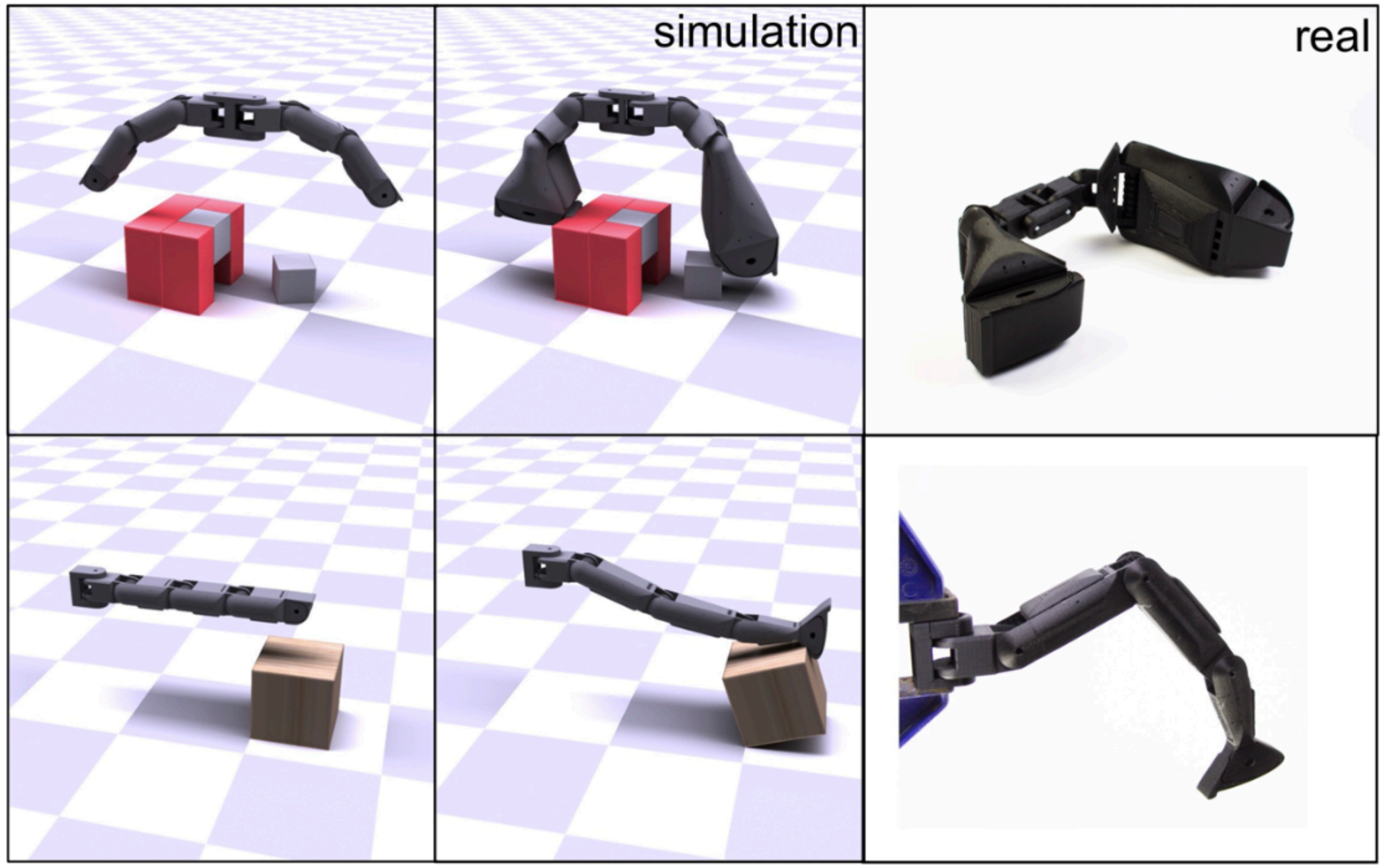
An End-to-End Differentiable Framework for Contact-Aware Robot Design

Jie Xu[†], Tao Chen[†], Lara Zlokapa[†], Michael Foshey[†], Wojciech Matusik[†], Shinjiro Sueda[‡] and Pulkit Agrawal[†]
[†]Massachusetts Institute of Technology
[‡]Texas A&M University
http://diffhand.csail.mit.edu

***Abstract*—The current dominant paradigm for robotic manipulation involves two separate stages: manipulator design and control. Because the robot's morphology and how it can be controlled are intimately linked, joint optimization of design and control can significantly improve performance. Existing methods for co-optimization are limited and fail to explore a rich space of designs. The primary reason is the trade-off between the complexity of designs that is necessary for contact-rich tasks against the practical constraints of manufacturing, optimization, contact handling, etc. We overcome several of these challenges by building an end-to-end differentiable framework for contact-aware robot design. The two key components of this framework are: a novel deformation-based parameterization that allows for the design of articulated rigid robots with arbitrary, complex geometry, and a differentiable rigid body simulator that can handle contact-rich scenarios and computes analytical gradients**



JIE XU

CONTACT

Email: jiex {at} csail {dot} mit {dot} edu
Office Address: The Stata Center, 32-311,
32 Vassar Street, Cambridge, MA 02139

Last Update: Feb 2022

JIE XU

I am a final-year Ph.D. student at MIT CSAIL, advised by Professor Wojciech Matusik in the Computational Design and Fabrication Group (CDFG). I obtained my bachelor's degree from Department of Computer Science and Technology at Tsinghua University with honor in 2016. During my undergraduate period, I worked with Professor Shi-Min Hu in the Tsinghua Graphics & Geometric Computing Group and with Professor Stelian Coros as an exchange student at CMU.

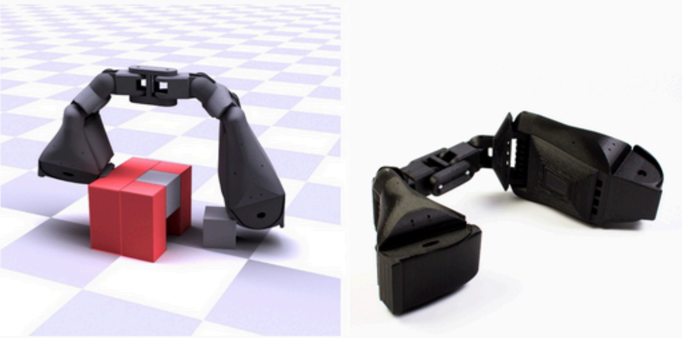
My research mainly focuses on the intersection of Robotics, Simulation, and Machine Learning. Specifically, I am interested in the following topics: *Robotics Control*, *Reinforcement Learning*, *Differentiable Physics-based Simulation*, *Robotics Control and Design Co-Optimization*, and *Sim-to-Real*.

PUBLICATIONS

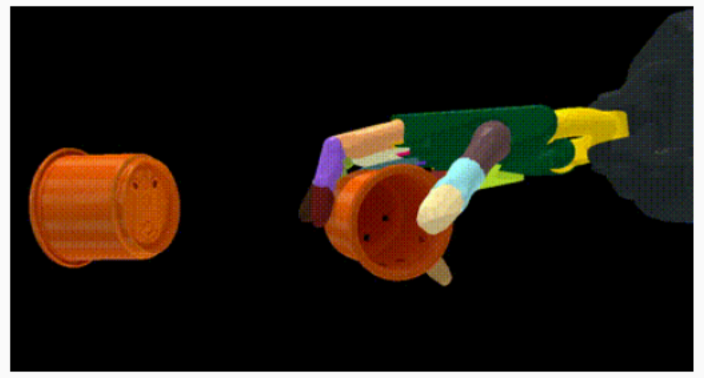
Accelerated Policy Learning with Parallel Differentiable Simulation
Jie Xu, Viktor Makoviychuk, Yashraj Narang, Fabio Ramos, Wojciech Matusik, Animesh Garg, Miles Macklin
International Conference on Learning Representations (ICLR 2022)
[Project Page] | [Paper]

DiffCloth: Differentiable Cloth Simulation with Dry Frictional Contact
Yifei Li, Tao Du, Kui Wu, ***Jie Xu***, Wojciech Matusik
ACM Transactions on Graphics, 2022 (minor revisions)
[Project Page] | [Arxiv]

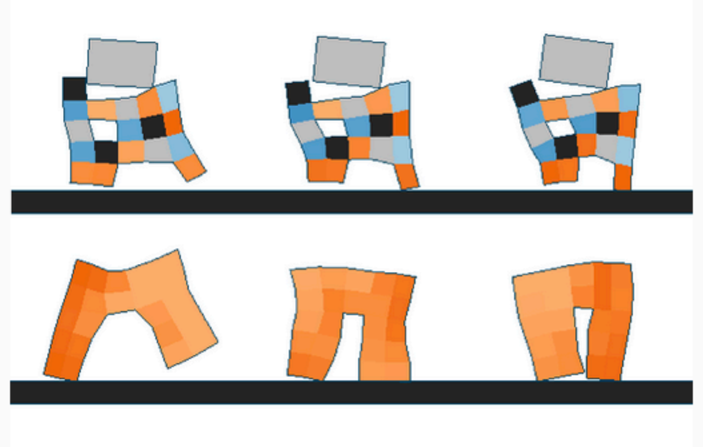
An End-to-End Differentiable Framework for Contact-Aware Robot Design
Jie Xu, Tao Chen, Lara Zlokapa, Michael Foshey, Wojciech Matusik, Shinjiro Sueda, Pulkit Agrawal
Robotics: Science and Systems (RSS 2021)
[Project Page] | [Paper] | [Supp] | [Sim Doc] | [Code] | [Video] | [Talk] | [MIT News]



