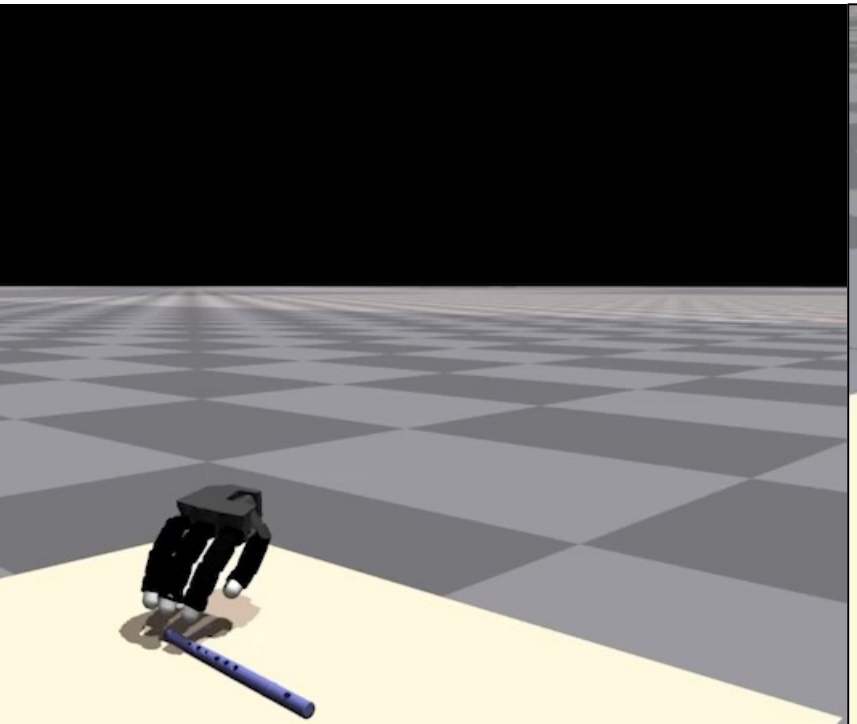
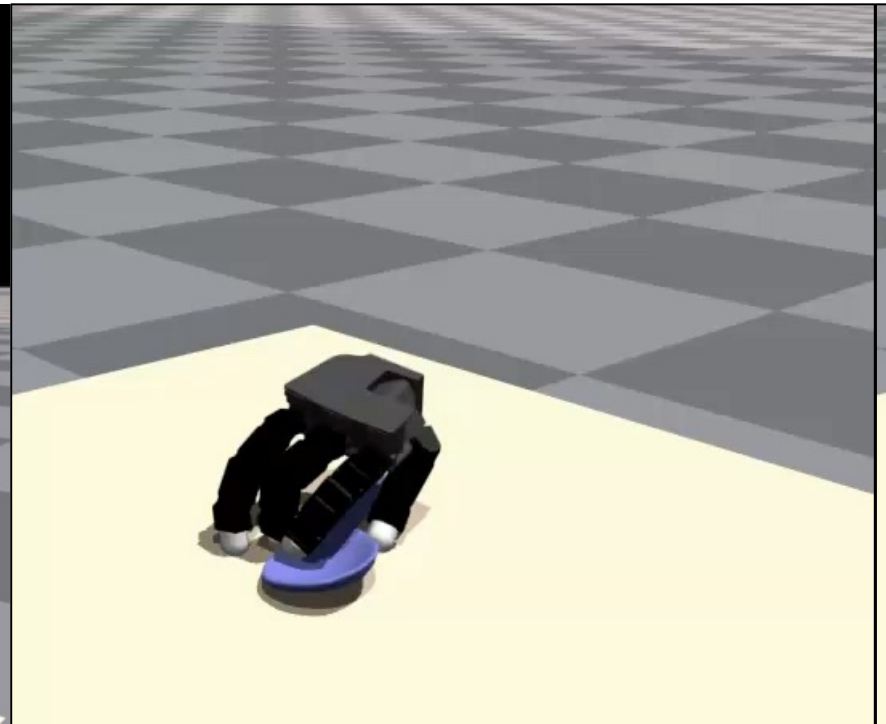
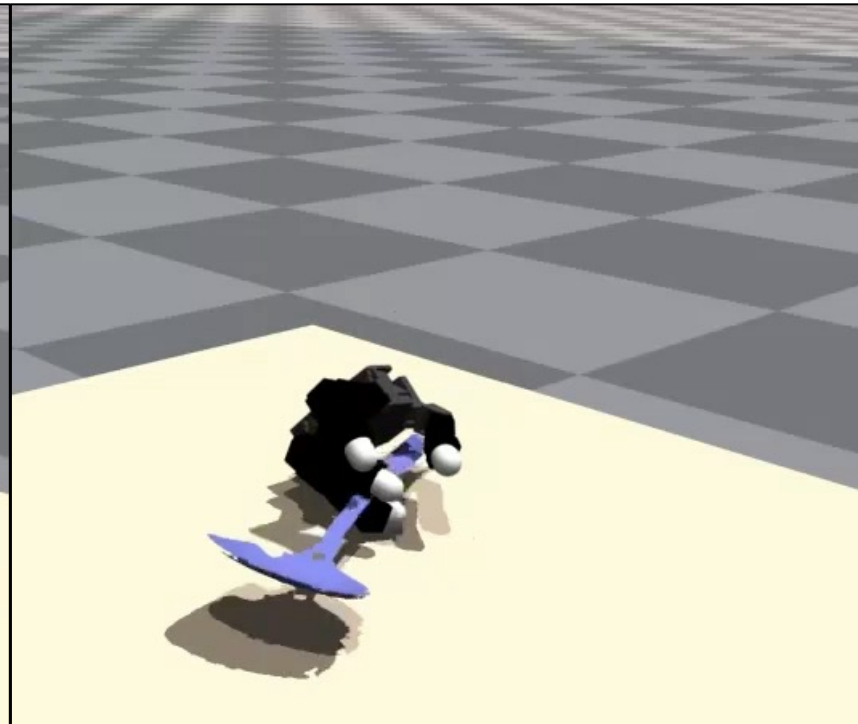
DexTrack

Towards Generalizable Neural Tracking Control for *Dexterous Manipulation from Human References*

https://projectwebsite7.github.io/gene-dex-manip/



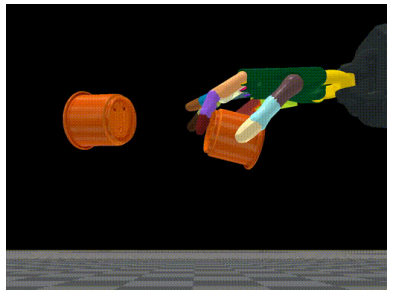




Dexterous Manipulation Skill Acquisition

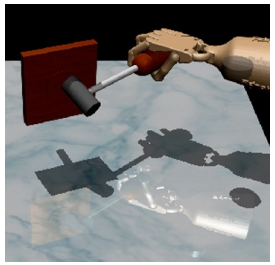
Diverse Manipulation Skills

Grasp

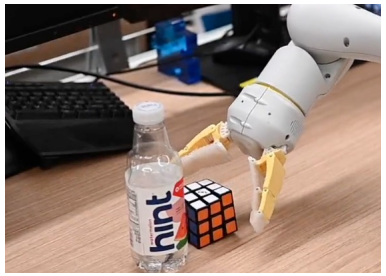


Re-orientation

A *single* manipulation policy capable of solving each task



Tool-using



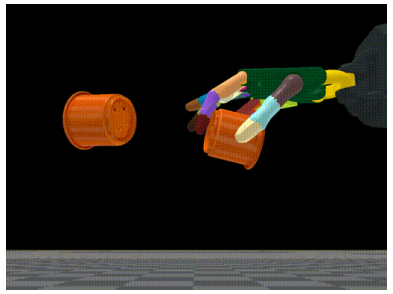
Pick-and-place

Can we learn a single model to master all kinds of diverse skills?

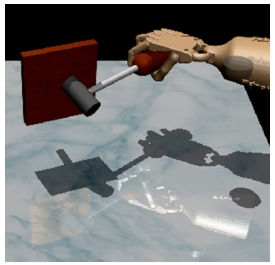
Dexterous Manipulation Skill Acquisition

Diverse Manipulation Skills

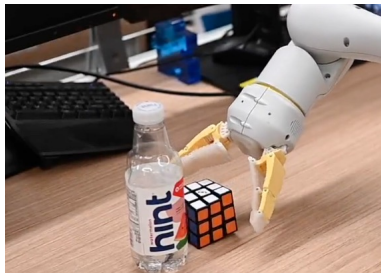
Grasp



Re-orientation



Tool-using



Pick-and-place

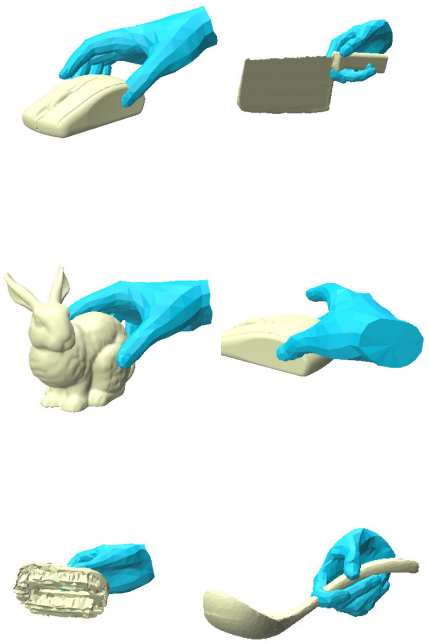
Can we learn *a single model* to master all kinds of *diverse skills*?

Task-specific rewards
Known system dynamics
Restricted contact modeling

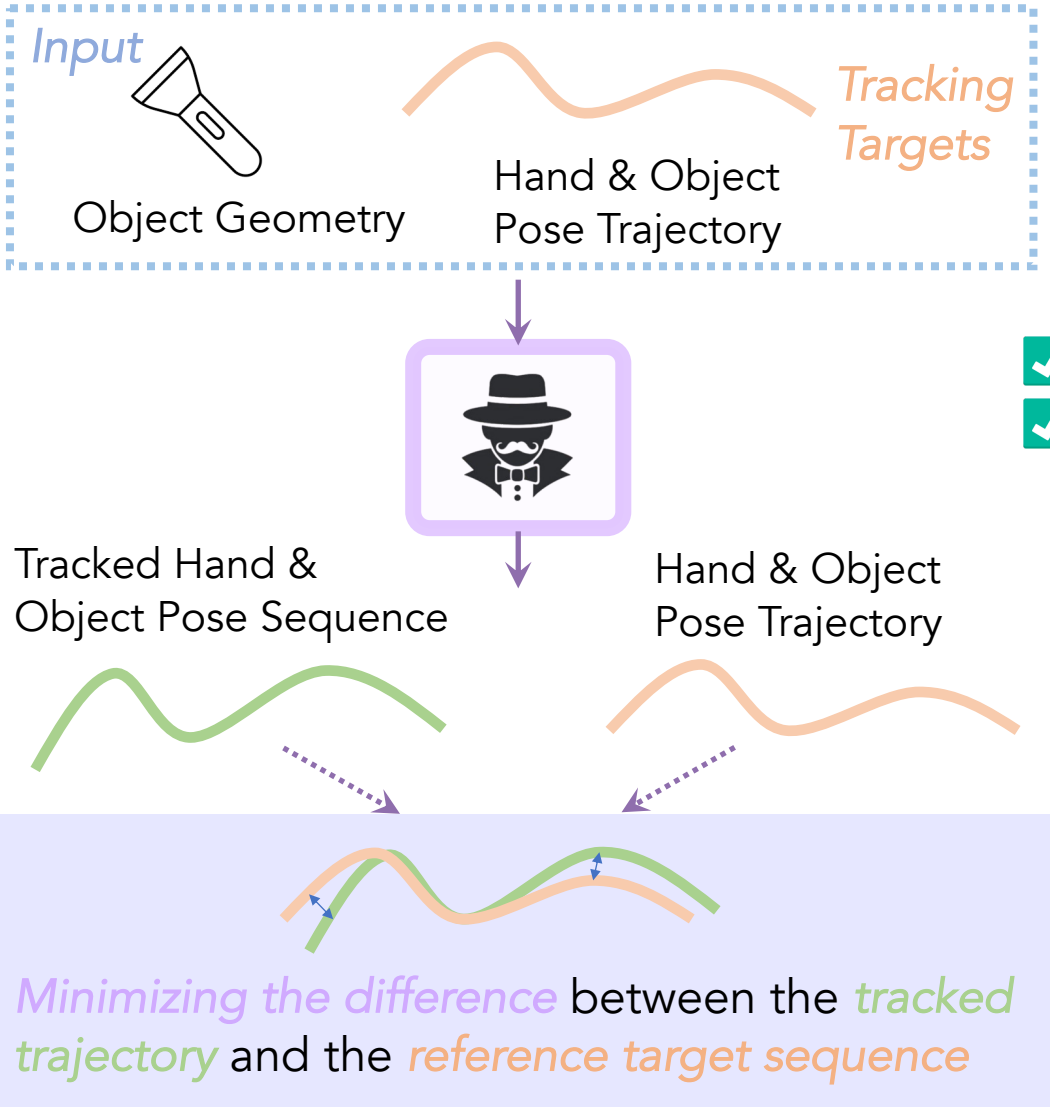
They are not generalizable designs!

- Isolating the *control problem* from the planning problem
- Learning a *generalizable tracking controller* from abundant *human references*

Generalizable *Neural Tracking Control for Dexterous Manipulations*



Abundant human references



- ✓ *No task-specific rewards*
- ✓ *Diverse task space & abundant data sources*

A *universal* manipulation task description

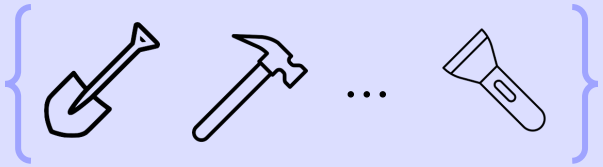
*A **Generalizable** Neural Tracking Controller?*

Complex Dynamics

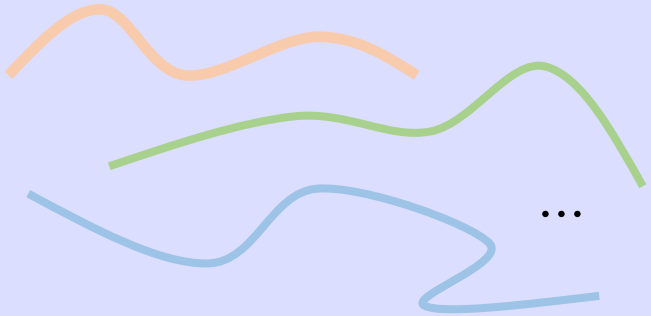
Precise Control

Even *Single* Trajectory Tracking is Difficult to Solve!

