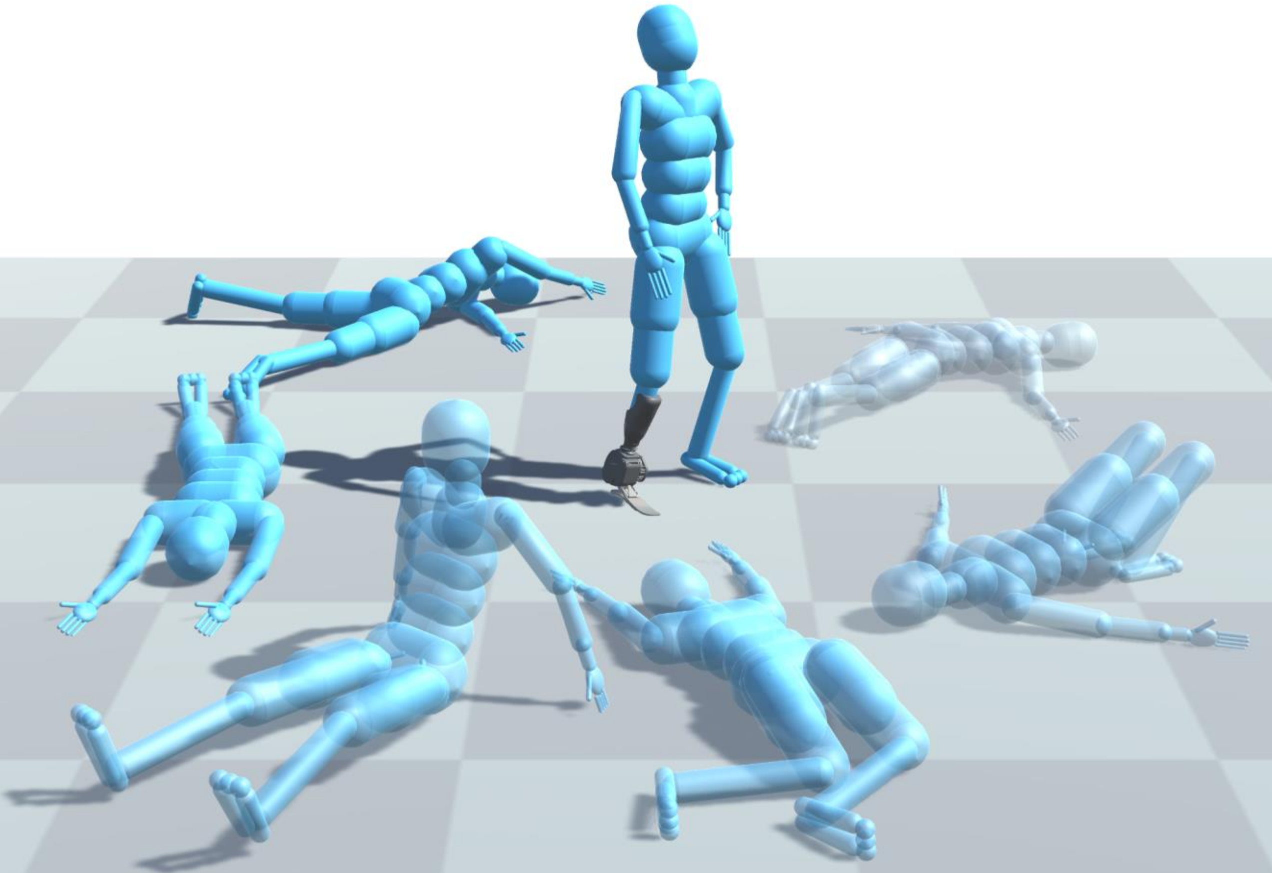
Imperial College London



Reinforcement Learning for Humanoid motion and Virtual Wearable Robotics

Balint Hodossy

Why use RL for control?