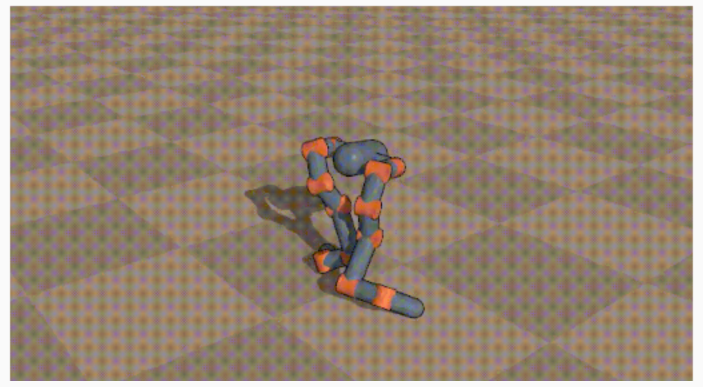
A System for General In-Hand Object Re-Orientation
Tao Chen, ***Jie Xu***, Pulkit Agrawal
Conference on Robot Learning (CoRL 2021) **(Best Paper Award)** *(Oral: 6.5%)*
[Project Page] | [Paper] | [Supp] | [Arxiv] | [Video] | [BibTeX] | [MIT News]



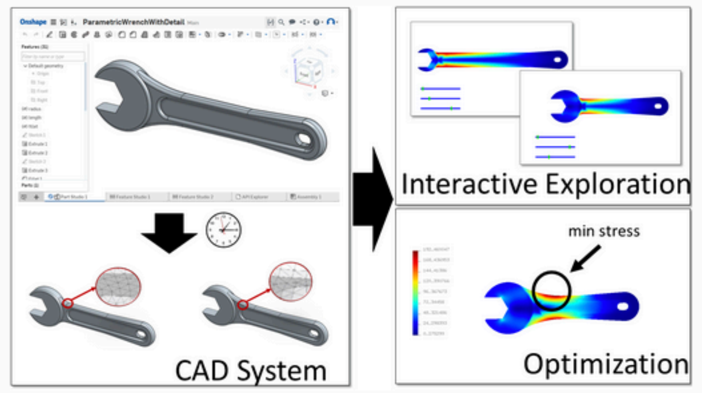
Evolution Gym: A Large-Scale Benchmark for Evolving Soft Robots
Jagdeep Singh Bhatia, Holly Jackson, Yunsheng Tian, ***Jie Xu***, Wojciech Matusik
Conference on Neural Information Processing Systems (NeurIPS 2021)
[Project Page] | [Paper] | [Supp] | [Code] | [Video] | [BibTeX] | [MIT News]



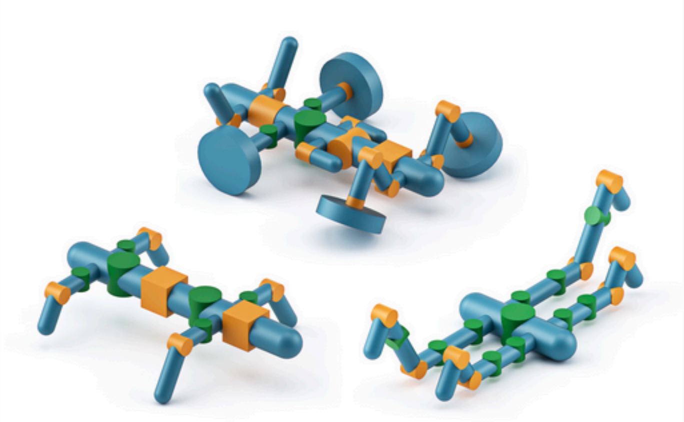
Multi-Objective Graph Heuristic Search for Terrestrial Robot Design
Jie Xu, Andrew Spielberg, Allan Zhao, Daniela Rus, Wojciech Matusik
IEEE International Conference on Robotics and Automation (ICRA 2021)
[Project Page] | [Paper] | [Arxiv] | [Video] | [Talk]



Interactive Design Space Exploration and Optimization for CAD Models
Adriana Schulz, ***Jie Xu***, Bo Zhu, Changxi Zheng, Eitan Grinspun, Wojciech Matusik
ACM Transactions on Graphics 36(4) (Proc. SIGGRAPH 2017)
[Project Page] | [Paper] | [Supplementary] | [Code] | [Video]



RoboGrammar: Graph Grammar for Terrain-Optimized Robot Design
Allan Zhao, ***Jie Xu***, Mina Konaković Luković, Josephine Hughes, Andrew Spielberg, Daniela Rus, Wojciech Matusik
ACM Transactions on Graphics 39(6) (Proc. SIGGRAPH Asia 2020)
[Project Page] | [Paper] | [Video] | [Code] | [MIT News]



CDFG The Computational Design & Fabrication Group

Massachusetts Institute of Technology

CSAIL

Homepage | Team | Research | Publications | Courses | News | Contact

Wojciech Matusik

Professor

Department of Electrical Engineering and Computer Science

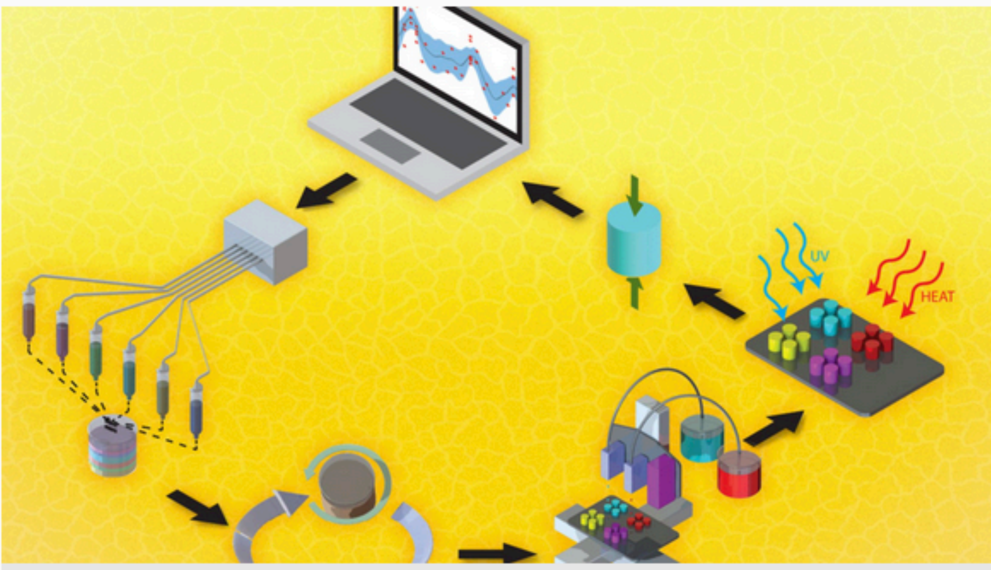
Biography

Wojciech Matusik is a Professor of Electrical Engineering and Computer Science at the Computer Science and Artificial Intelligence Laboratory at MIT, where he leads the Computational Design and Fabrication Group and is a
member of the
to MIT, he work
Laboratories, A
He studied com
in 2003. He also
of California at
in 2001. His rese
computational c
robotics, and hc
top 100 young i
Magazine. In 20
Researcher Awa

https://cdfg.mit.edu/

Selected Publications

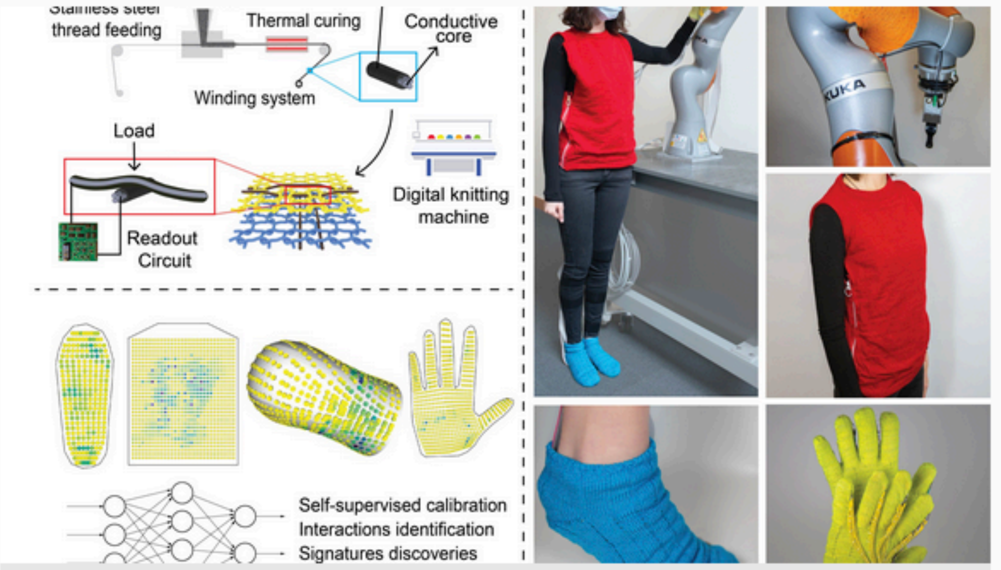
View All Publications →



SCIENCE ADVANCES

Accelerated discovery of 3D printing materials using data-driven multiobjective optimization

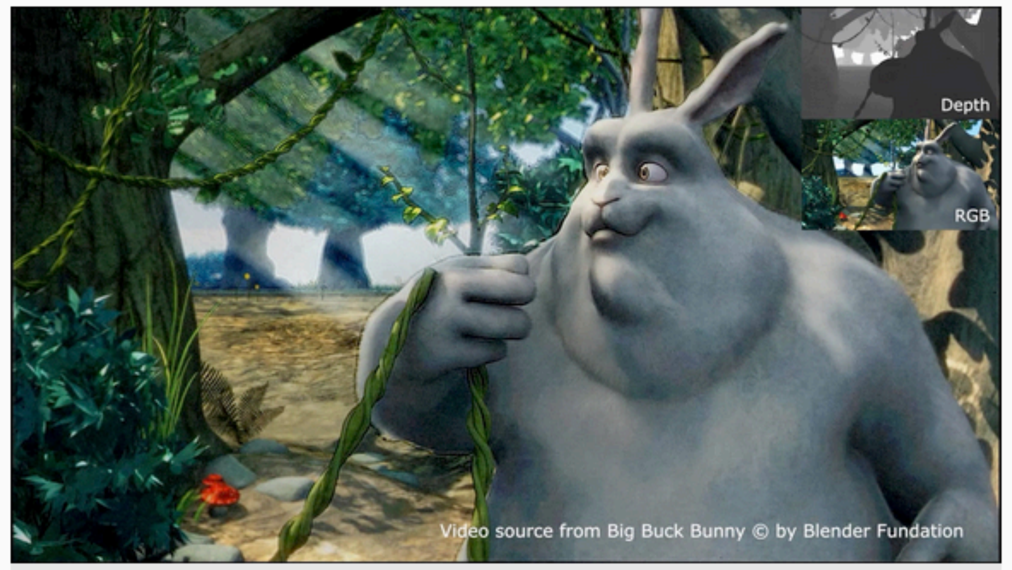
T. Erps, M. Foshey, M. K. Luković, W. Shou, H. H. Goetzke, H. Dietsch, K. Stoll, B. von Vacano, W. Matusik