Tracking Control



Diverse objects



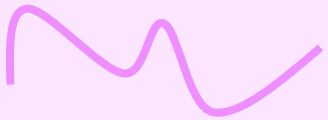
Various manipulation trajectories

A single model for all tasks

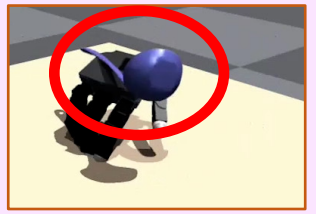
Versatility

Generalize to unseen tracking references and being robust to large noises

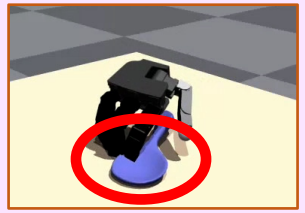
New objects



Novel interactions



Unreasonable targets

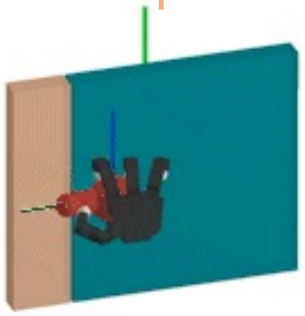


Penetrations

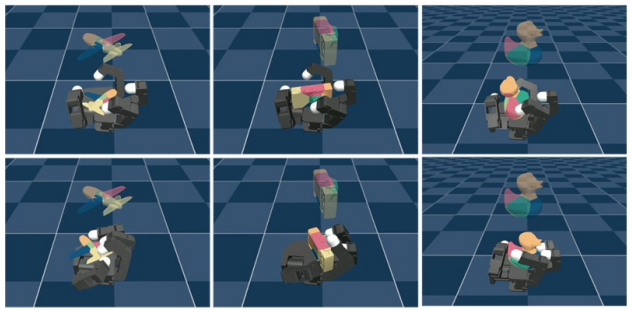
Generalizability & Robustness

Approach: Model-free RL with Learning from Demonstrations

Developing an effective optimization strategy with accurate system modeling for dexterous manipulation is difficult!!



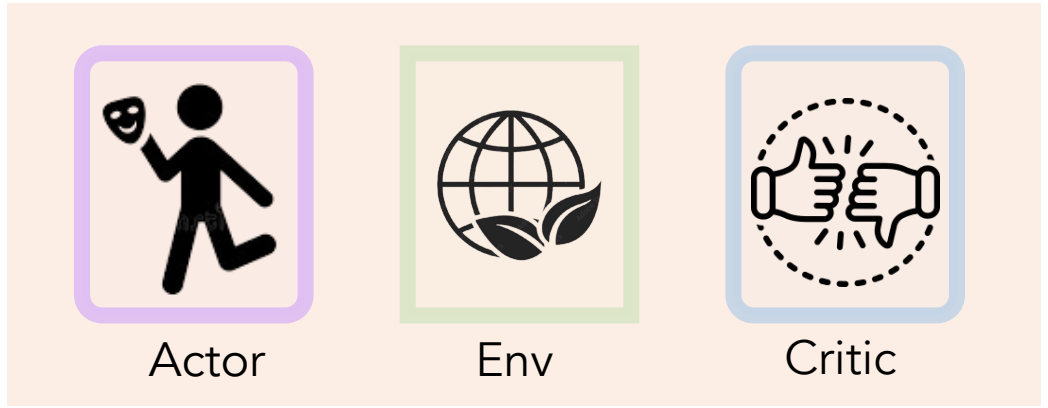
[T. Pang, 2023]



[W. Jin, 2024]

Quasi-Dynamic Contact Model

(Simplified and reduced-order models)

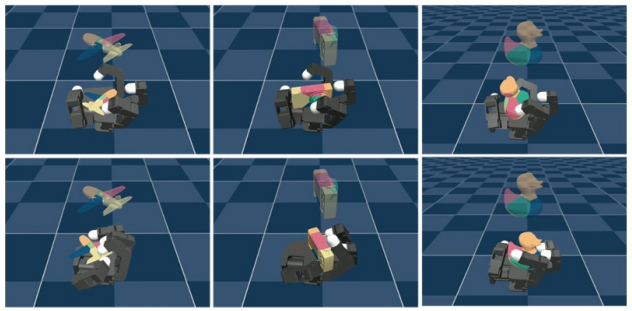


Model-free RL requires no system modeling

Approach: Model-free RL with Learning from Demonstrations

Developing an effective optimization strategy with accurate system modeling for dexterous manipulation is difficult!!

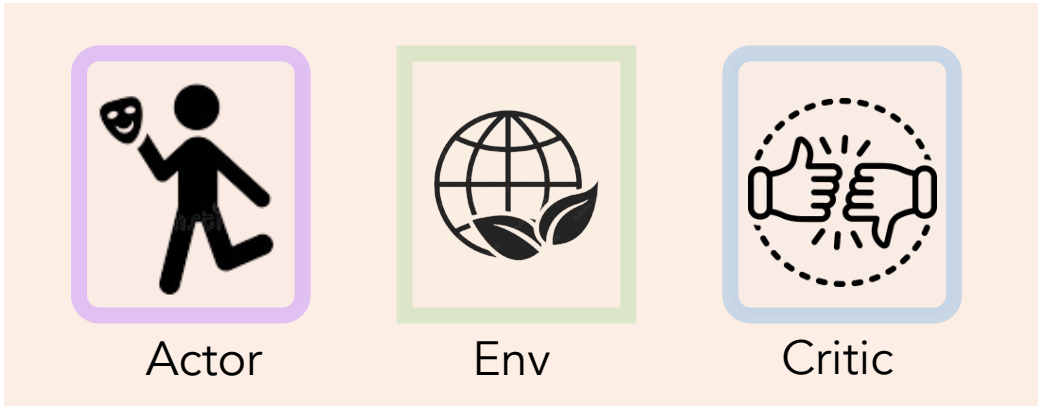
[T. Pang, 2023]



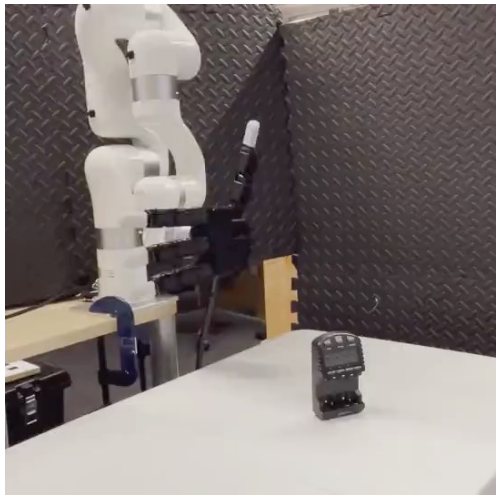
[W. Jin, 2024]

Quasi-Dynamic Contact Model

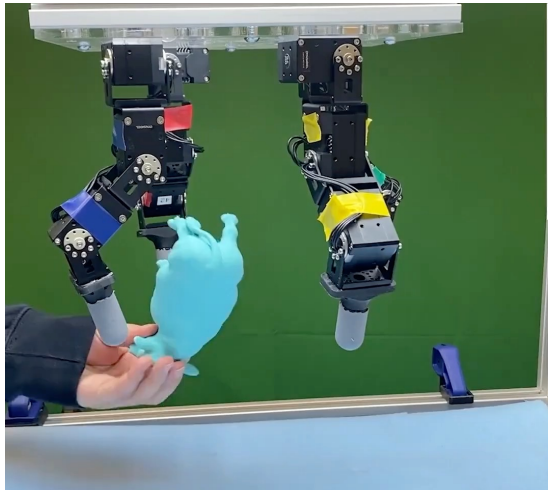
(Simplified and reduced-order models)



Model-free RL requires no system modeling and has demonstrated strong capability in dexterous manipulations.



[Y. Qin, 2022]

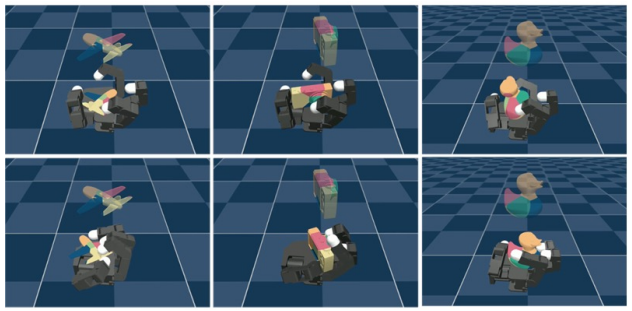


[T. Chen, 2023]

Approach: Model-free RL with Learning from Demonstrations

Developing an effective optimization strategy with accurate system modeling for dexterous manipulation is difficult!!

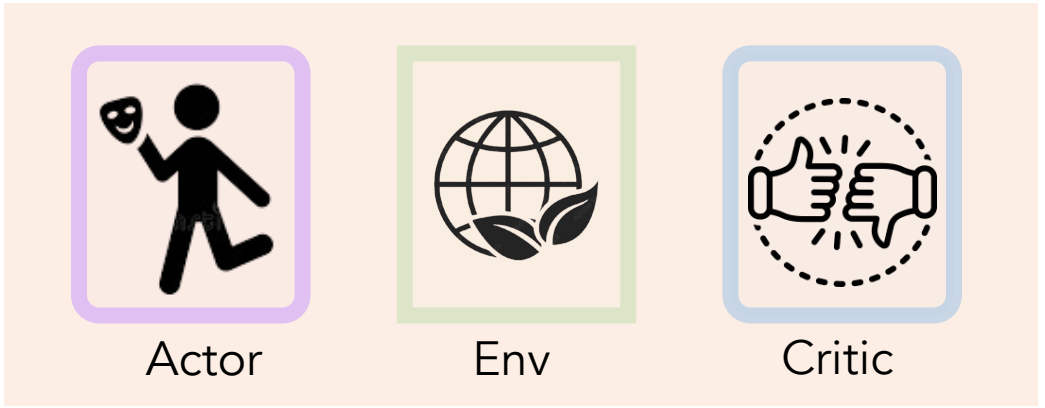
[T. Pang, 2023]



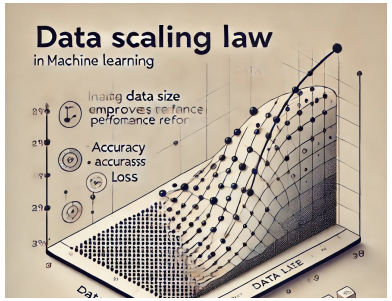
[W. Jin, 2024]

Quasi-Dynamic **Contact Model**

(Simplified and reduced-order models)



Model-free RL requires no system modeling but it still faces challenges in training *one single model* to solve *diverse tracking tasks*.



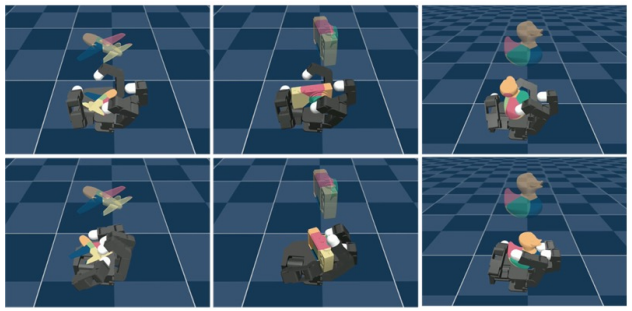
Internet Data

- ☑ Data-driven methods are *scalable* and can benefit from *guidance from high-quality data*
- ☑ RL can be *robust to disturbances* and can handle *unexpected situations*

Approach: Model-free RL with Learning from Demonstrations

Developing an effective optimization strategy with accurate system modeling for dexterous manipulation is difficult!!

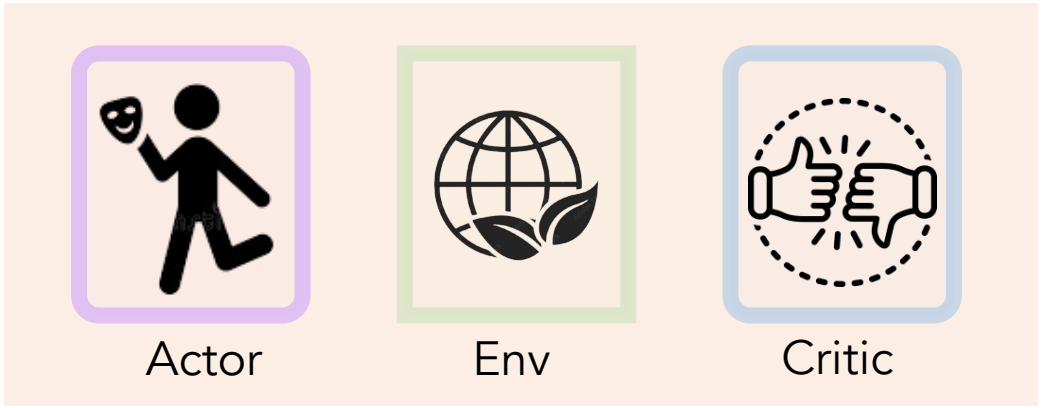
[T. Pang, 2023]



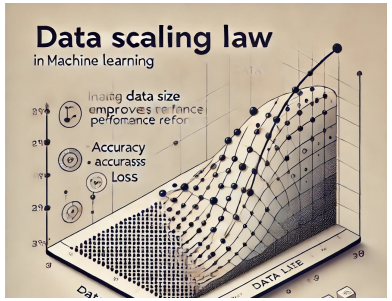
[W. Jin, 2024]

Quasi-Dynamic Contact Model

(Simplified and reduced-order models)



Model-free RL requires no system modeling but it still faces challenges in training *one single model* to solve *diverse tracking tasks*.