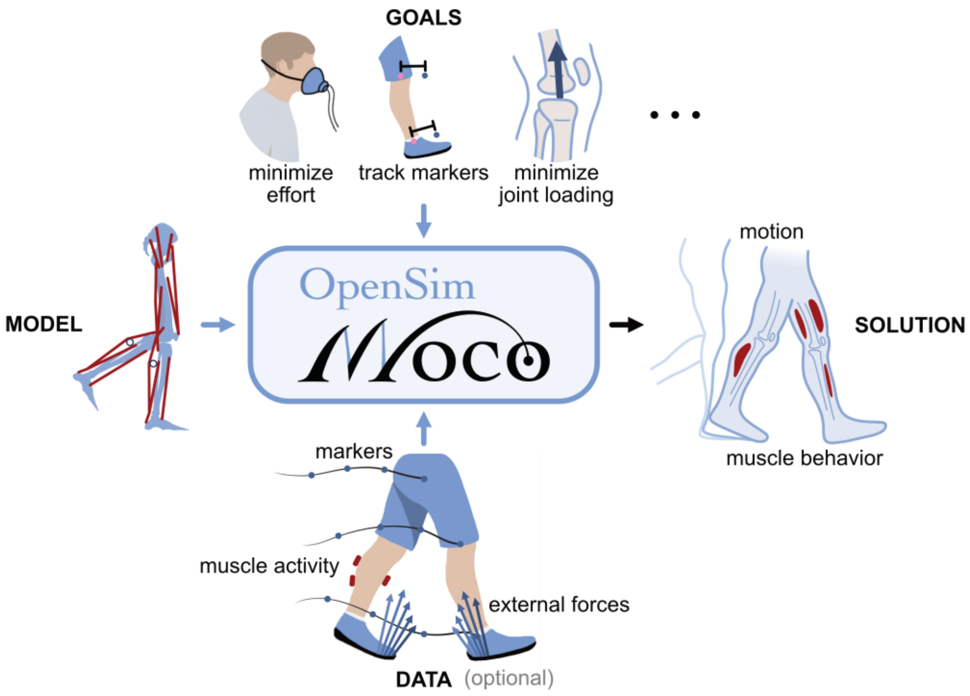
- What are the alternatives?

Trajectory optimization:

Iteratively adjust parameters to optimize an objective (motion goal, minimize effort, etc)

Subject to constraints (ground contact, joint limits).

Goal:
Identify single optimal movement path.



Why use RL for control?

- What are the alternatives?

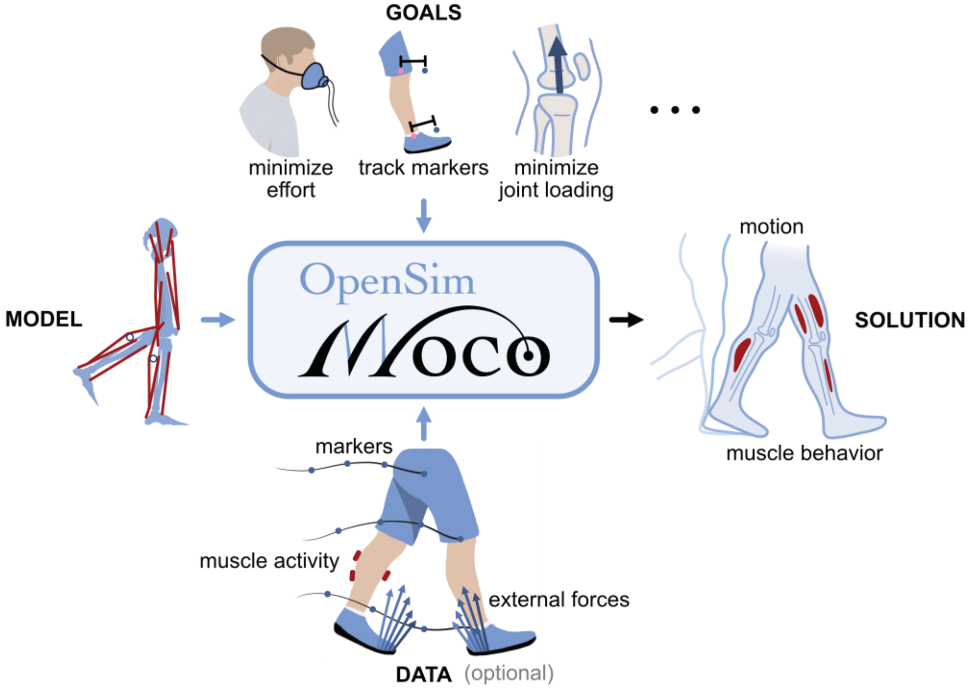
Trajectory optimization:

Iteratively adjust parameters to optimize an objective (motion goal, minimize effort, etc)

Subject to constraints (ground contact, joint limits).

Goal:
Identify single optimal movement path.