Nature Electronics

Learning Human-environment Interactions using Conformal Tactile Textiles

Yiyue Luo, Yunzhu Li, Pratyusha Sharma, Wan Shou, Kui Wu, Michael Foshey, Beichen Li, Tomás Palacios, Antonio Torralba, Wojciech Matusik



Nature

Towards real-time photorealistic 3D holography with deep neural networks

Liang Shi, Beichen Li, Changil Kim, Petr Kellnhofer, Wojciech Matusik

Pulkit Agrawal

I am an Assistant Professor in the department of Electrical Engineering and Computer Science (EECS) at MIT. My lab is a part of the Computer Science and Artificial Intelligence Lab (CSAIL), is affiliated with the Laboratory for Information and Decision Systems (LIDS) and involved with NSF AI Institute for Artificial Intelligence and Fundamental Interactions (IAIFI).

I completed my Ph.D. at UC Berkeley; undergraduate studies from IIT Kanpur. Co-founded SafelyYou Inc. that builds fall prevention technology.

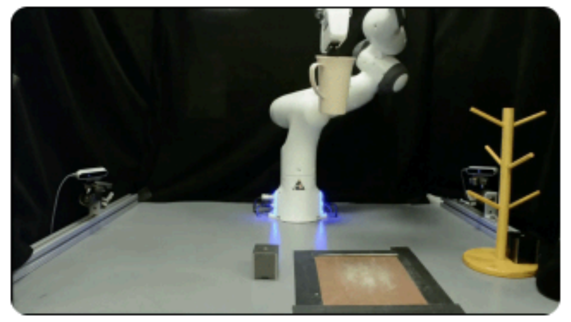
Email / CV / Biography / Google Scholar / LinkedIn

Research

The overarching research interest is to build machines that can automatically and continuou
environment. The hope is that the end result of such learning will be similar to development
sense. I refer to this line of work as **"computational sensorimotor learning"** and it encompas
reinforcement learning, and other learning based approaches to control. Some of my past w
principles of **cognitive science, neuroscience** to draw upon inspiration from these discplines

Ph.D. Thesis (Computational Sensorimotor Learning) / Thesis

Pre-Prints

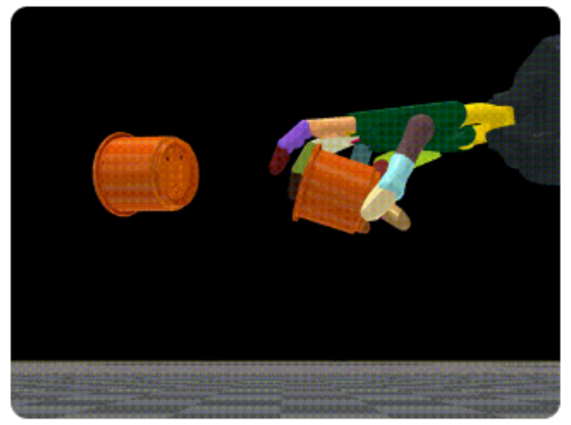


Neural Descriptor Fields: SE(3)-Equivariant Object Representations for Manipulation
Anthony Simeonov*, Yilun Du*, Andrea Tagliasacchi, Joshua B. Tenenbaum, Alberto Rodriguez, **Pulkit Agrawal****, Vincent Sitzmann**
(*equal contribution, order determined by coin flip. **equal advising) *arXiv*, 2021

paper / website and code / bibtex

An SE(3) Equivariant method for specifying and finding correspondences which enables data efficient object manipulation.

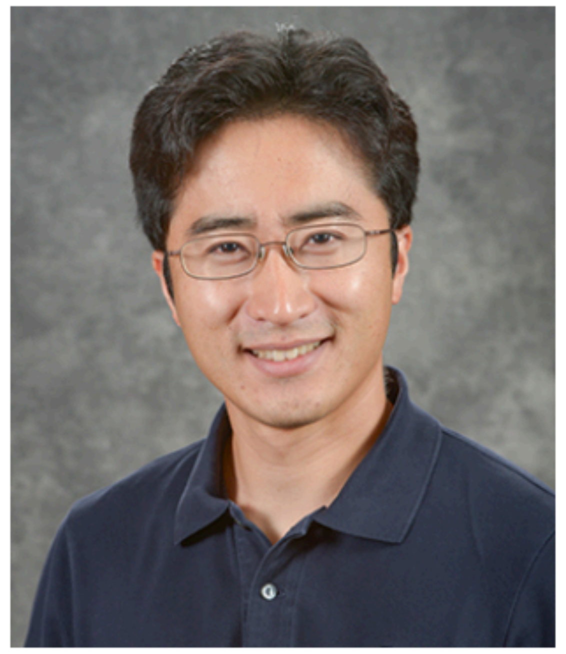
Publications



A System for General In-Hand Object Re-Orientation
Tao Chen, Jie Xu, **Pulkit Agrawal**
CoRL, 2021 **(Best Paper Award)**

paper / bibtex / project page
Press: MIT News

A framework for general in-hand object reorientation.



Assistant Professor
Computer Science
Texas A&M University
Aggie Graphics Group

Peterson 405
sueda@tamu.edu

Prior to this appointment at Texas A&M University, I enjoyed two yea
doctoral fellow at Disney Research Boston and the MIT Computationa
main research area is computer graphics, specializing in physically base
CAREER award.

GitHub Repositories:

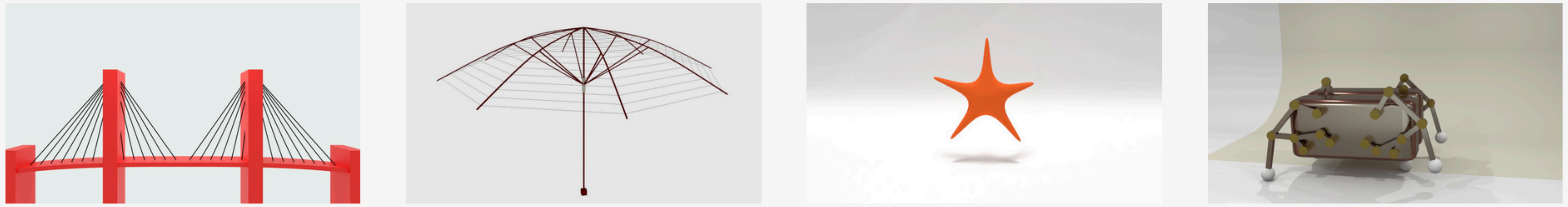
- REDMAX: Analytically Differentiable Articulated Rigid Body D
- EoL Cloth: Eulerian-on-Lagrangian Cloth Simulation
- FBX extract: For extracting data from FBX files
- Brender: For rendering simulations with Blender

REDMAX: Efficient & Flexible Approach for Articulated Dynamics

Ying Wang[1] Nicholas J. Weidner[1] Margaret A. Baxter[1] Yura Hwang[1]
Danny M. Kaufman[2] Shinjiro Sueda[1]

[1]Texas A&M University [2]Adobe Research

ACM Transactions on Graphics (SIGGRAPH 2019)



Abstract

It is well known that the dynamics of articulated rigid bodies can be solved in O(n) time using a recursive method, where n is the number of joints. However, when elasticity is added between the bodies (eg damped springs), with linearly implicit integration, the stiffness matrix in the equations of motion breaks the tree topology of the system, making the recursive O(n) method inapplicable. In such cases, the only alternative has been to form and solve the system matrix, which takes $O(n^3)$ time. We propose a new approach that is capable of solving the linearly implicit equations of motion in near linear time. Our method, which we call RedMax, is built using a combined reduced/maximal coordinate formulation. This hybrid model enables direct flexibility to apply arbitrary combinations of constraints and contact modeling in both reduced and maximal coordinates, as well as mixtures of implicit and explicit forces in either coordinate representation. We highlight Red/Max's flexibility with seamless integration of deformable objects with two-way coupling, at a standard additional cost. We further highlight its flexibility by constructing an efficient internal (joint) and external (environment) frictional contact solver that can leverage bilateral joint constraints for rapid evaluation of frictional articulated dynamics.

NEW: We now have an *analytically differentiable* version of REDMAX! It now supports fully implicit time integration and the adjoint method for parameter optimization. Please visit the github page!