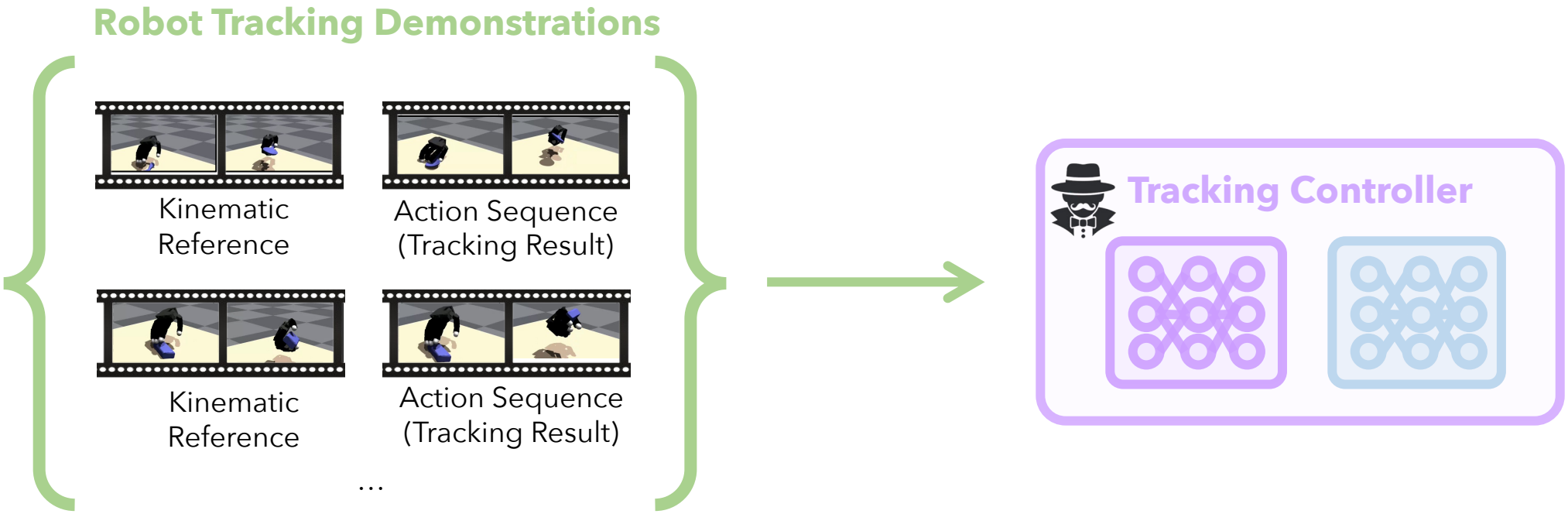


Internet Data

Data-driven Imitation Learning

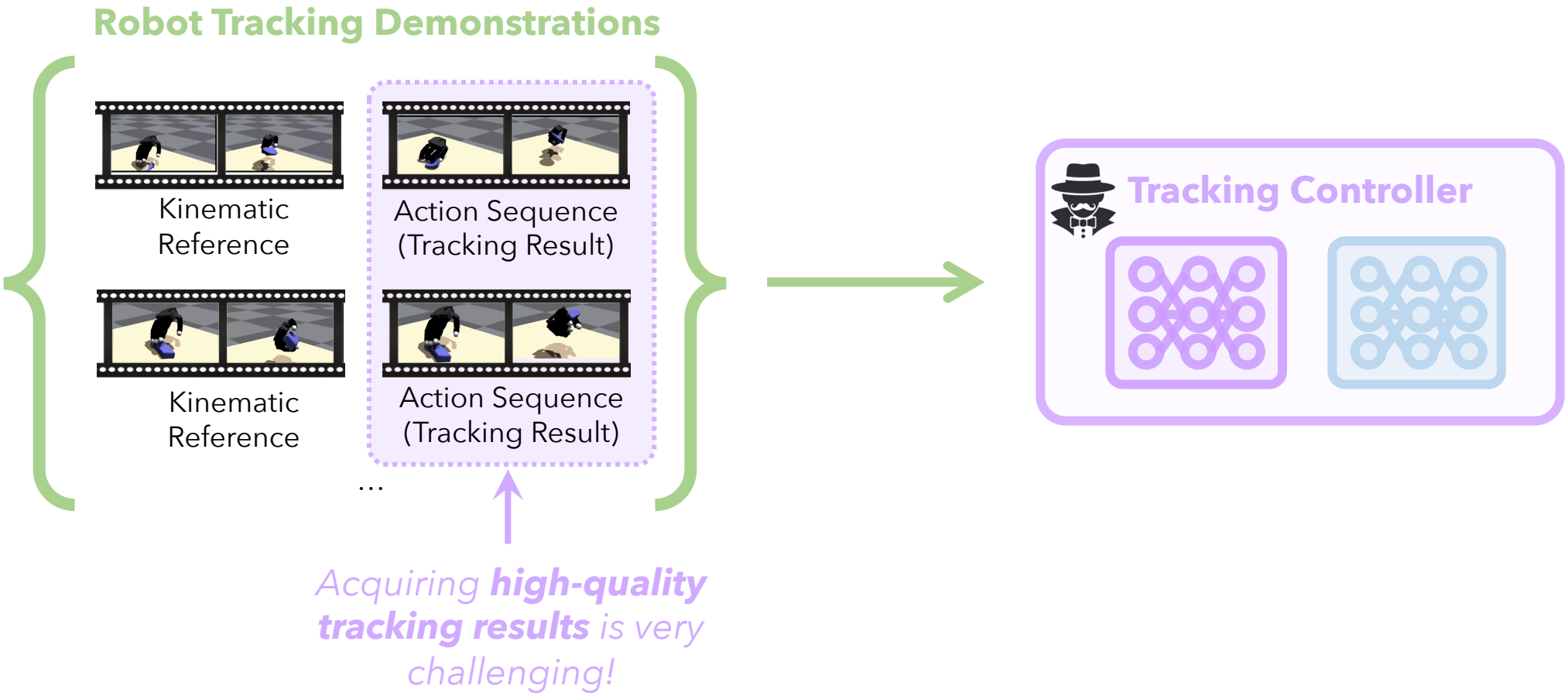
Online Self-improving RL

Approach: Model-free RL with Learning from Demonstrations



Approach: Model-free RL with Learning from Demonstrations

Approach : Improving Tracking Demonstrations using Tracking Controller in a Homotopy Optimization Scheme



Approach : Improving Tracking Demonstrations using Tracking Controller in a Homotopy Optimization Scheme

How to acquire *diverse* and *high-quality* robot tracking demonstrations?

Naïve strategy

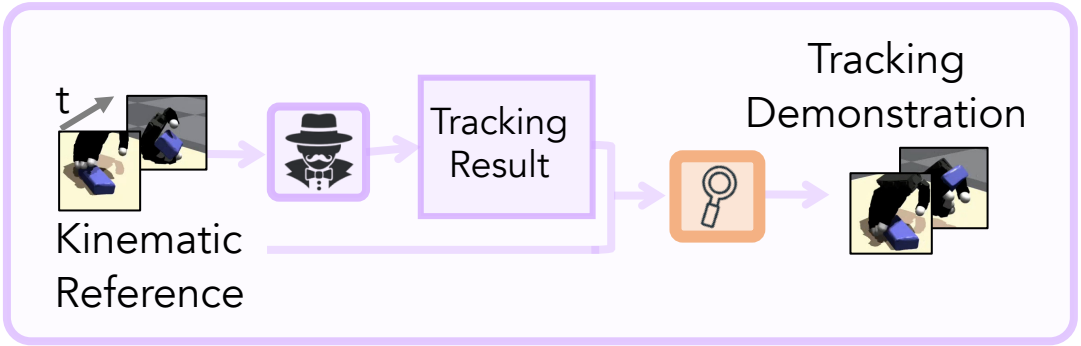
Per-Trajectory Tracker

Solving Per-trajectory tracking via RL

Not an ideal solution!

Perfect single trajectory tracking is difficult, restricting the data *diversity* and *quality*

General Tracking Prior



Improving per-trajectory tracking via using the tracking controller to initialize the policy

Approach : Improving Tracking Demonstrations using Tracking Controller in a Homotopy Optimization Scheme

How to acquire *diverse* and *high-quality* robot tracking demonstrations?

Naïve strategy

Per-Trajectory Tracker

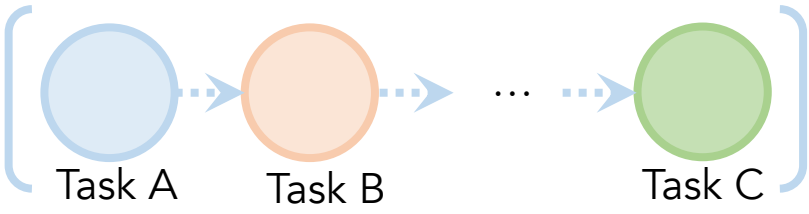
Solving Per-trajectory tracking via RL

Not an ideal solution!

Perfect single trajectory tracking is difficult, restricting the data *diversity* and *quality*

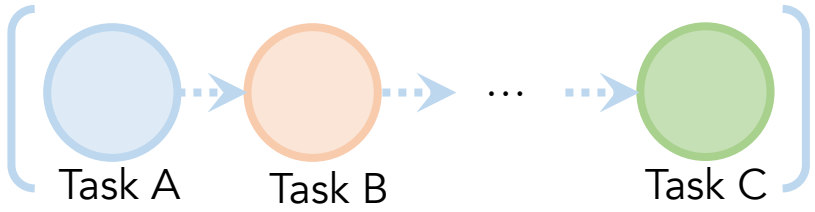
General Tracking Prior

Homotopy Optimization Scheme

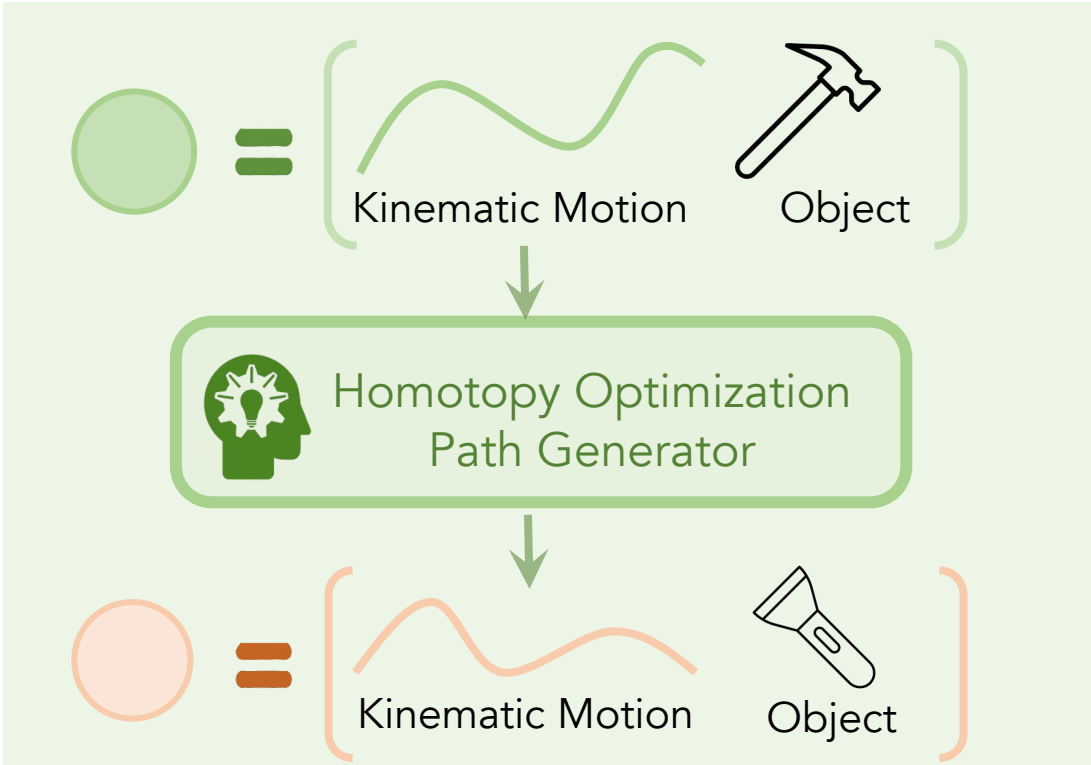


Solve the tracking task C via *an effective optimization path* can improve the per-trajectory tracking result of C

Method Details: Homotopy Path Generation

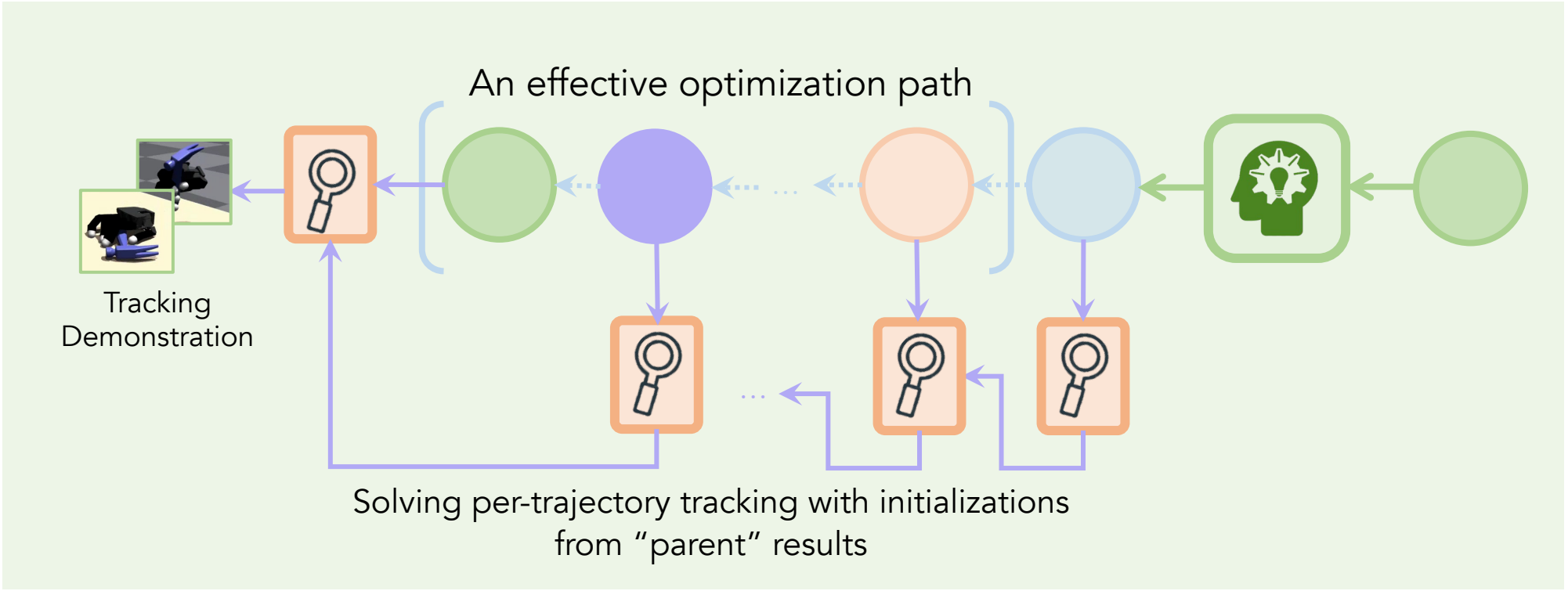


How to model and utilize such cross-tracking task relations?