Slow to run for diverse movements, doesn't represent learning dynamics

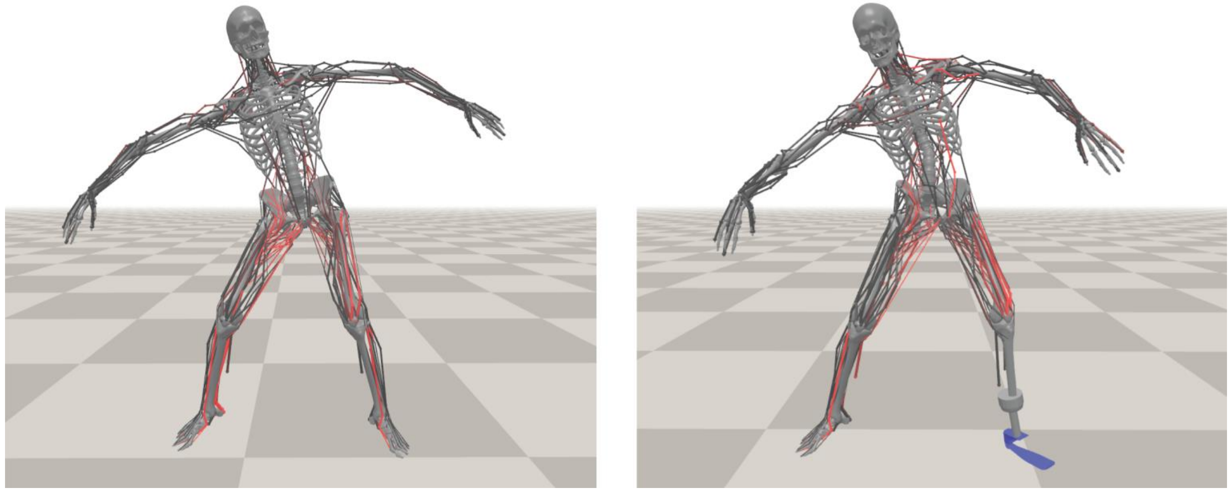


Plan for summary:

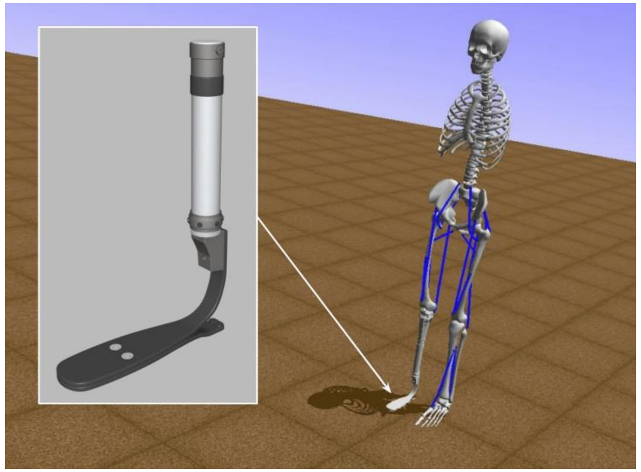
1. Example of how reinforcement learning can work
2. Summary of a few state-of-the-art algorithms for RL based motion synthesis
3. Applications to Neuromechanical MSk models
4. Simulation to real life transfer concerns

Simulated lower limb P&O Device

- Challenging, as you need to also simulate human movement that can react to the device



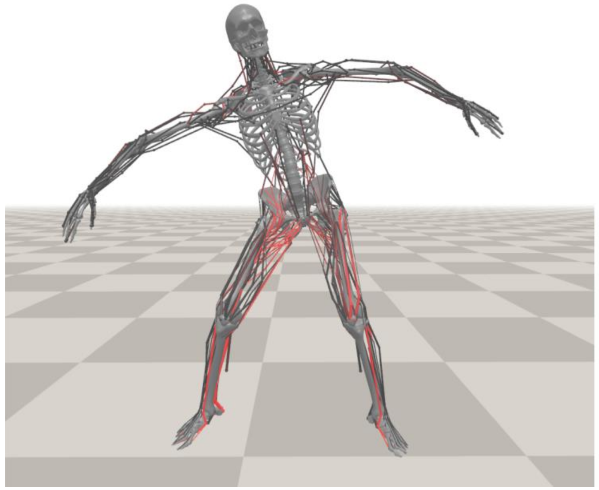
Scalable Muscle-Actuated Human Simulation and Control
Lee et al., 2019