Co-optimize design and control

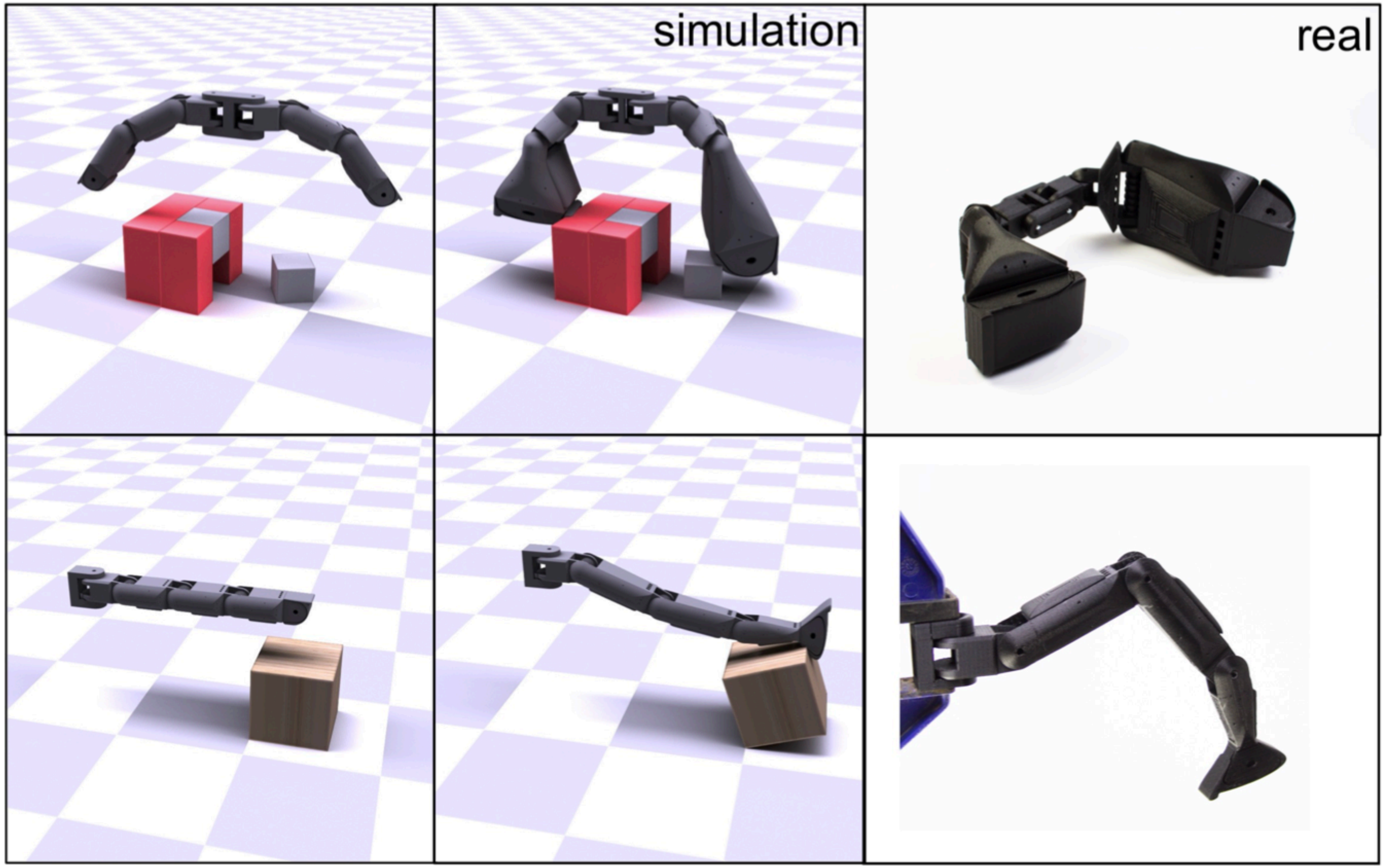


Fig. 1. *Left column*: only optimizing the control algorithm using a nominal robot design fails to complete the task; *Middle*: co-optimization of morphology and control results in success; *Right:* pictures of 3D-printed manipulators. Our method outputs designs that are easy to print and assemble.

Make it possible to use analytical derivatives everywhere

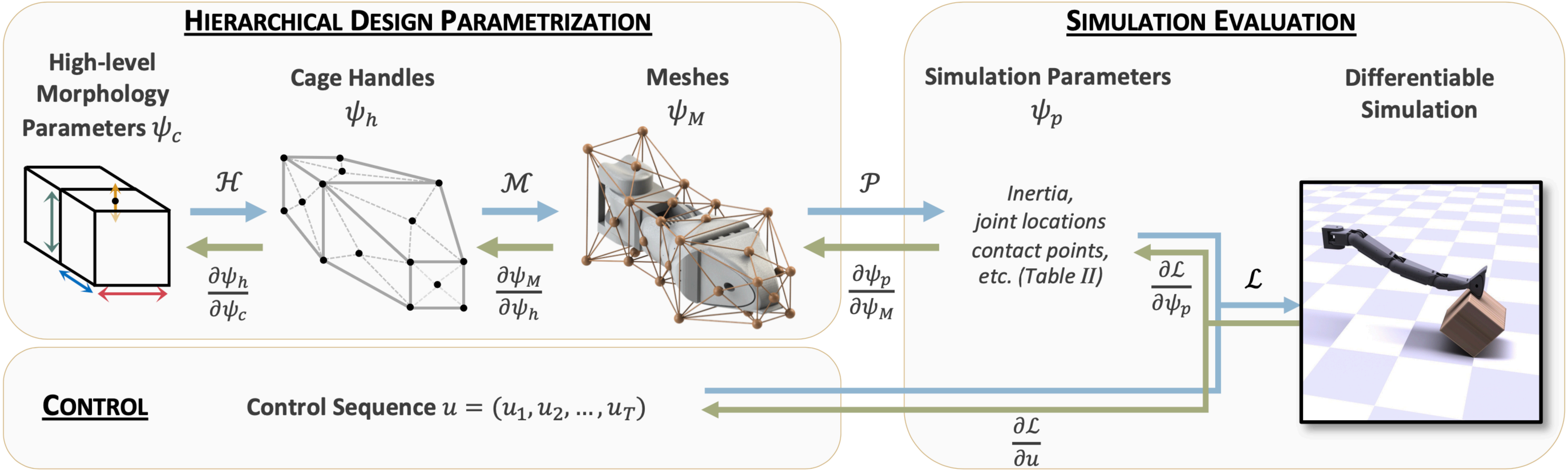


Fig. 2. **End-to-end differentiable framework for morphology and control co-optimization.** Blue arrows labeled as $\mathcal{H}, \mathcal{M}, \mathcal{P}$, and $\mathcal{L}$ are hierarchical functions that evaluate the loss function given the high-level morphology parameters, ψ_c and controls, u. The corresponding green arrows are the derivatives.

Grammar for a manipulator

$S \Rightarrow B$

$B \Rightarrow B \leftarrow F$

$B \leftarrow F \Rightarrow F \rightarrow B \leftarrow F$

$F \Rightarrow \varepsilon$

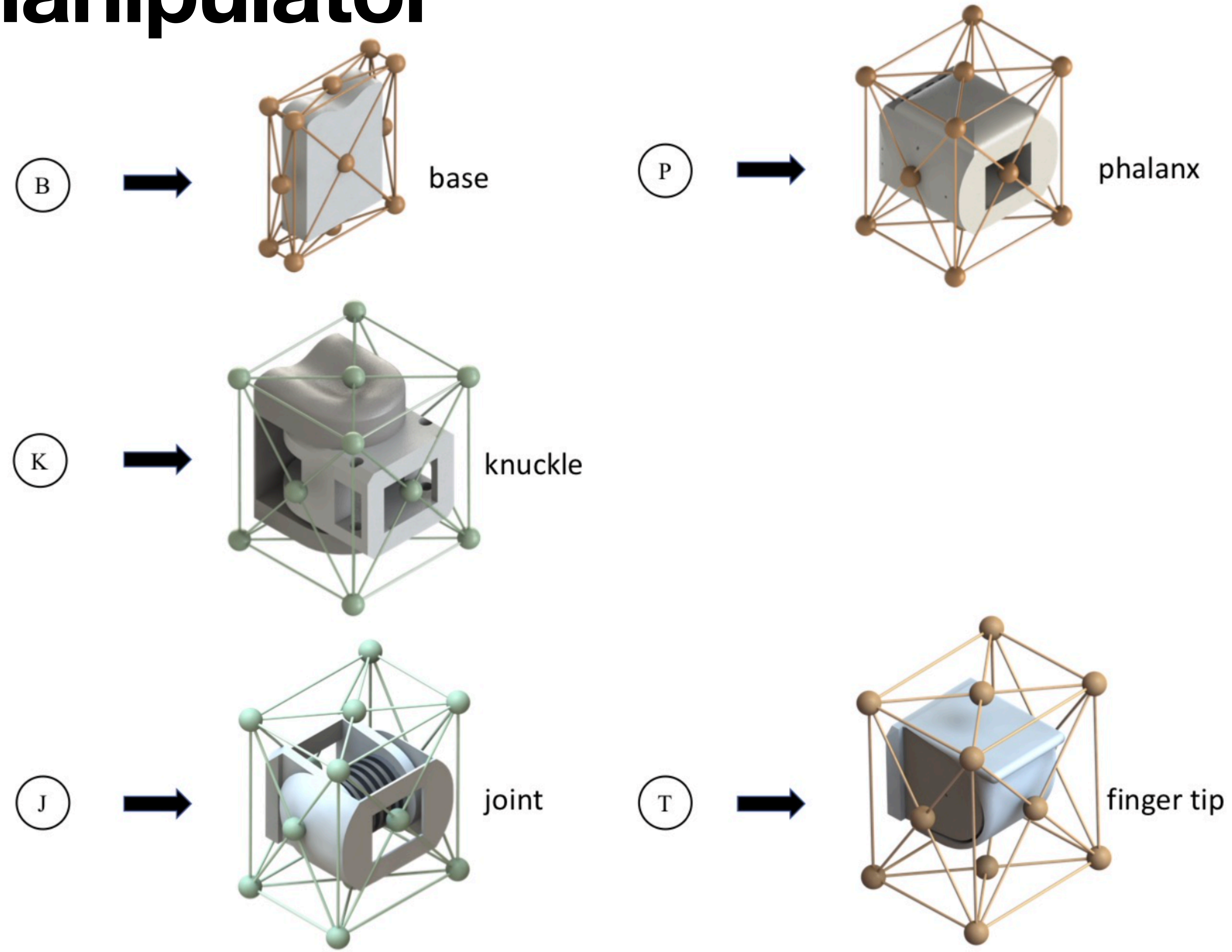
$F \Rightarrow K \leftarrow J \leftarrow M$