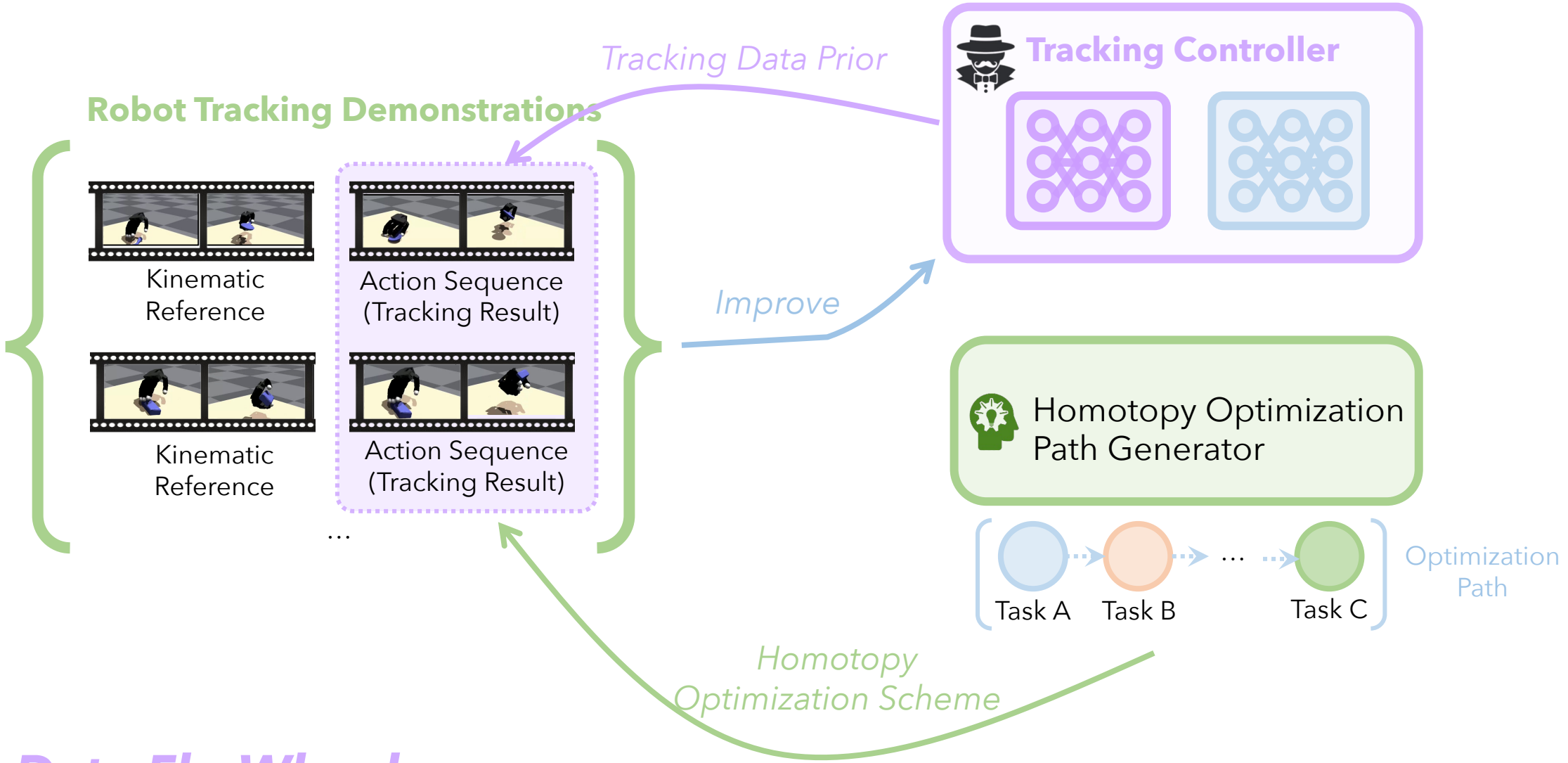


Modeling "parent-child" relations

Method Details: Solving Per-Trajectory Tracking via a Homotopy Optimization Scheme



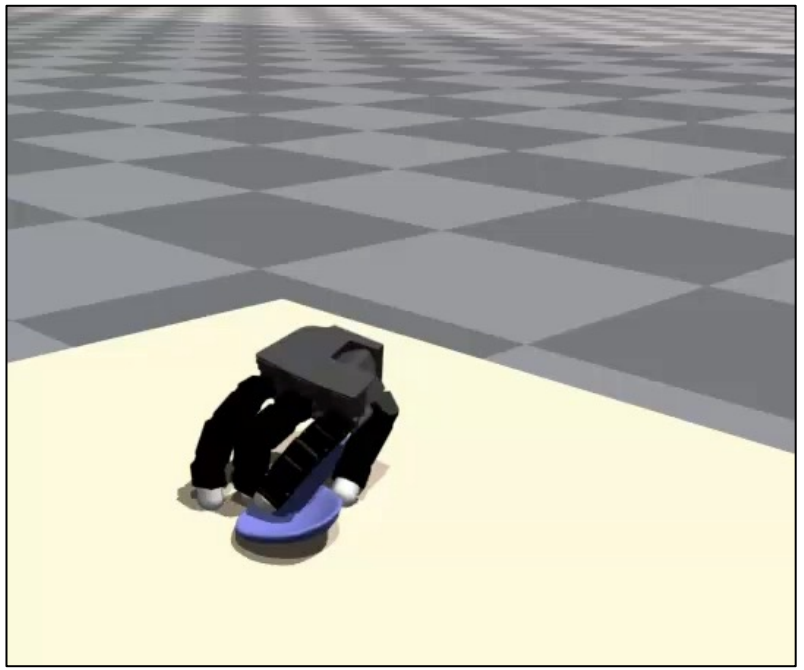
Grow the *Robot Tracking Demonstrations* and Enhance the *Tracking Controller* in a Bootstrapping Manner



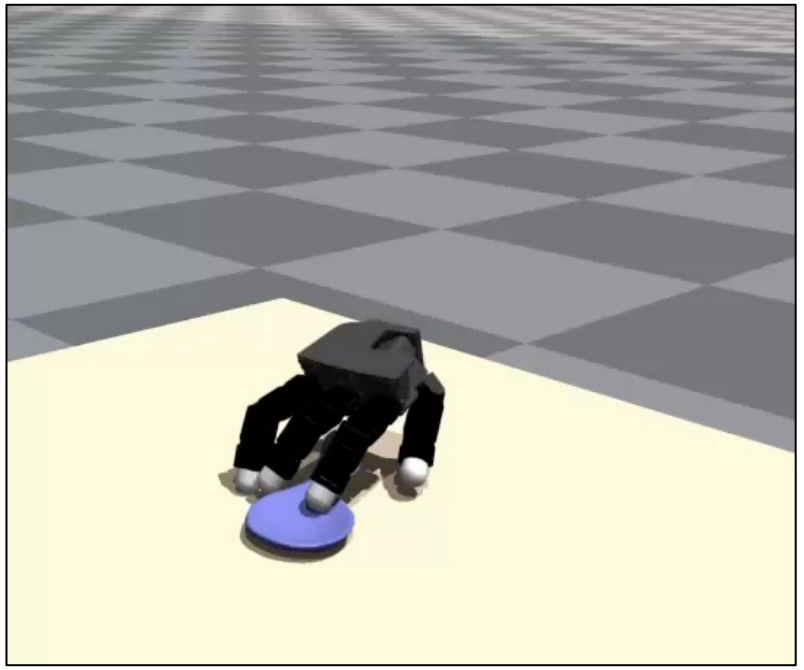
Data Fly-Wheel

Generalizable Dexterous Manipulation
Tracking Control

Retargeted
Kinematic Reference

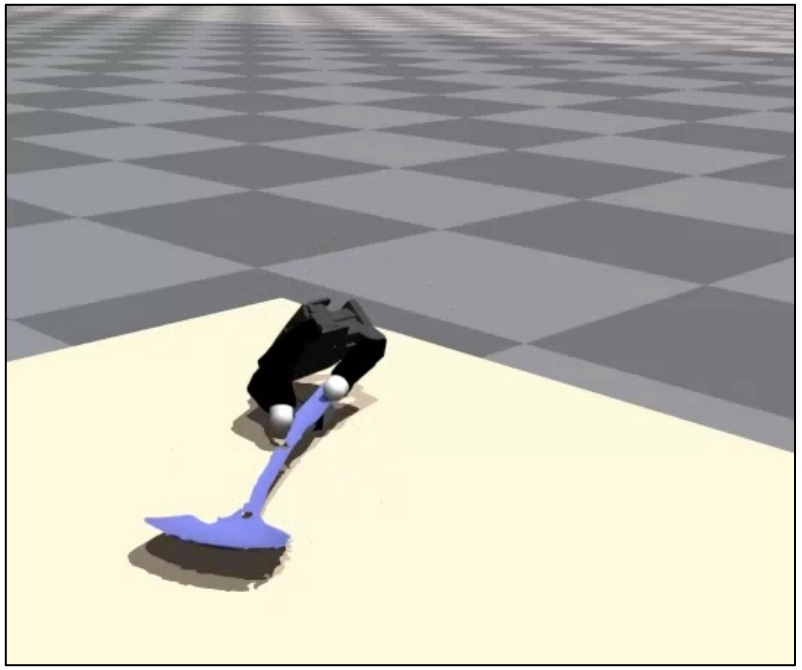


Ours

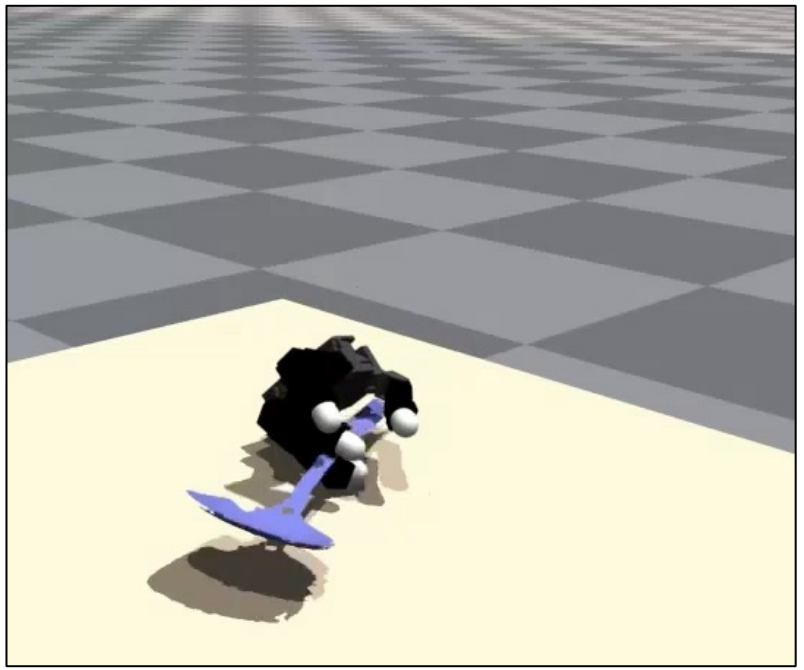


Baseline - PPO
(tracking rew.,
w/o sup.)

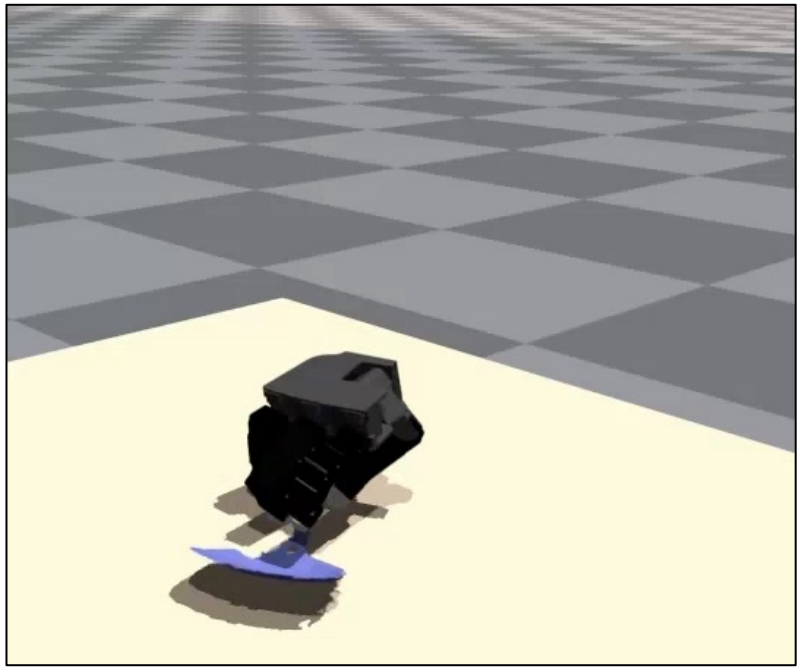
Generalizable Dexterous Manipulation
Tracking Control



