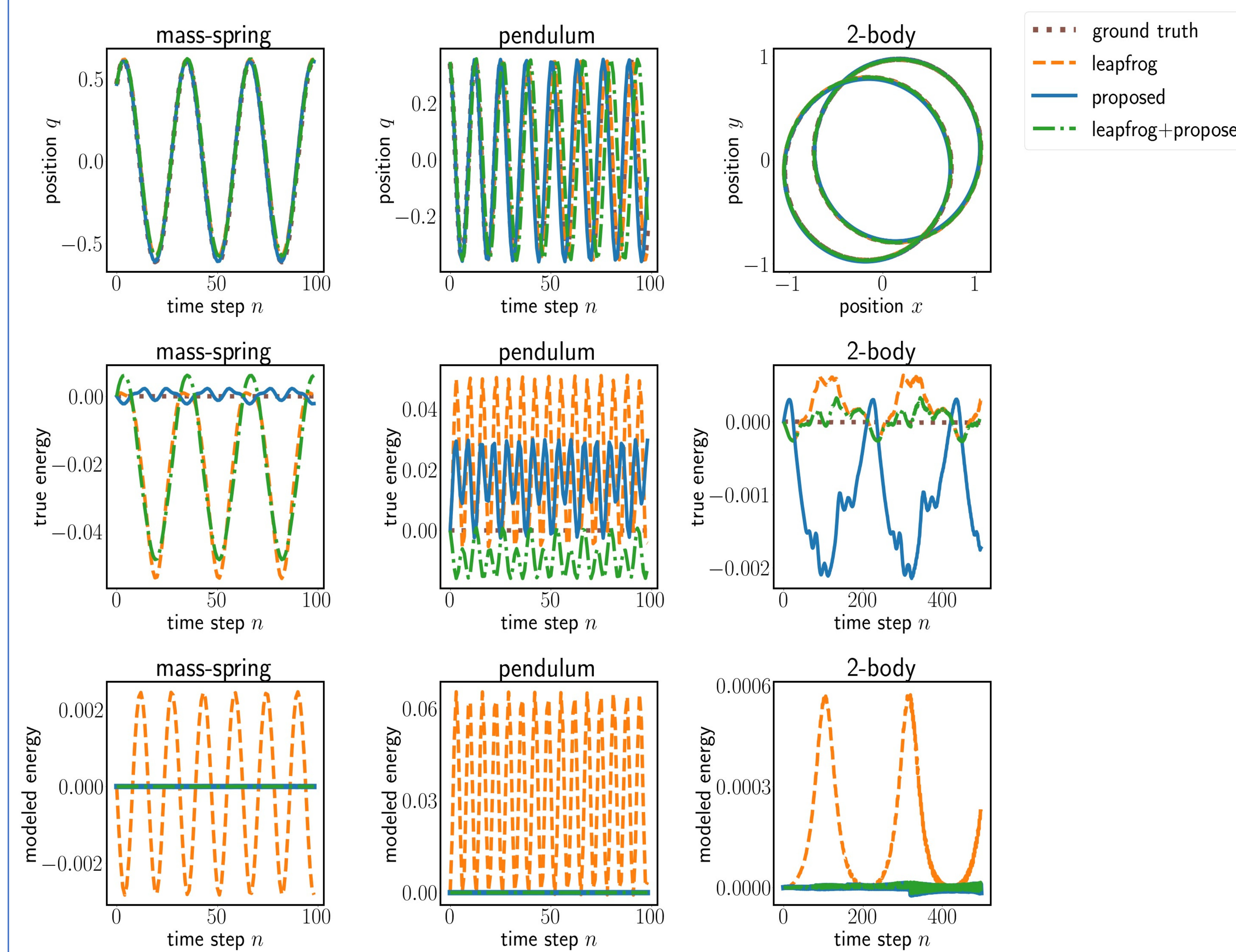

Figure 1: Results. (top) Position $\boldsymbol{q}$. (center) True energy. (bottom) Energy modeled by each model (modeled energy).

Table 1: Mean squared errors averaged over 15 trials across all three tasks.

Model	Mass-Spring		Pendulum		2-Body	
	Position q	**Energy**	**Position q**	**Energy**	**Position q**	**Energy**
Euler	4.48×10^{0}	1.23×10^{1}	2.96×10^{2}	8.54×10^{1}	3.71×10^{-1}	8.64×10^{-4}
symplectic Euler	7.09×10^{-2}	1.67×10^{-3}	$\mathbf{4.89\times10^{-3}}$	7.45×10^{-2}	$\mathbf{1.65\times10^{-4}}$	2.77×10^{-7}
leapfrog	7.09×10^{-2}	6.57×10^{-4}	$\mathbf{4.89\times10^{-3}}$	2.04×10^{-2}	$\mathbf{1.65\times10^{-4}}$	$\mathbf{1.14\times10^{-7}}$
proposed	$\mathbf{6.20\times10^{-2}}$	$\mathbf{4.52\times10^{-5}}$	5.43×10^{-3}	$\mathbf{2.20\times10^{-3}}$	7.31×10^{-3}	3.00×10^{-6}
leapfrog+proposed	8.30×10^{-2}	6.19×10^{-4}	1.47×10^{-1}	$\mathbf{8.38\times10^{-4}}$	$\mathbf{1.59\times10^{-4}}$	$\mathbf{1.01\times10^{-7}}$

Table 2: The variance of modeled energy.

Model	Mass-Spring	Pendulum	2-Body
Euler	3.84×10^{-1}	3.85×10^{1}	1.35×10^{-4}
symplectic Euler	1.06×10^{-3}	7.15×10^{-2}	1.70×10^{-7}
leapfrog	3.10×10^{-5}	1.15×10^{-2}	2.78×10^{-9}
proposed	$\mathbf{2.11\times10^{-12}}$	$\mathbf{4.58\times10^{-11}}$	$\mathbf{2.42\times10^{-11}}$

References

[1] Greydanus *et al.* (2019) "Hamiltonian Neural Networks." In: Advances in Neural Information Processing Systems.

[2] Chen *et al.* (2020) "Symplectic Recurrent Neural Networks." In: International Conference on Learning Representations.

[3] Matsubara *et al.* (2020). "Deep Energy-Based Modeling of Discrete-Time Physics." In: Advances in Neural Information Processing Systems.

[4] Hairer *et al.* (2006). Geometric Numerical Integration: Structure-Preserving Algorithm for Ordinary Differential Equations. Springer.