$F \Rightarrow J \leftarrow M$

$M \Rightarrow P \leftarrow J \leftarrow M$

$M \Rightarrow P \leftarrow T$

$M \Rightarrow T$



Component database

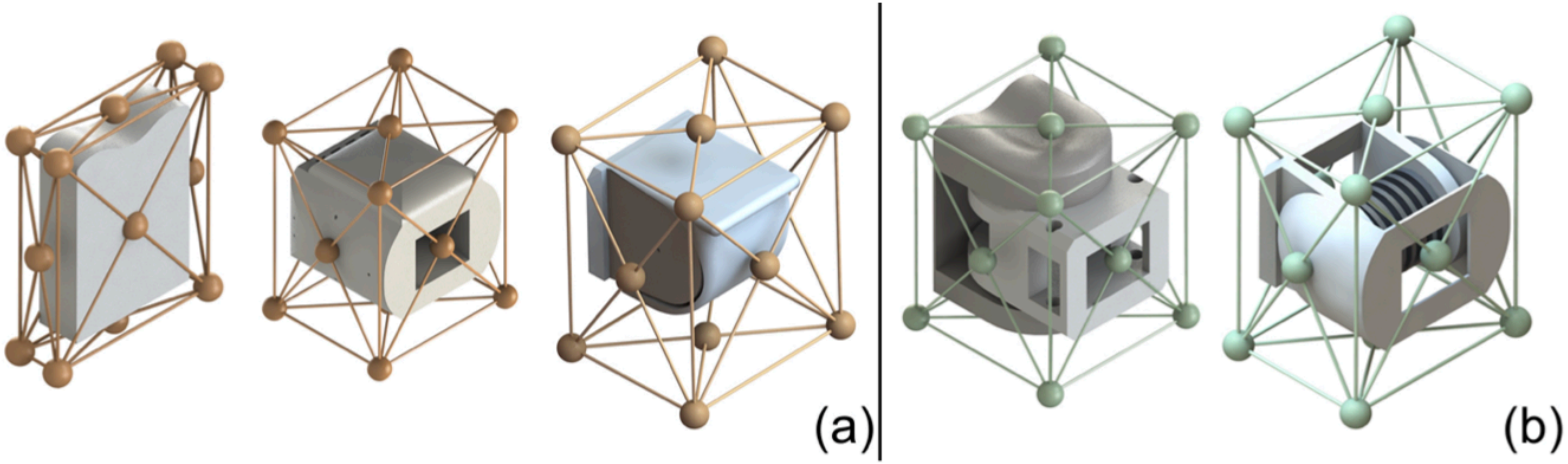


Fig. 3. Our component database for our manipulators. From left to right: finger base, phalanx segment, finger tip, knuckle, and joint. Each component comes with its own deformation cage. (a) The components in the yellow cages can be deformed arbitrarily with the cage, whereas (b) the components in the green cages can only be expanded along the axis of rotation.

Connectivity

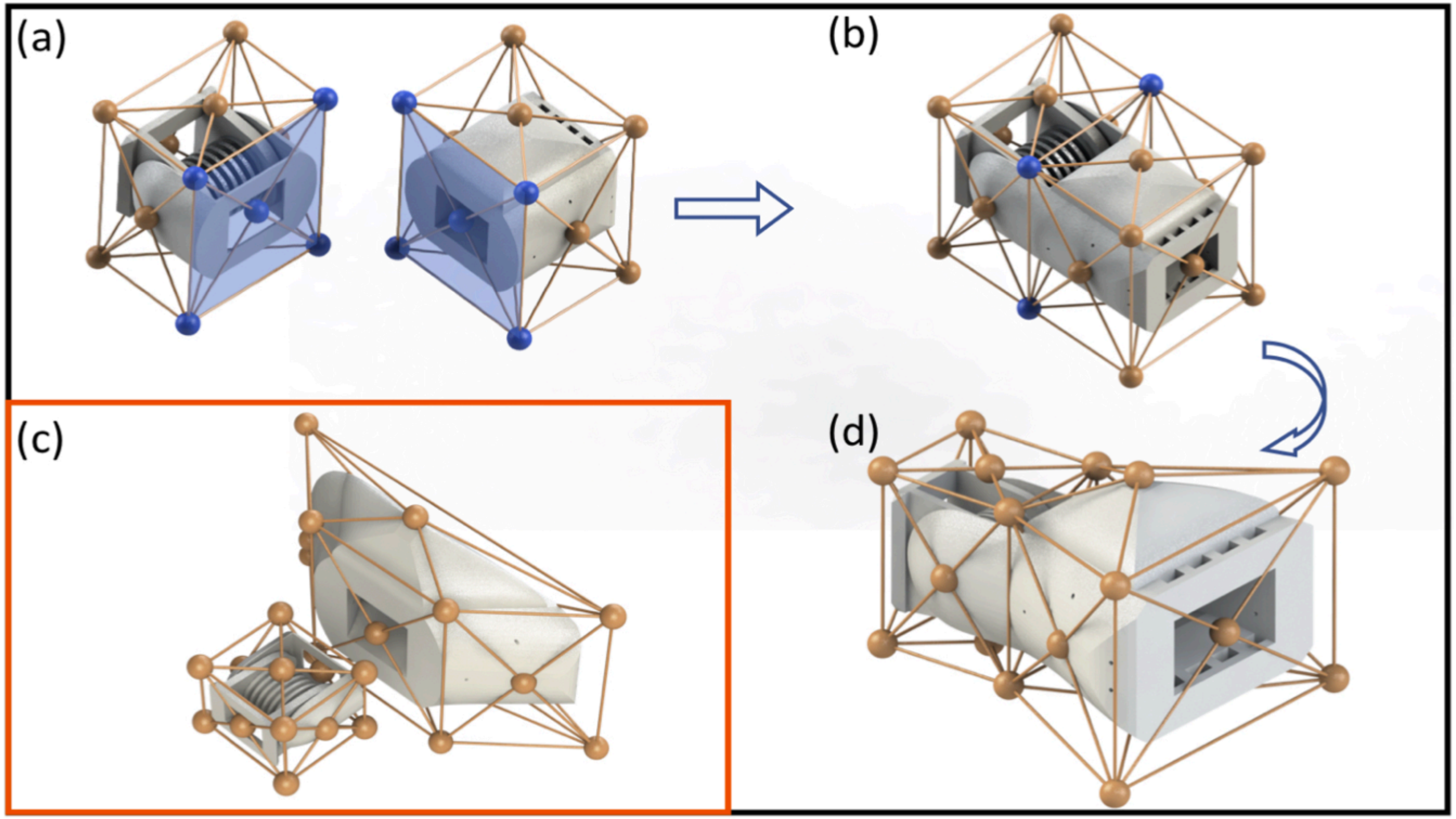


Fig. 4. A *joint* component and a *body phalanx segment* component are shown in the figures. We parameterize the articulated components into lower-dimensional parameters ψ_c by posing different deformation constraints on each component and merging their handle points on the connection surface (highlighted in blue in (a) and (b)). We can then freely explore the ψ_c space to change the underlining articulated robot shape (d). The two components come apart from each other and become to be not manufacturable if they are deformed individually and arbitrarily by their associated cages (c).

Diversity of shape space

Fig. 5. **Morphology design space**. The initial morphology of the single finger and the two-finger gripper designs are shown on the left. We randomly sample different parameters for each configuration and show the deformed morphology on the right.

Analytical differentiation: Most of these steps are straightforward to differentiate…

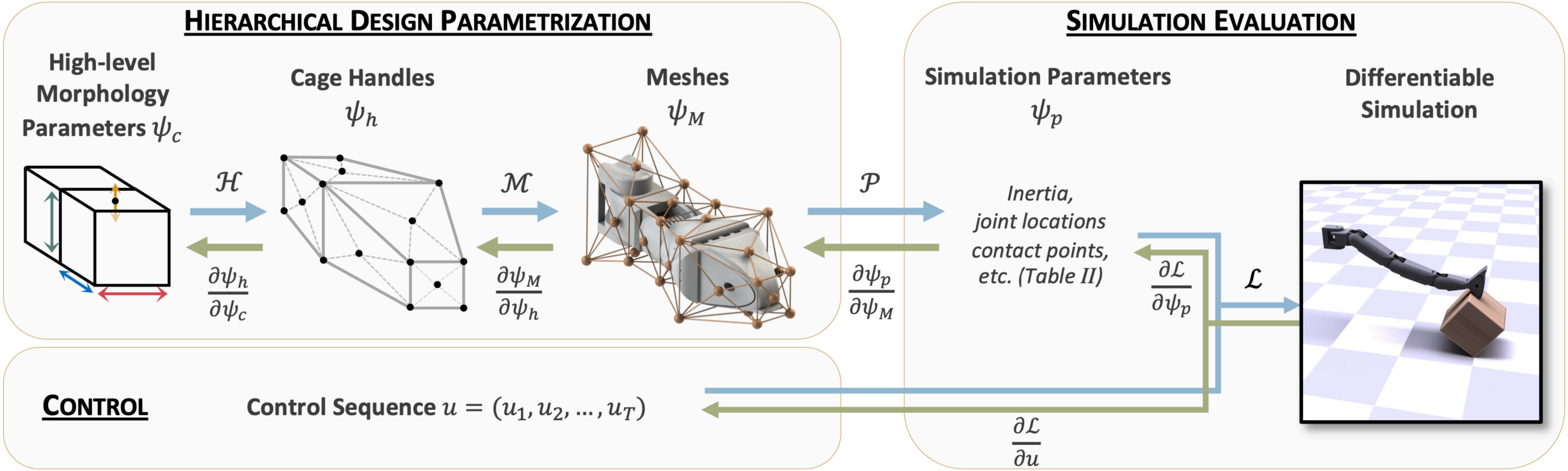


Fig. 2. **End-to-end differentiable framework for morphology and control co-optimization.** Blue arrows labeled as $\mathcal{H}, \mathcal{M}, \mathcal{P}$, and $\mathcal{L}$ are hierarchical functions that evaluate the loss function given the high-level morphology parameters, ψ_c and controls, u. The corresponding green arrows are the derivatives.