Retargeted
Kinematic Reference

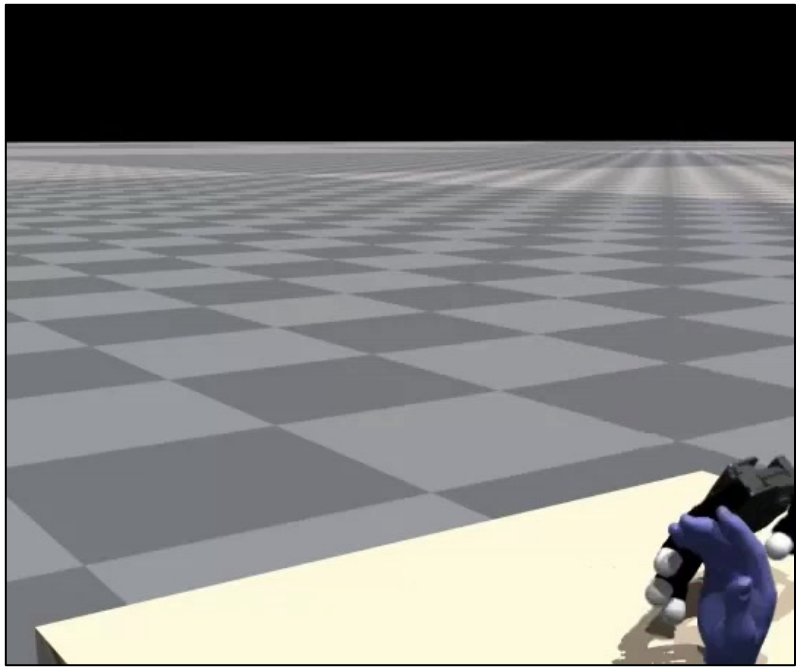


Ours

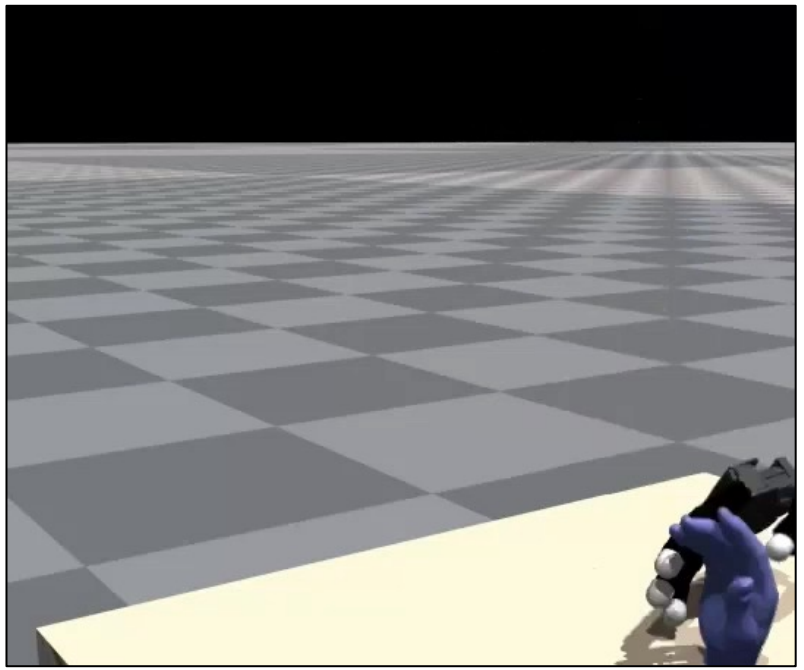


Baseline - PPO
(tracking rew.,
w/o sup.)

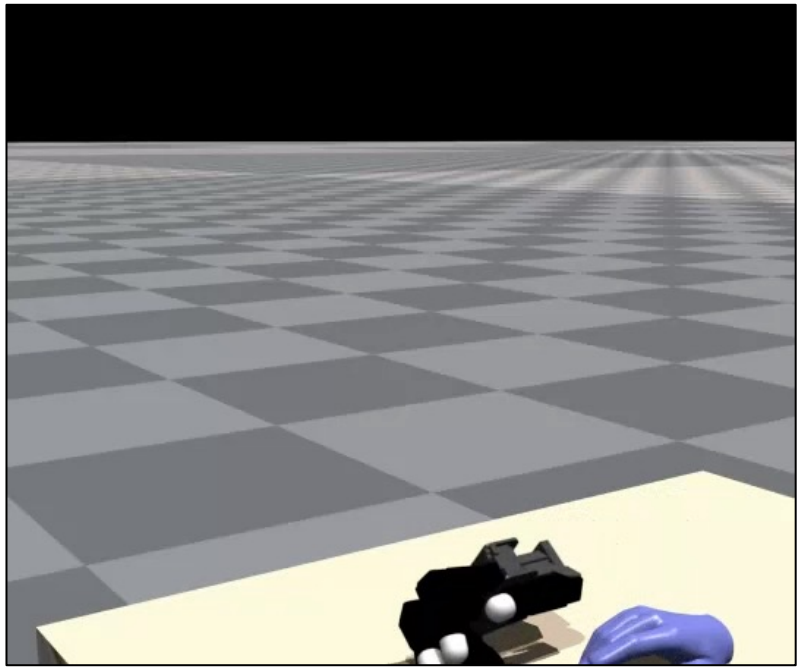
Generalizable Dexterous Manipulation
Tracking Control



Retargeted
Kinematic Reference

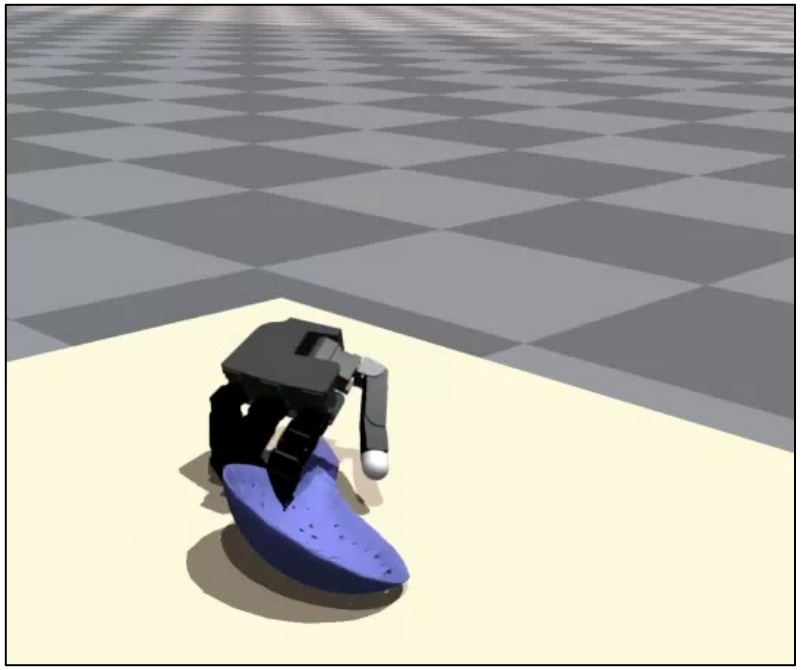


Ours

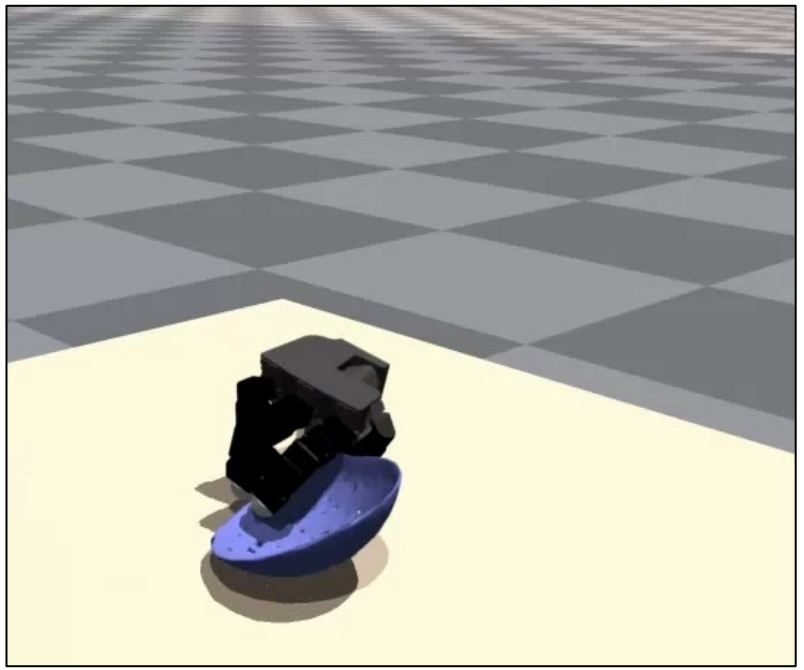


Baseline - PPO
(tracking rew.,
w/o sup.)

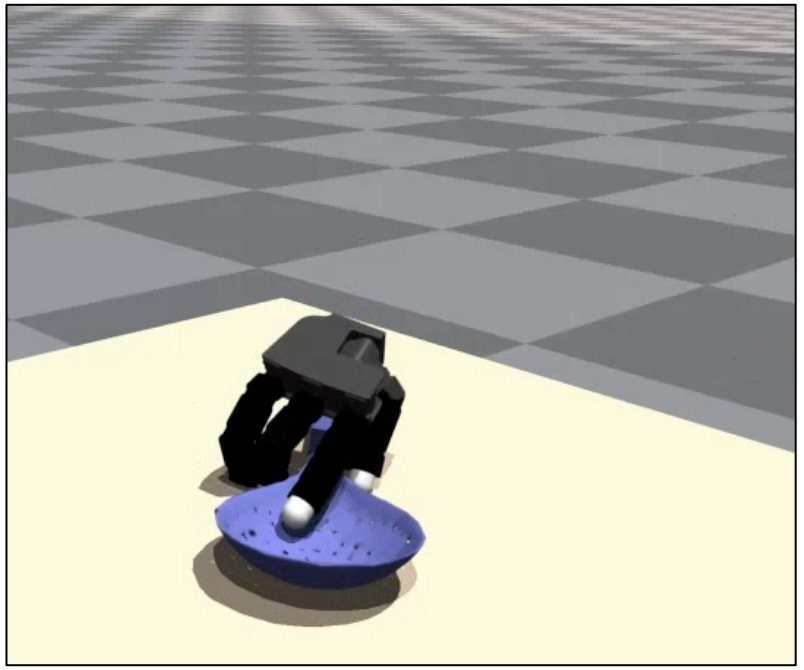
Generalizable Dexterous Manipulation
Tracking Control



Retargeted
Kinematic Reference

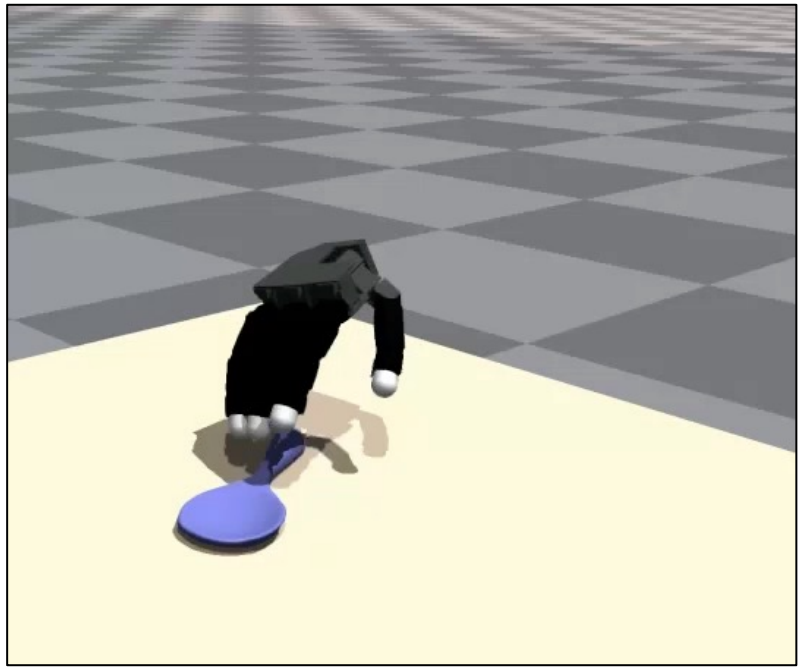


Ours

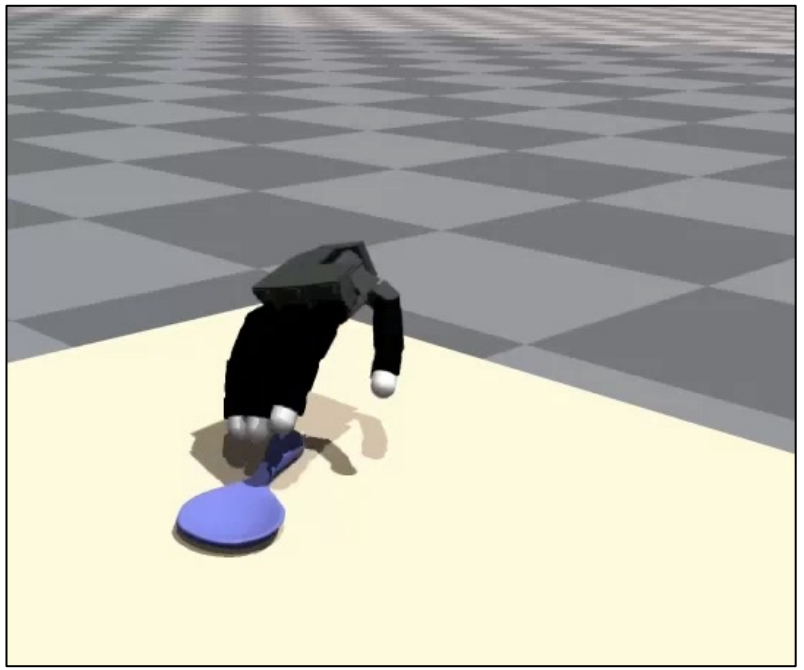


Baseline - PPO
(tracking rew.,
w/o sup.)

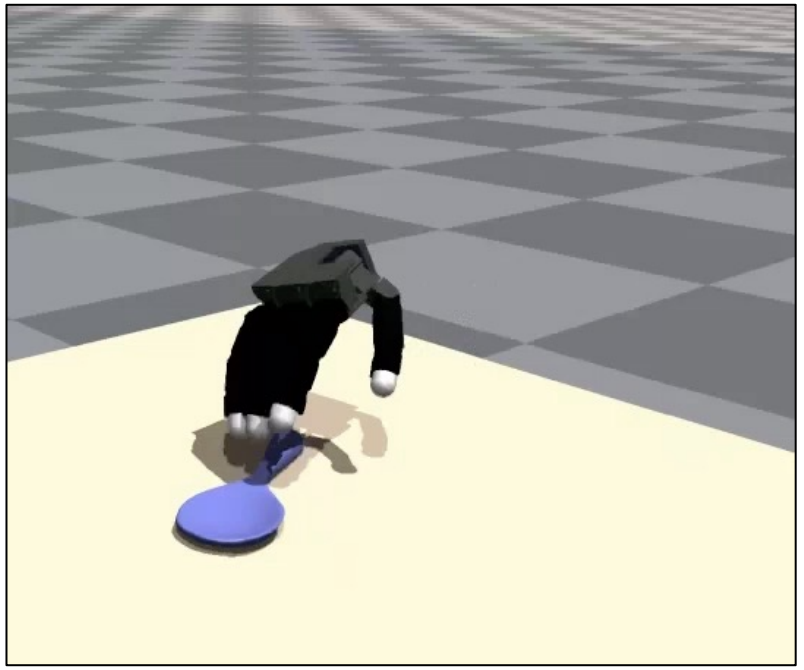
Generalizable Dexterous Manipulation
Tracking Control



Retargeted
Kinematic Reference



Ours

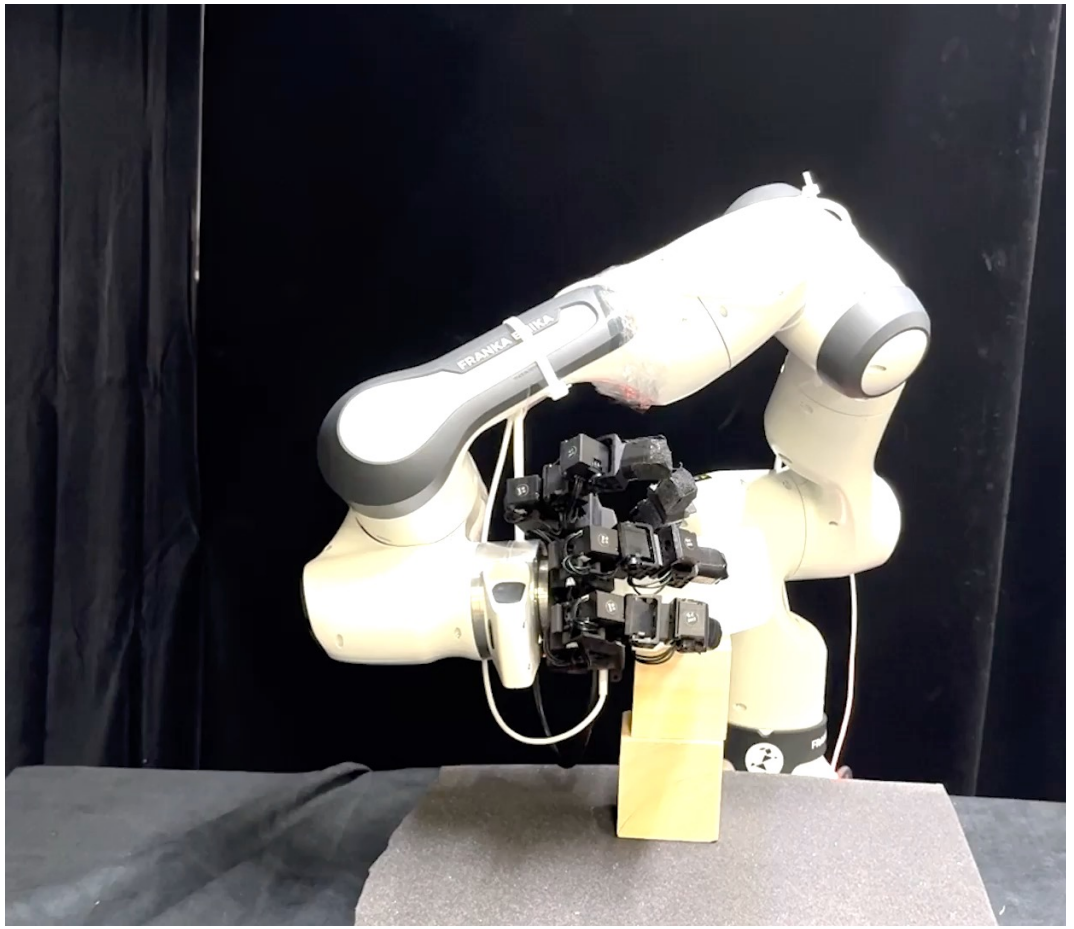


Baseline - PPO
(tracking rew.,
w/o sup.)


Dexterous Manipulation Tracking
from Human References

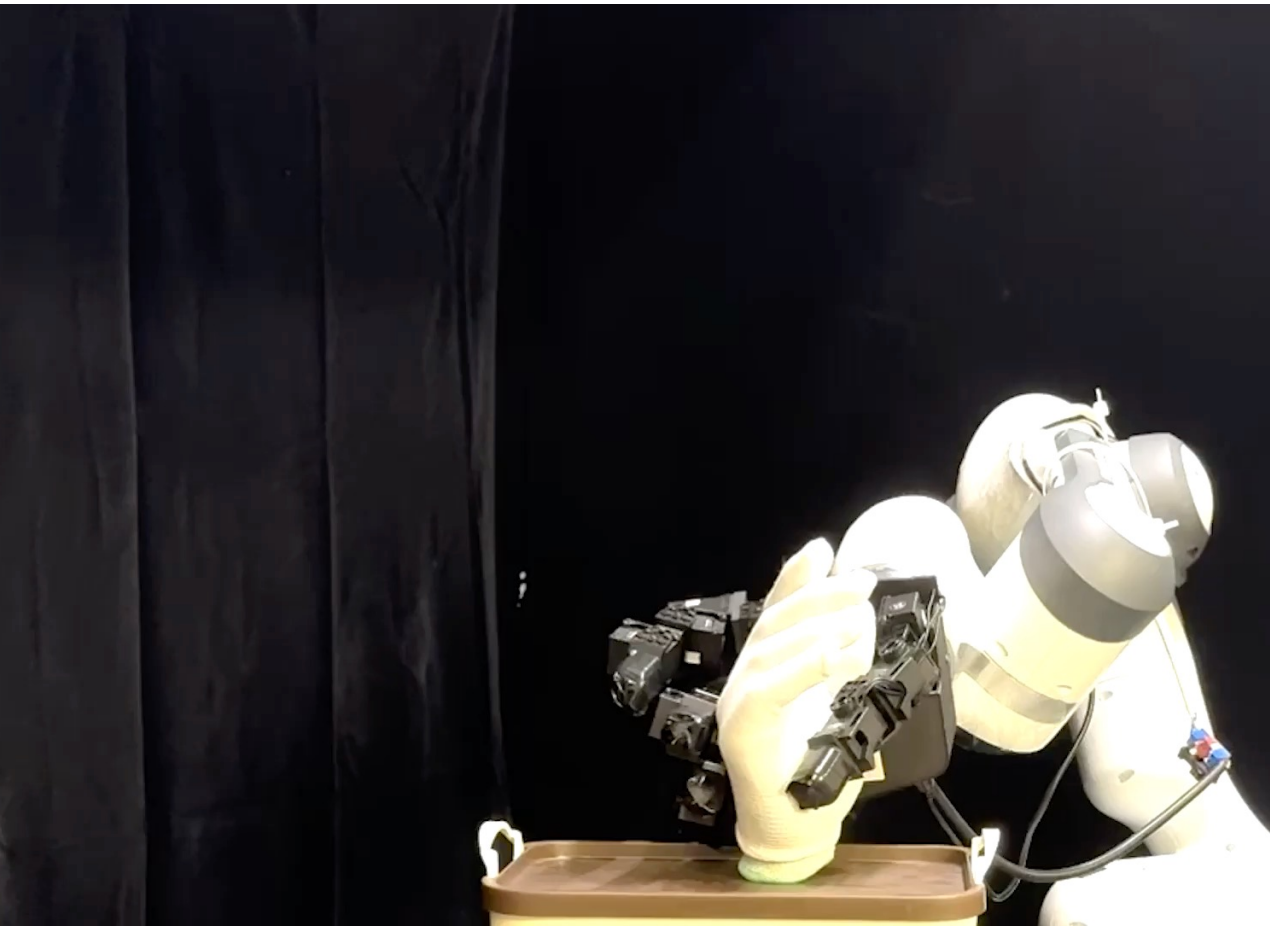
Real-World Evaluations

Ours

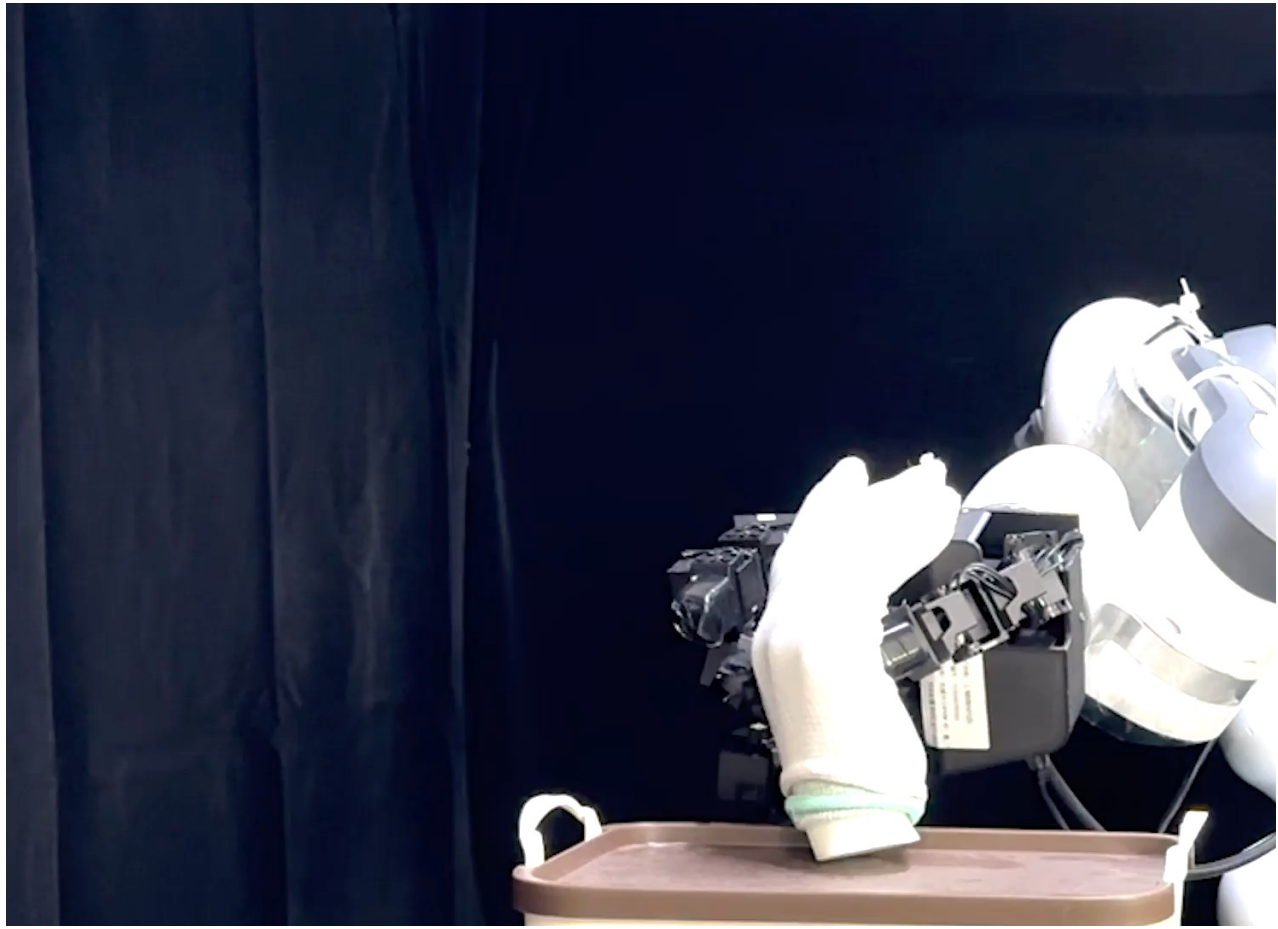


Baseline – PPO
(tracking rew., w/o sup.)