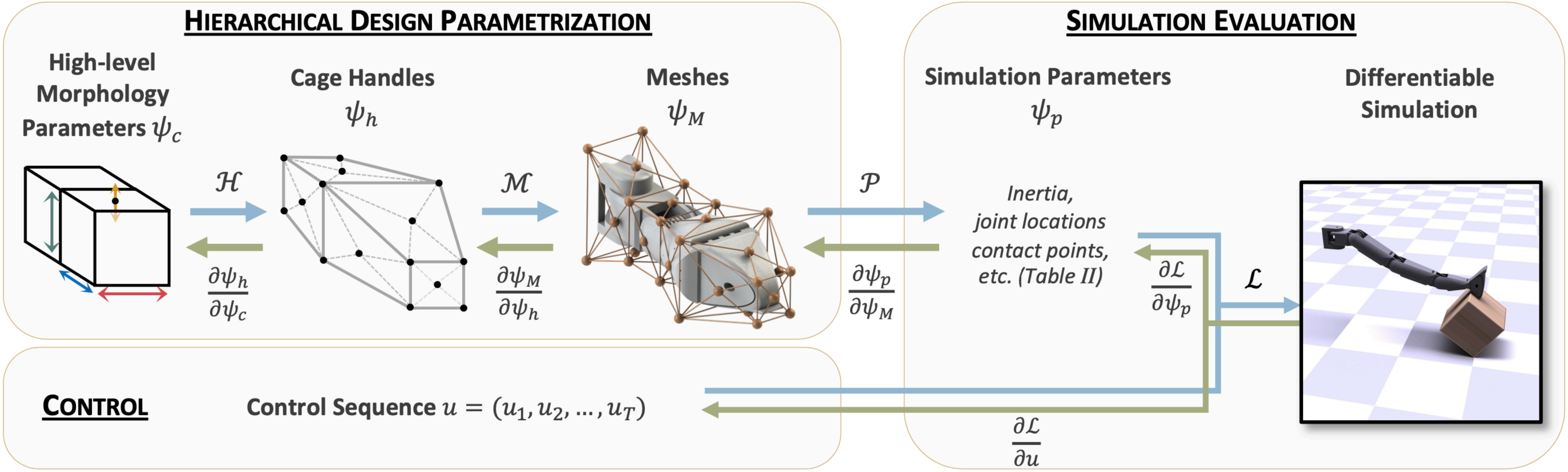


Fig. 2. **End-to-end differentiable framework for morphology and control co-optimization.** Blue arrows labeled as $\mathcal{H}, \mathcal{M}, \mathcal{P}$, and $\mathcal{L}$ are hierarchical functions that evaluate the loss function given the high-level morphology parameters, ψ_c and controls, u. The corresponding green arrows are the derivatives.

$$\frac{\partial \mathcal{L}}{\partial u} \qquad \frac{\partial \mathcal{L}}{\partial \psi_c} = \frac{\partial \mathcal{L}}{\partial \psi_p} \frac{\partial \psi_p}{\partial \psi_c} \qquad \frac{\partial \psi_p}{\partial \psi_c} = \frac{\partial \psi_p}{\partial \psi_M} \frac{\partial \psi_M}{\partial \psi_h} \frac{\partial \psi_h}{\partial \psi_c}.$$

Parameters available to the optimizer

Contacts are made differentiable using signed distance functions

To solve this, we use a signed distance field augmented with derivative information. This distance field is attached to the manipulated object, which we assume is rigid. Let the generalized coordinates of the robot and the manipulated object be q_b and q_o, respectively, and the generalized velocity be $\dot{q}_b$ and $\dot{q}_o$, respectively. Then the world position and velocity of the contact point C_b on the robot body are computed as $\mathbf{x}(q_b, C_b)$ and $\dot{\mathbf{x}}(q_b, \dot{q}_b, C_b)$. Using this world point, we query the signed distance function attached to the manipulated object, which gives us the following quantities:

d, $\dot{d}$	penetration distance and speed
$\mathbf{n}$	contact normal
$\dot{\mathbf{t}}$	tangential velocity

as well as the *derivatives* of these quantities with respect to q_b, q_o, and C_b.

Solve for design and control using L-BFGS-B

Three other baselines were used for comparison

Baselines We adopted the following three baseline algorithms for comparison.

1) *ES*: Evolutionary strategy is widely used to search for optimal design and control parameters for robots [14, 36]. We tried various ES algorithms in the open-sourced Nevergrad library [43] and found that the $(1+1)$-*ES* [5, 44] algorithm and *CMA-ES* [23] work best on the proposed tasks.
2) *RL*: Luck et al. [34] is one of the state-of-the-art morphology and control co-optimization approaches using sample-efficient reinforcement learning (soft actor critic, SAC) algorithm and particle swarm optimization. We used their released implementation as a baseline.
3) *Control Only*: In this algorithm, we freeze the morphology parameters and only optimize the control sequence with L-BFGS-B.

Tasks

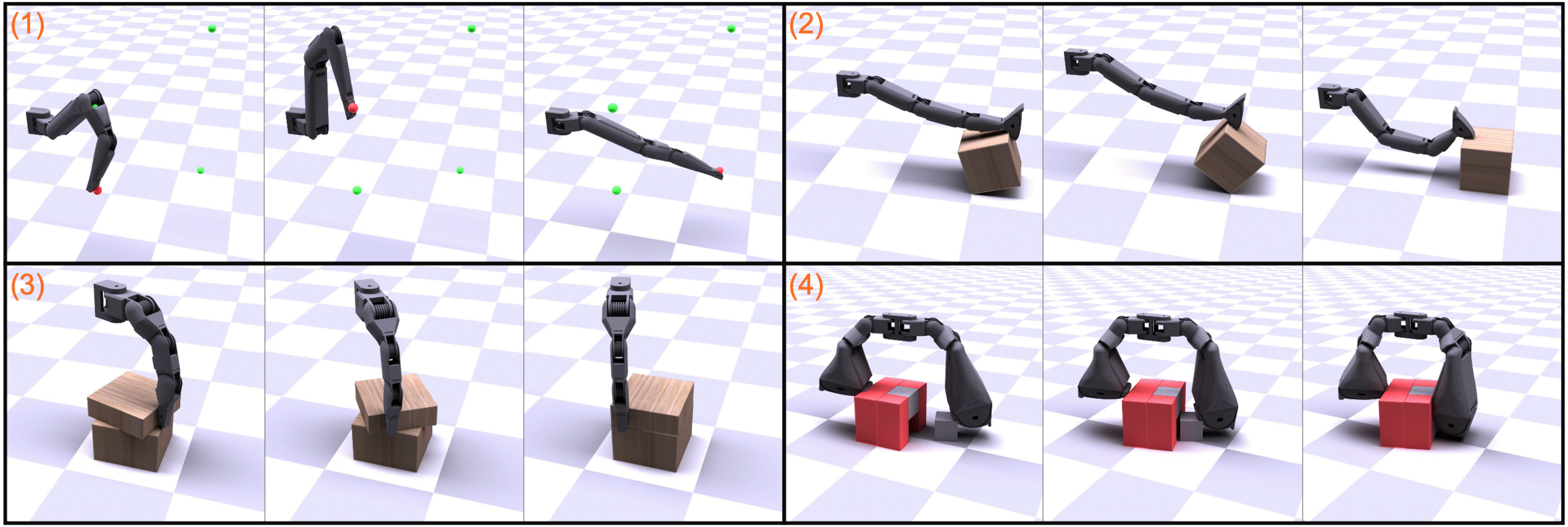
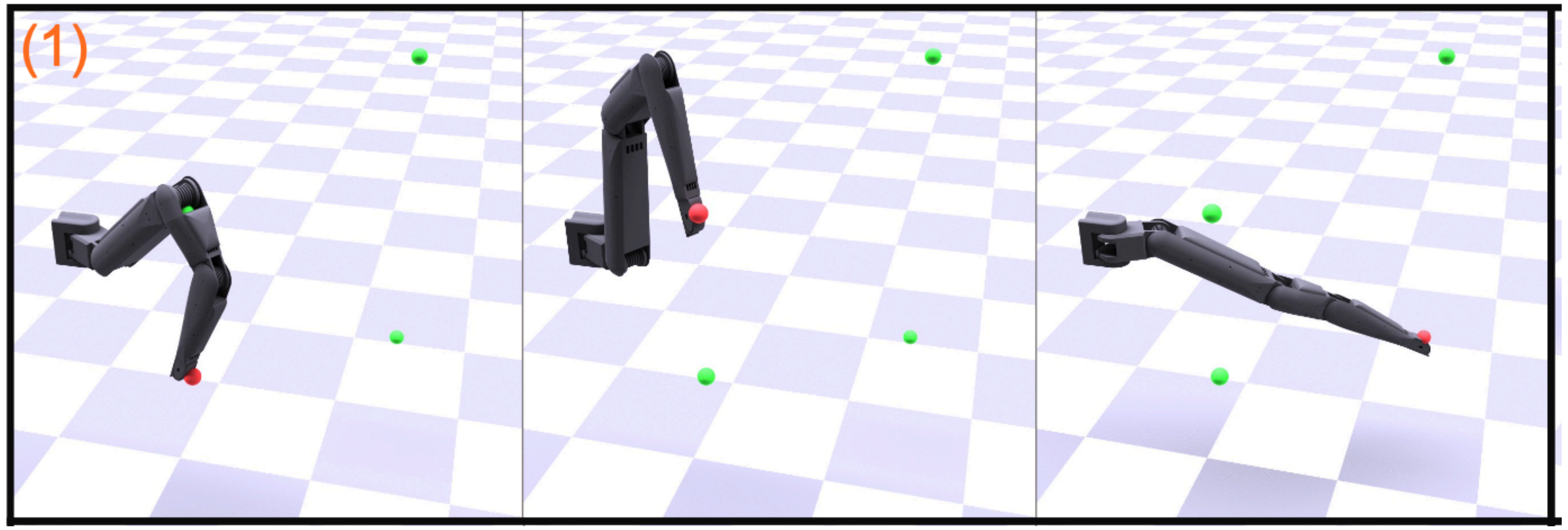


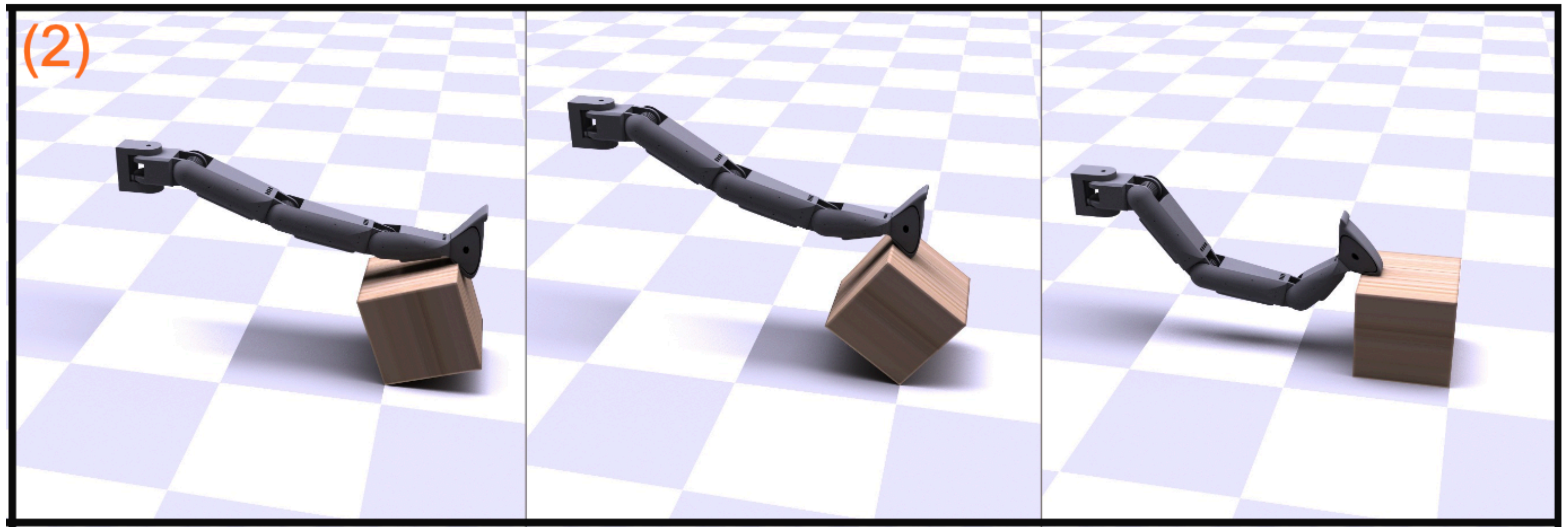
Fig. 7. **Optimized designs and controls for four manipulator tasks**. (1) *Finger Reach*. (2) *Flip Box*. (3) *Rotate Rubik's Cube*. (4) *Assemble*. More visual results are provided in the supplementary video.



1) *Finger Reach*: In this task, the finger's base is mounted on the wall, and the finger is required to reach four scattered target positions in the space in sequence. The initial design of the finger is not long enough to reach two of them. Thus it requires the algorithm to optimize to finger to be longer in order to reach all four points. The cost $\mathcal{L}$ of this task is computed by:

$$\mathcal{L} = \sum_{t=1}^{T} c_u \|u_t\|^2 + c_p \|p_t - \hat{p}_t\|$$

$$\text{with } c_u = 0.1, c_p = 10$$

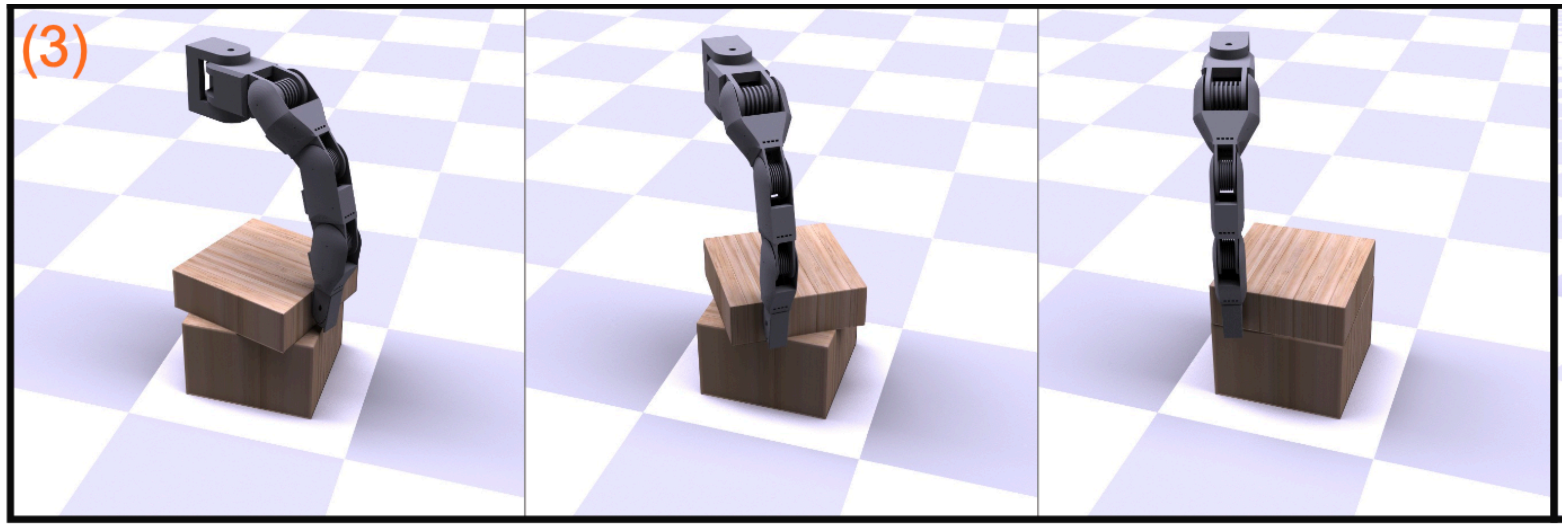


2) *Flip Box*: This task requires the finger to flip a heavy box by 90° and be as energy-efficient as possible. The bottom front edge of the heavy box is attached to the ground with a revolute joint. This task is more difficult than the previous one since the finger needs to interact with the box, which involves a rich amount of contacts and requires leverage of the contact force to flip the heavy box. The cost of this task is computed by:

$$\mathcal{L} = \sum_{t=1}^{T} c_u \|u_t\|^2 + c_{touch}\|p_t - p_{touch}\|^2 + c_{flip}\|\theta_t - \frac{\pi}{2}\|^2$$

$$\text{with } c_u = 5, c_{touch} = \begin{cases} 1 & t < T/2 \\ 0 & t \geq T/2 \end{cases}, c_{flip} = 50$$

where $u_t \in [-1, 1]$ is the action at time t, p_t is the finger tip position at time t, p_{tourch} is a point on the back surface of the box, and θ_t is the rotation angle of the box at time t. The second term is designed to encourage the manipulator to touch the box and provide some simple heuristics of solving the task.

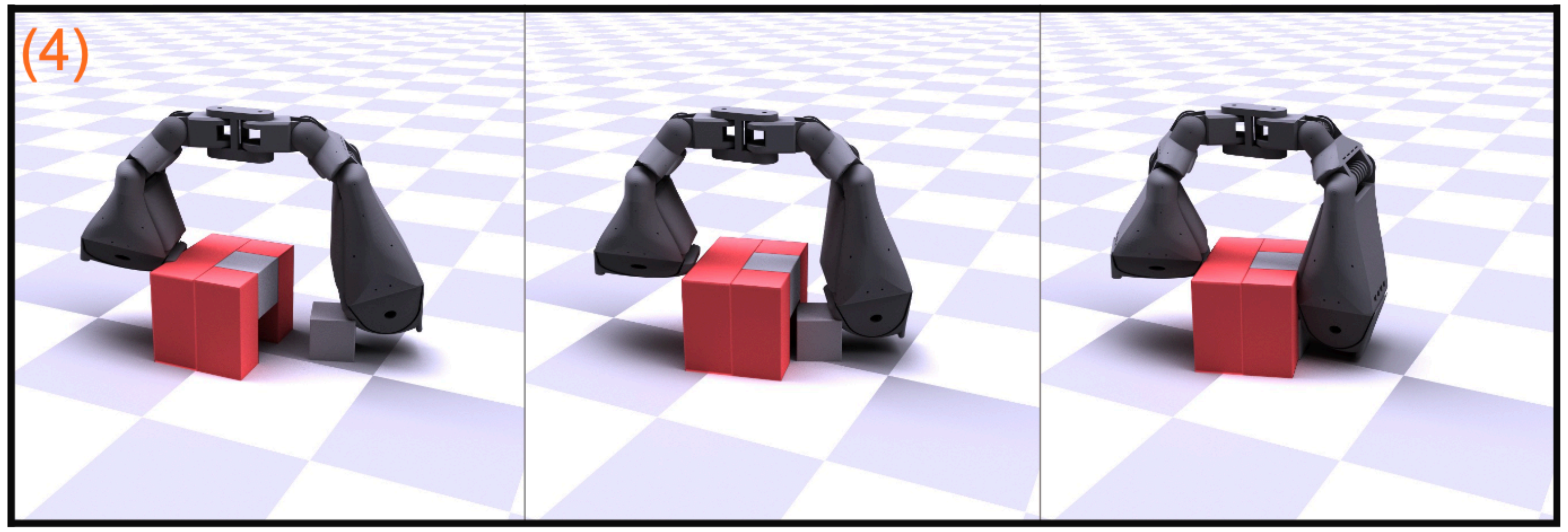


3) *Rotate Rubik's Cube*: The finger is required to rotate the top layer of a Rubik's cube by 90°. The bottom of the Rubik's cube is fixed on the ground. In this task, there is no clear heuristics in the objective function to guide the finger to touch a specific place on the cube, so the finger needs to be optimized to find the correct strategy. The cost of this task is defined by:

$$\mathcal{L} = \sum_{t=1}^{T} \left(c_u \|u_t\|^2 + c_{touch} \|p_t - p_{cube}\|^2\right) + c_{rotate} \|\theta_T - \frac{\pi}{2}\|^2$$

$$\text{with } c_u = 5, c_{touch} = 0.1, c_{rotate} = 1000$$

where u_t and p_t are same as previous tasks, p_{cube} is the center of the Rubik's cube, and θ_T is the rotation angle of the top layer of the cube at the last time step.