Real-World Evaluations

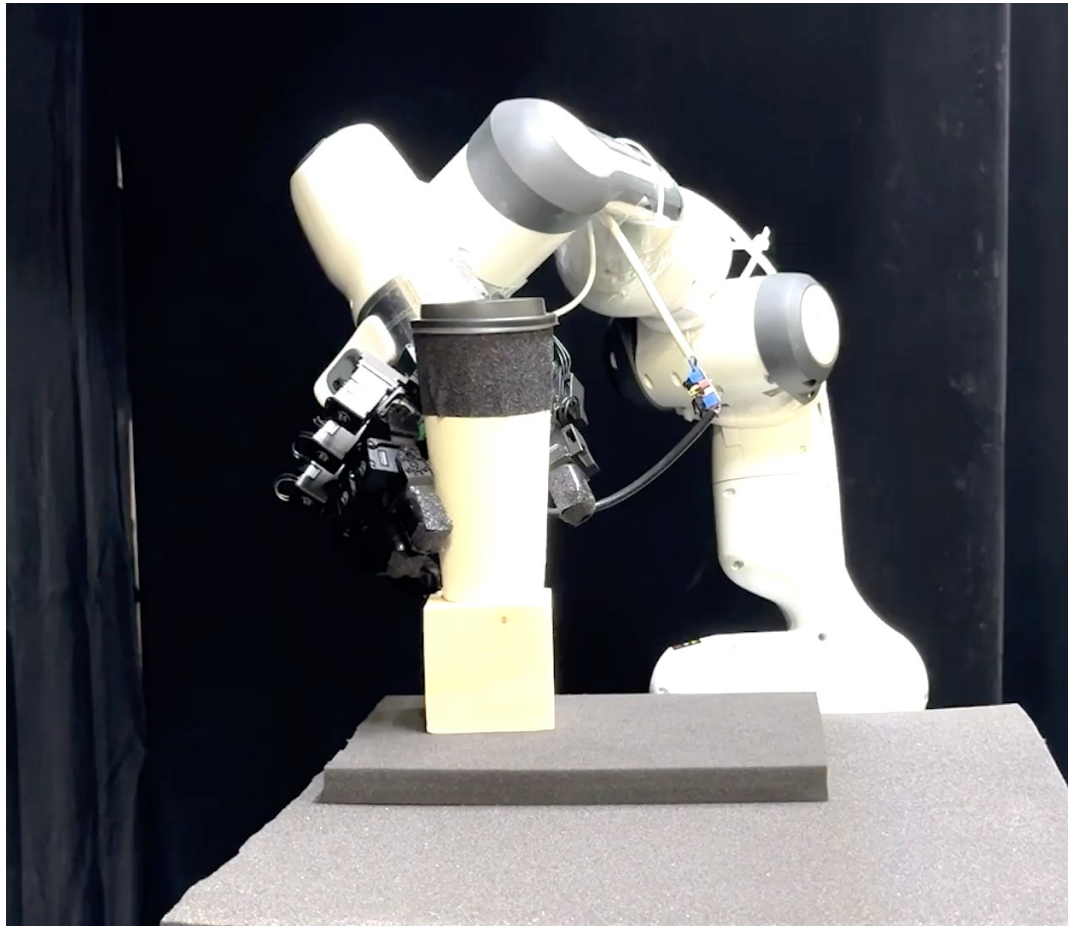


Ours

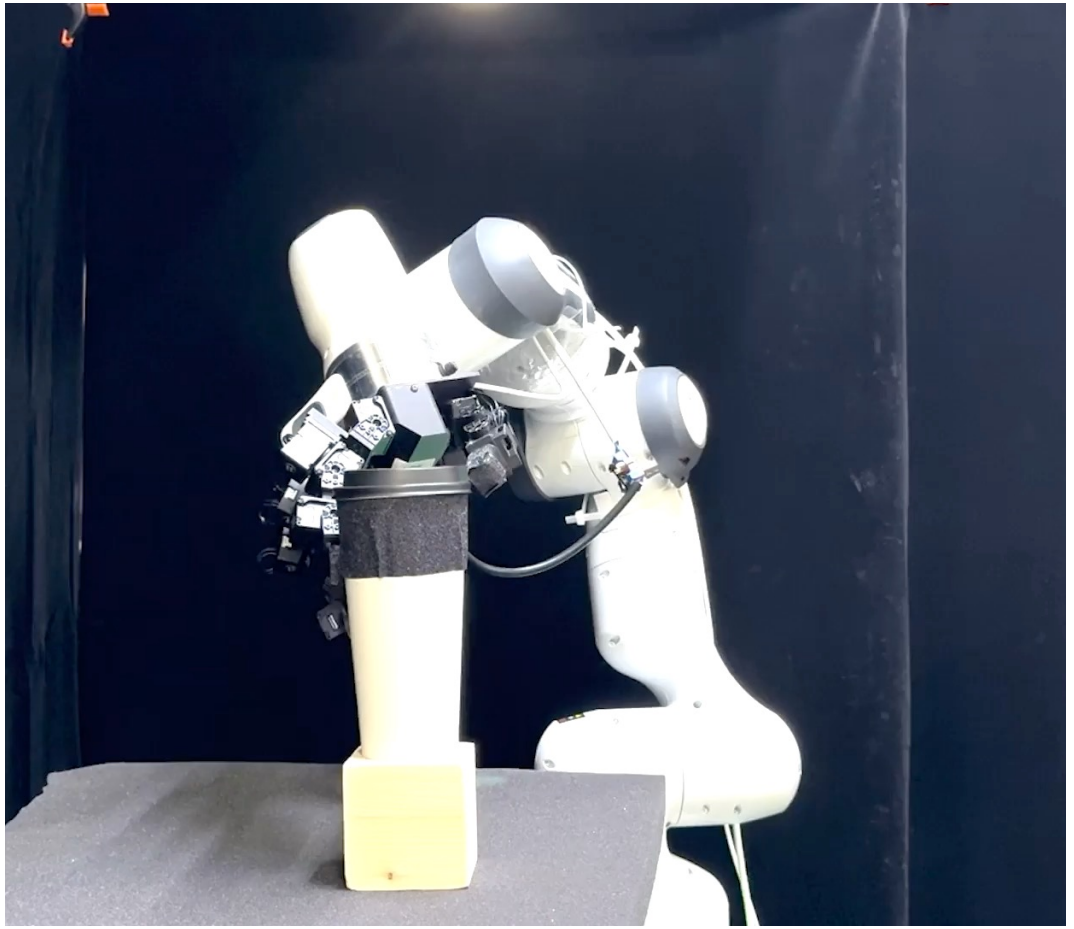


Baseline – PPO
(tracking rew., w/o sup.)

Real-World Evaluations

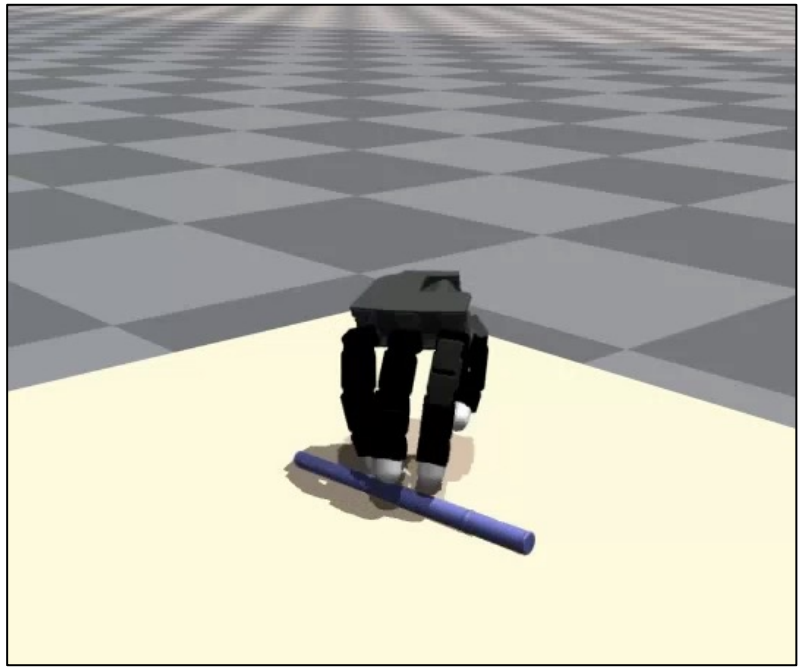


Ours

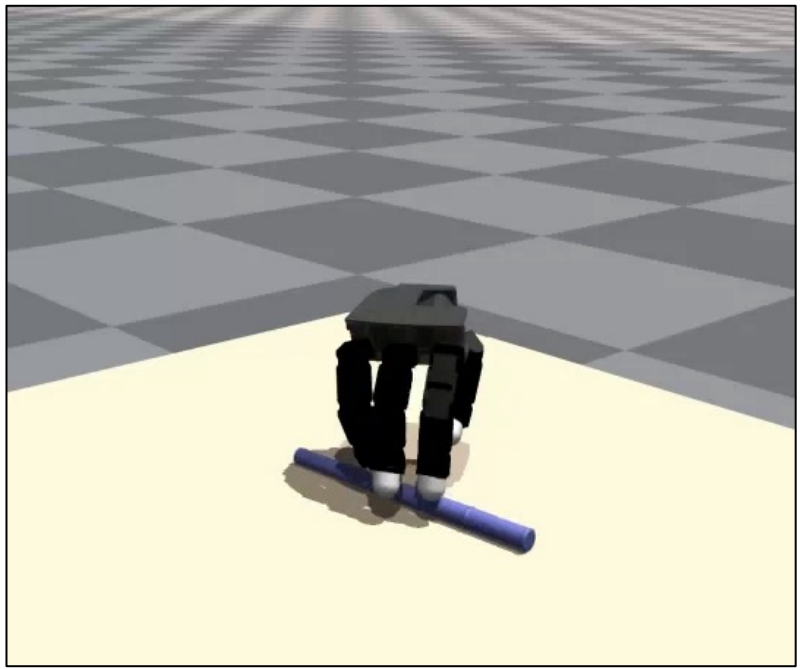


Baseline – PPO
(tracking rew., w/o sup.)

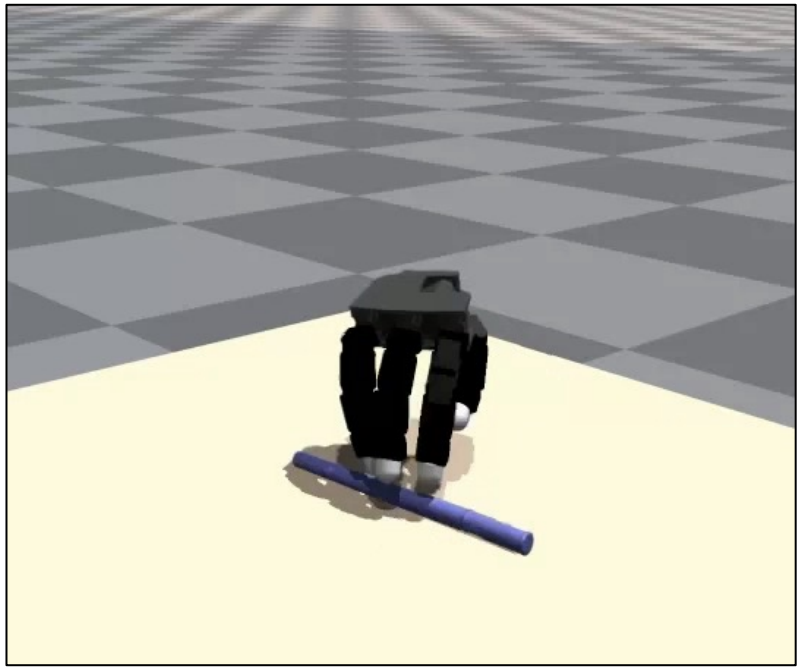
Effectiveness of the *Homotopy Optimization Scheme*



Retargeted
Kinematic Reference

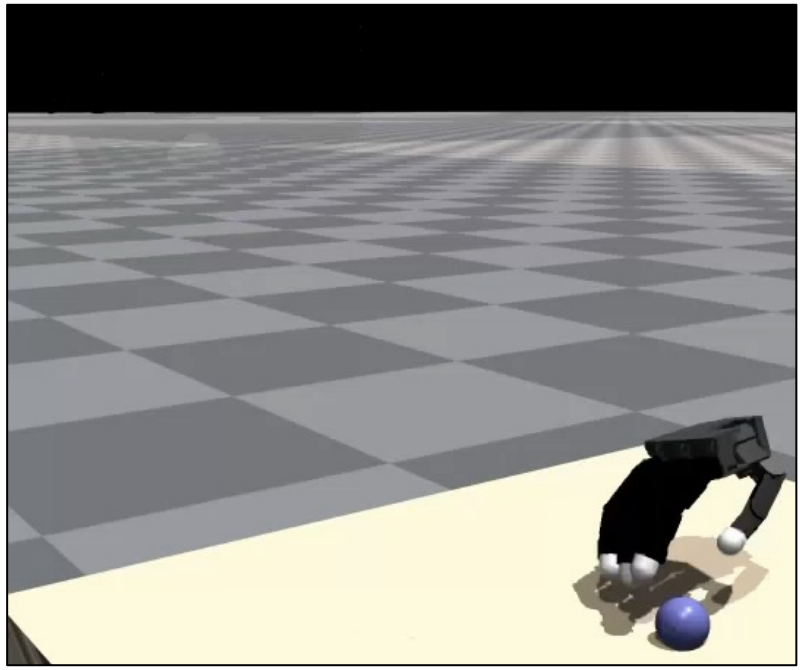


w/
structure
prior

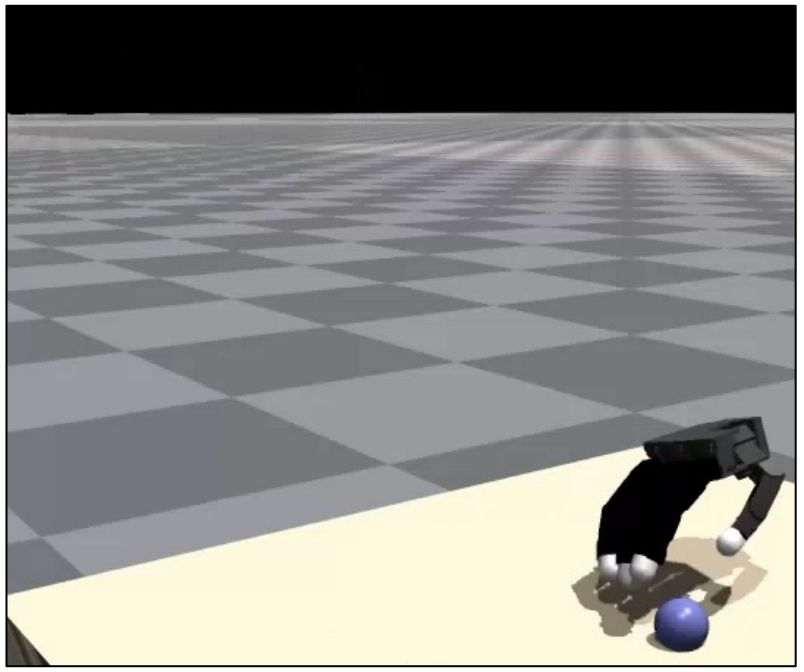


w/o
structure
prior

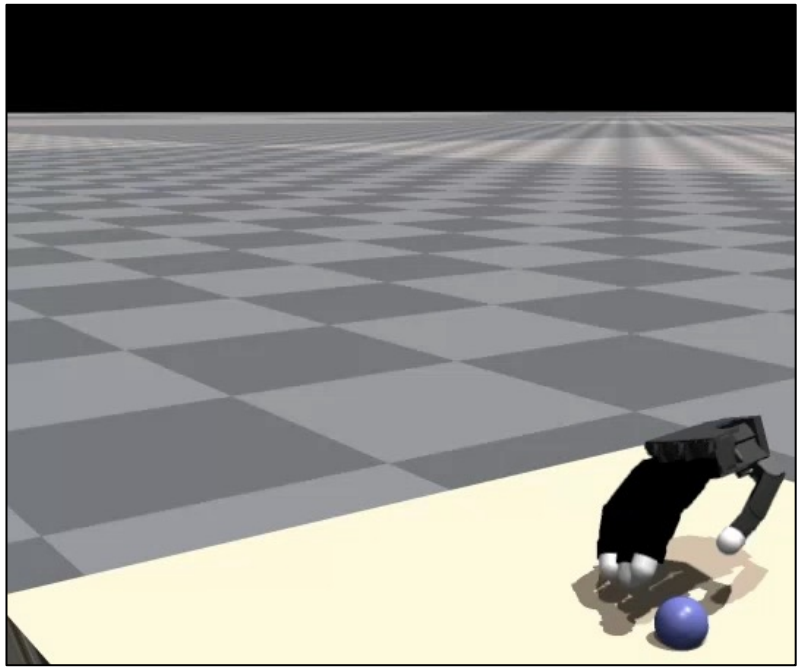
Effectiveness of the *Homotopy Optimization Scheme*