4) *Assemble*: In this task, two fingers need to collaborate together to push and insert a small box into its movable mount. The box and the hole on the mount have similar sizes, making the task much more challenging and requiring high-accuracy manipulation. Moreover, the movable mount needs to stay as close as possible to the original position to mimic a restricted working platform environment. The two fingers are mounted on a manipulator base that is allowed to move in the horizontal plane. The cost of this task is computed as:

$$\mathcal{L} = \sum_{t=1}^{T} c_{mount}\|p_t^M - p_0^M\|^2 + c_{touch}(\|p_t^{left} - p_t^M\|^2 + \|p_t^{right} - p_t^{box}\|^2)$$
$$+ c_p\|p_t^{box} - p_{hole}\|^2 + c_{rotation}\|\theta_t^M - \theta_t^{box}\|^2$$
$$\text{with } c_{mount} = 15, c_{touch} = 1, c_p = 5, c_{rotation} = 50$$

where p_t^M is the position of the movable mount at time t, p_t^{left} and p_t^{right} are the finger tip positions of left finger and right finger at time t, p_t^{box} is the position of the small box at time t, θ_t^M and θ_t^{box} are the rotation angle of the mount and the box. The first term is used to penalize moving the mount too far away from the original place, the second term is designed to encourage the fingers to touch on the objects (but not indicate any specific position on the object), and the third term and the last term together is to measure how well the box is inserted into the mount.

Results

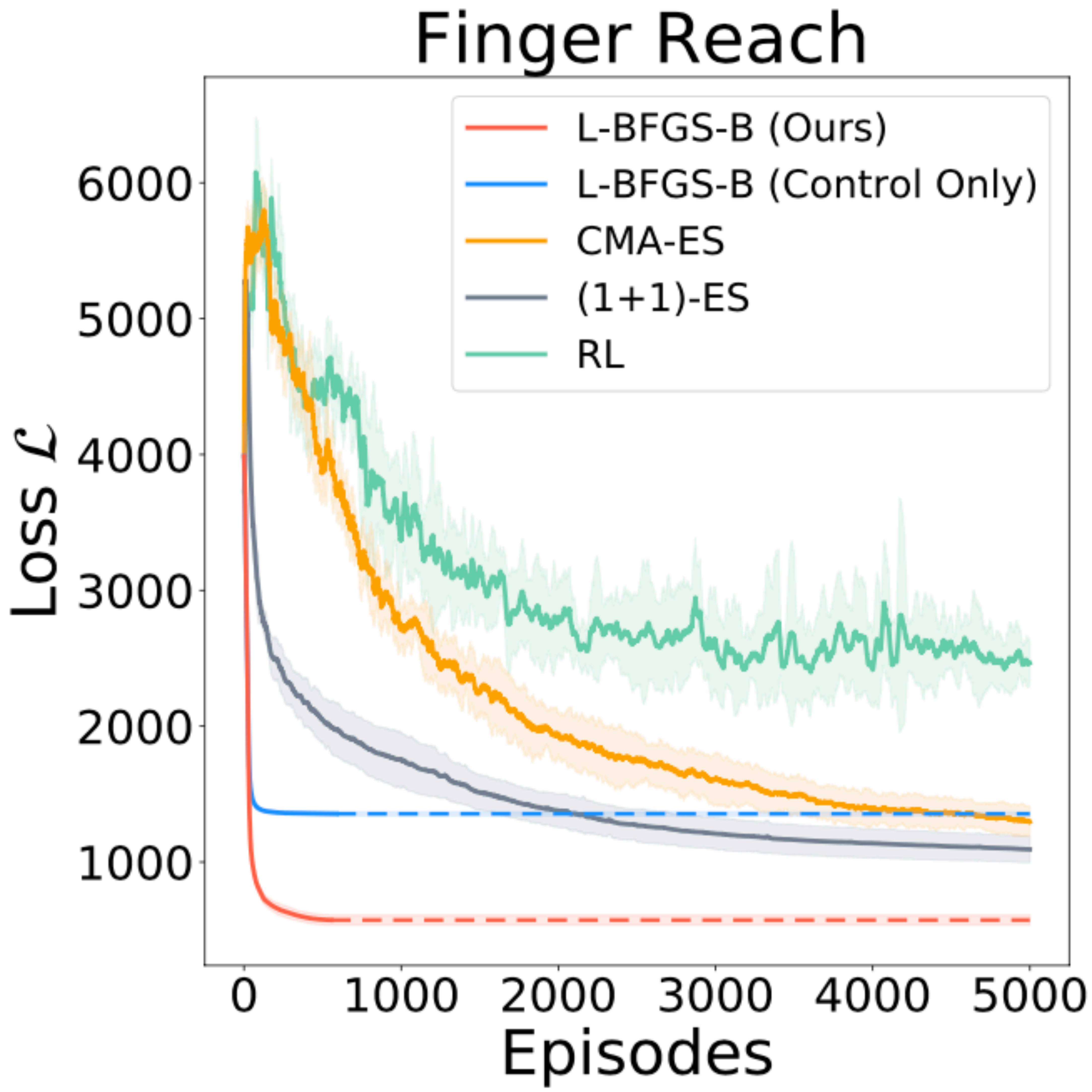


Fig. 6. **Optimization curves comparison**. We run all the methods on all tasks 5 times with different random seeds. Mean and standard deviation in the loss objective are reported. The horizontal axis of each plot is the number of simulation episodes, and vertical axis is the objective loss value. L-BFGS-B optimization can terminate early once it satisfies the termination criterion. For better visualization, we extend the actual learning curves that use L-BFGS-B horizontally using dotted lines. We also smooth out the curves with a window size of 10.

Results show benefit to analytic differentiation

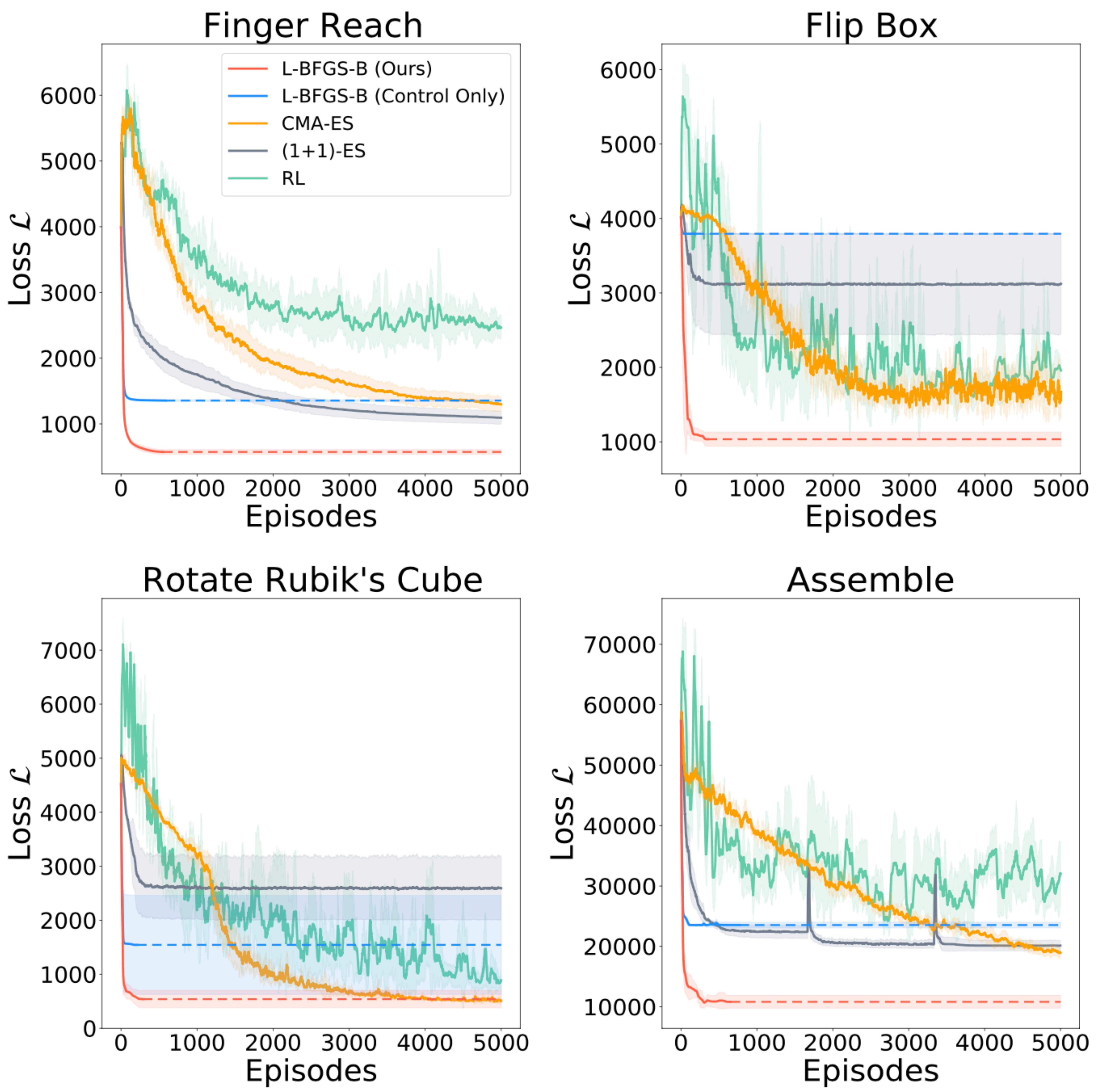


Fig. 6. **Optimization curves comparison**. We run all the methods on all tasks 5 times with different random seeds. Mean and standard deviation in the loss objective are reported. The horizontal axis of each plot is the number of simulation episodes, and vertical axis is the objective loss value. L-BFGS-B optimization can terminate early once it satisfies the termination criterion. For better visualization, we extend the actual learning curves that use L-BFGS-B horizontally using dotted lines. We also smooth out the curves with a window size of 10.