Retargeted
Kinematic Reference

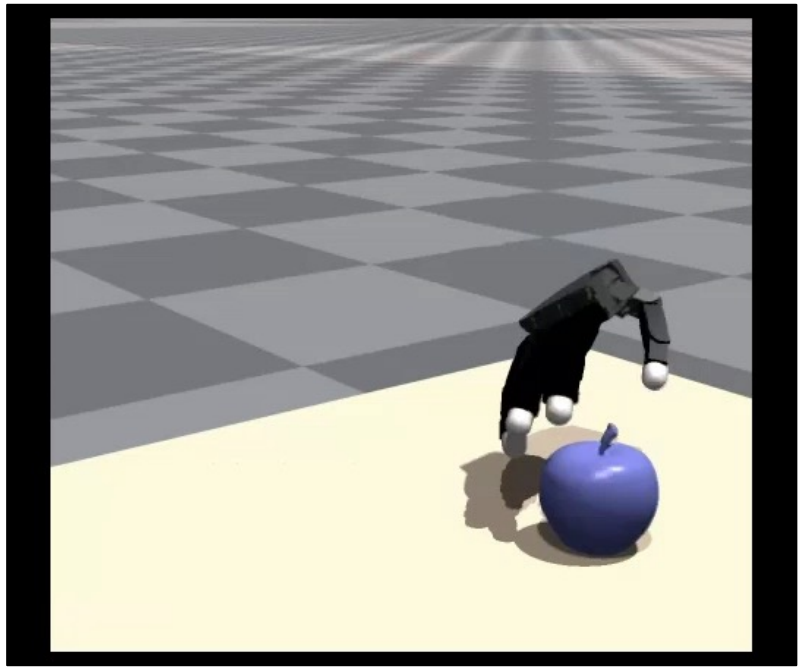


w/
structure
prior

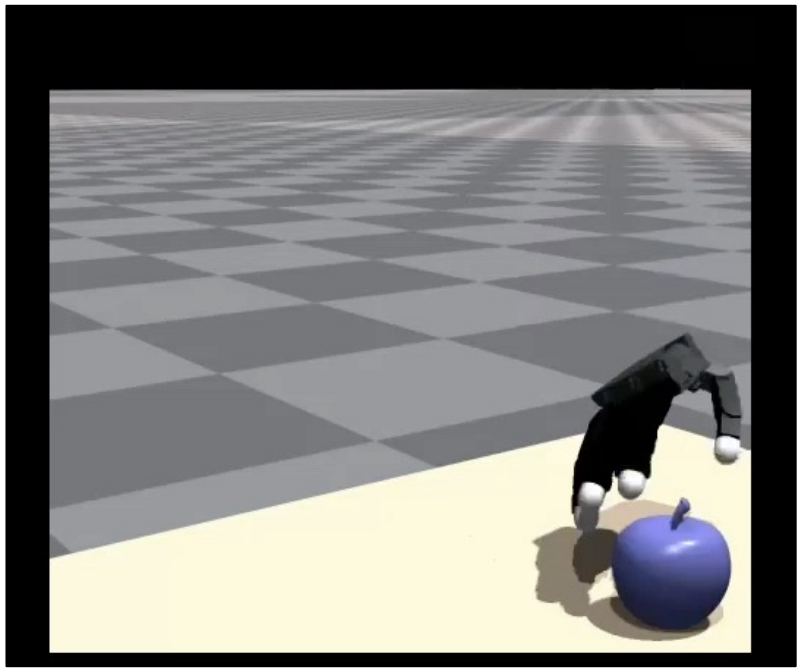


w/o
structure
prior

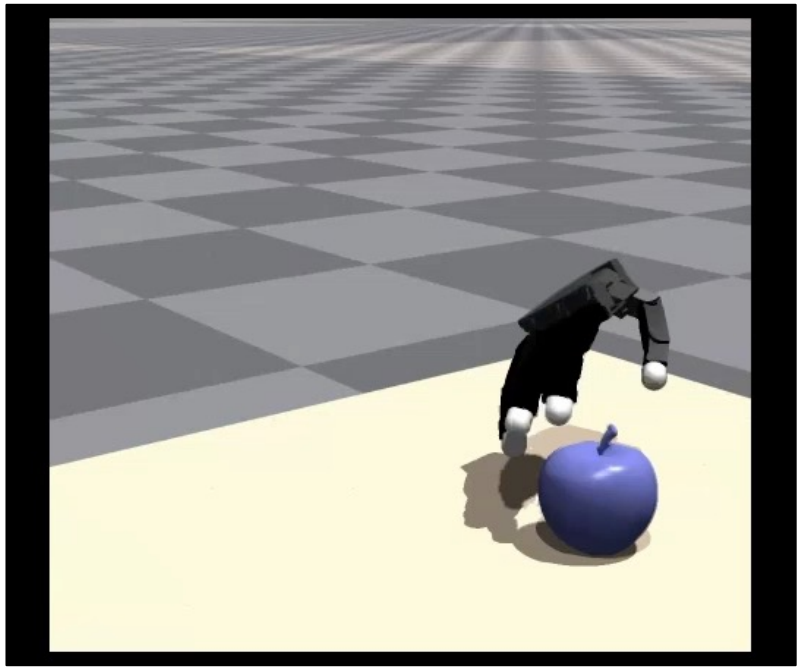
Effectiveness of the *Homotopy Optimization Scheme*



Retargeted
Kinematic Reference

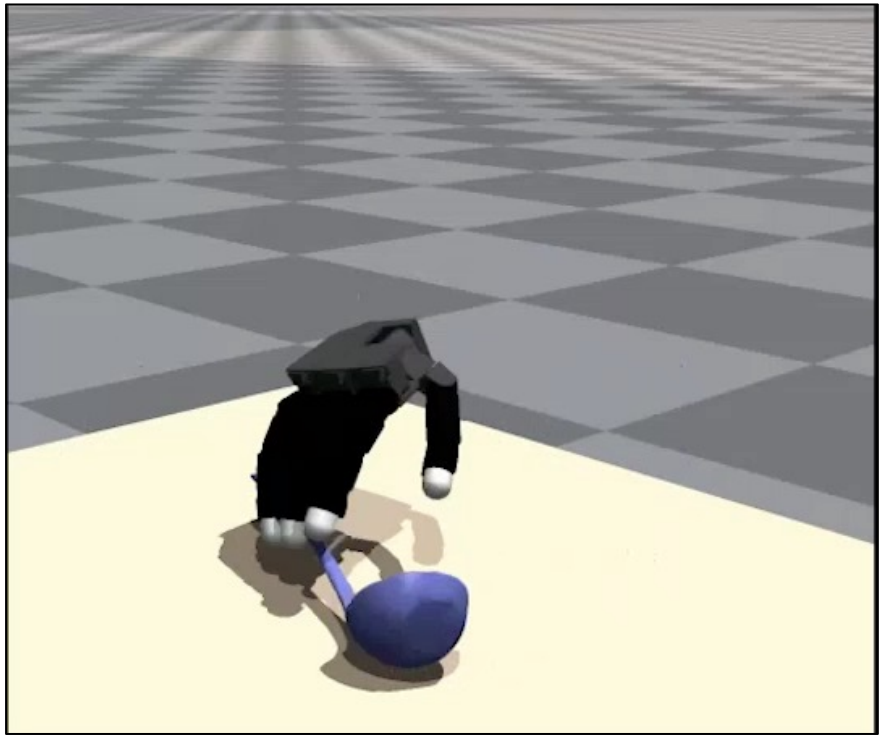


w/
structure
prior

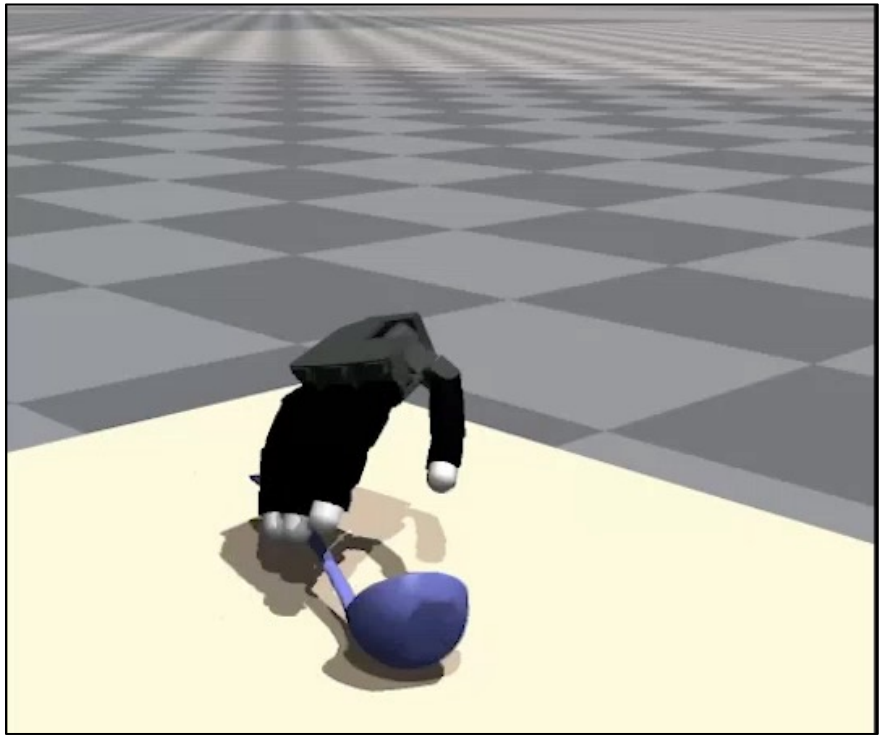


w/o
structure
prior

Robustness towards *Unreasonable* *Kinematic States*

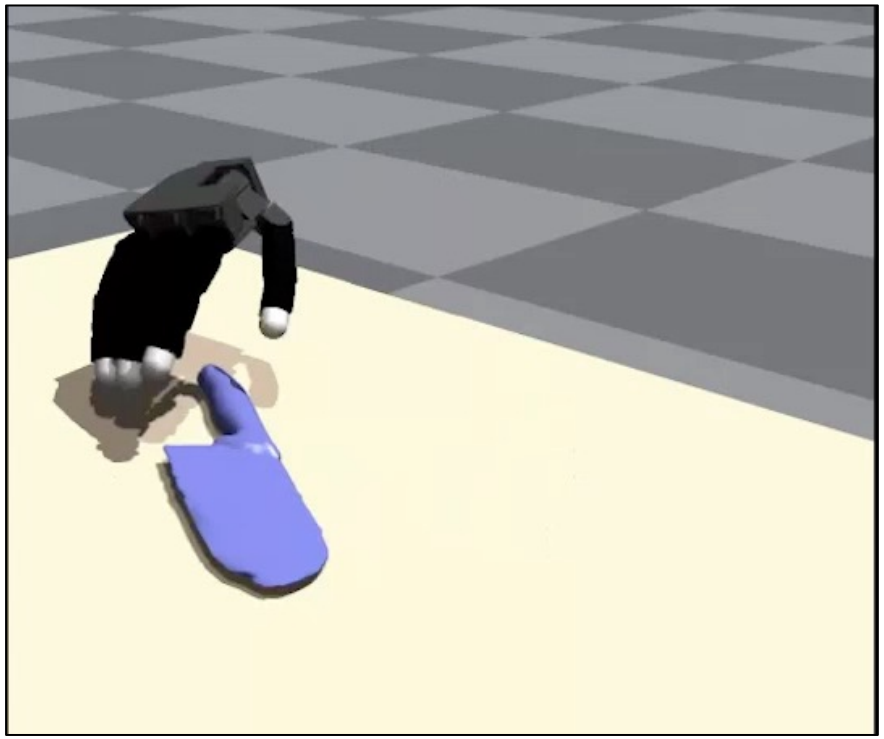


Retargeted Kinematic Reference

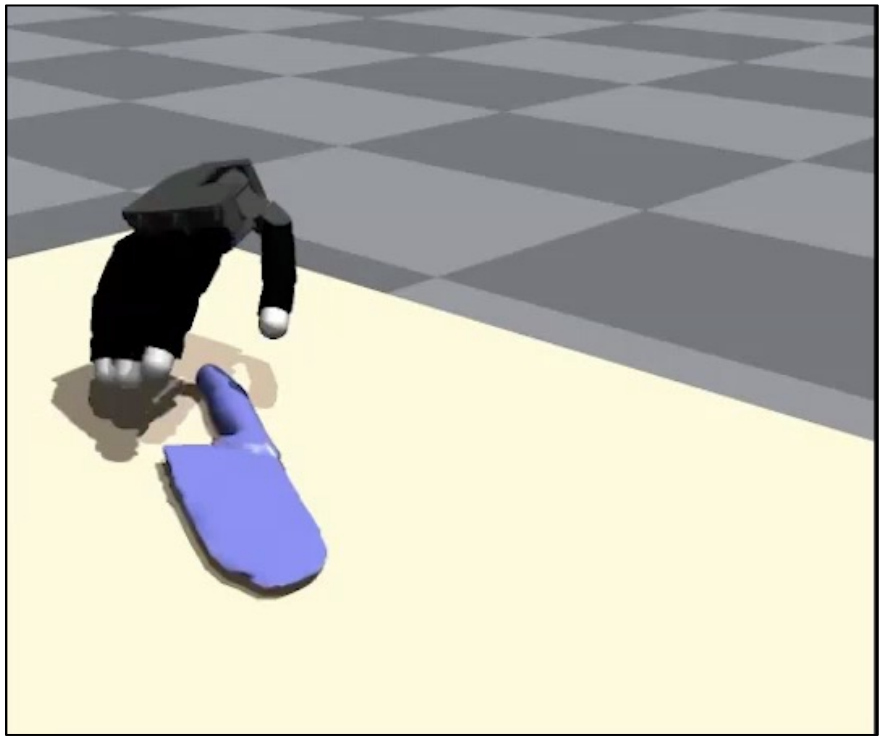


Tracking Result

Robustness towards *Out-of-Distribution* *Novel Interactions* with a *Brand New Object*



Retargeted Kinematic Reference



Tracking Result

DexTrack

Towards Generalizable Neural Tracking Control for *Dexterous Manipulation from Human References*

https://projectwebsite7.github.io/gene-dex-manip/

Thanks for listening!