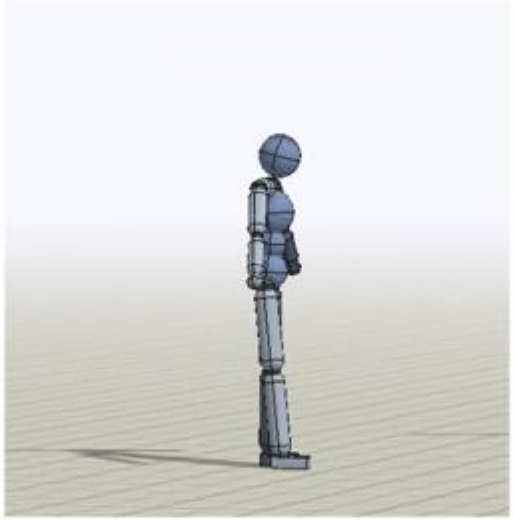
AI for prosthetics
Kidzinski et al., 2019

Simulating humanoid movement

- Decisions in structure, physics, and control



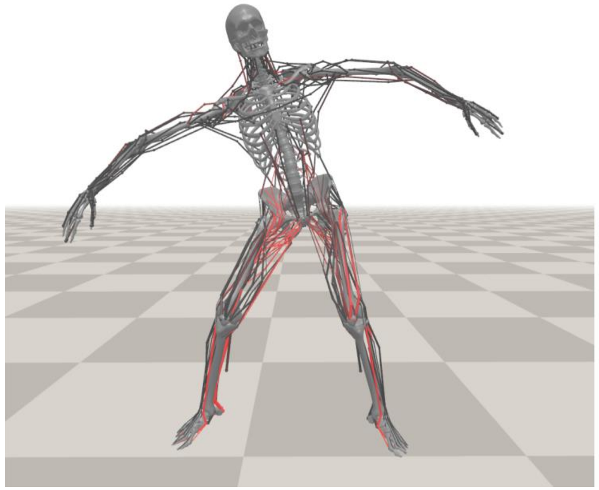
Scalable Muscle-Actuated Human Simulation and Control
Lee et al., 2019



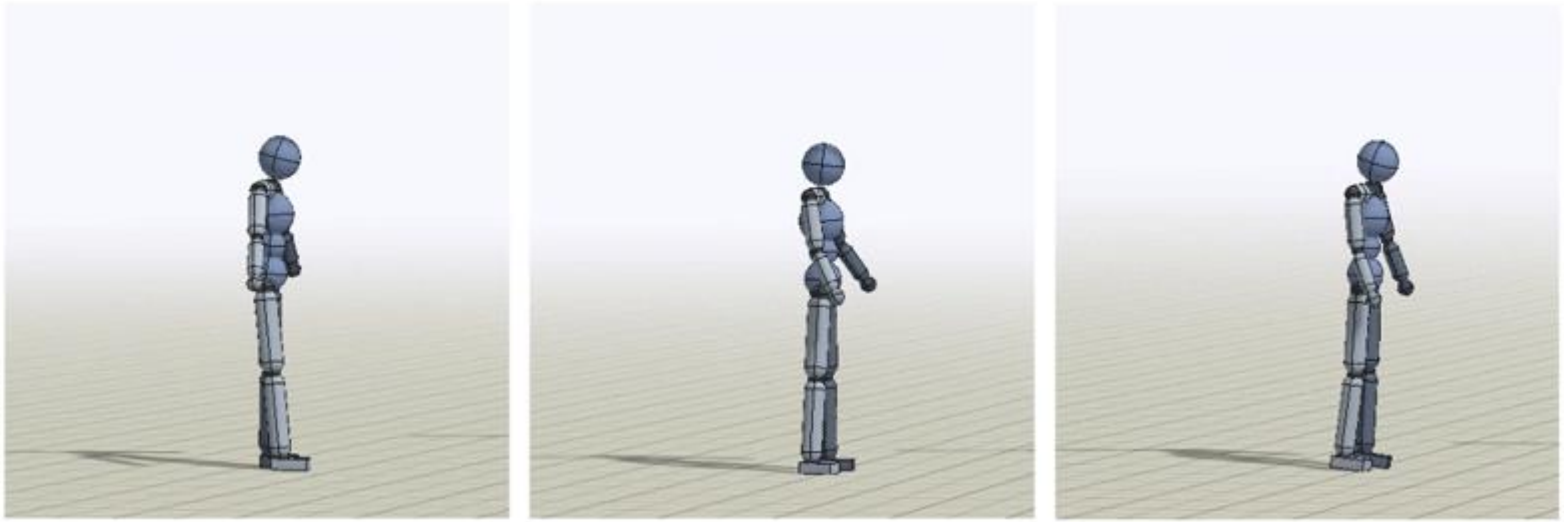
Deepmimic: Example-guided deep reinforcement learning of physics-based character skills
Peng et al., 2018

Simulating humanoid movement

- Decisions in structure, physics, and control

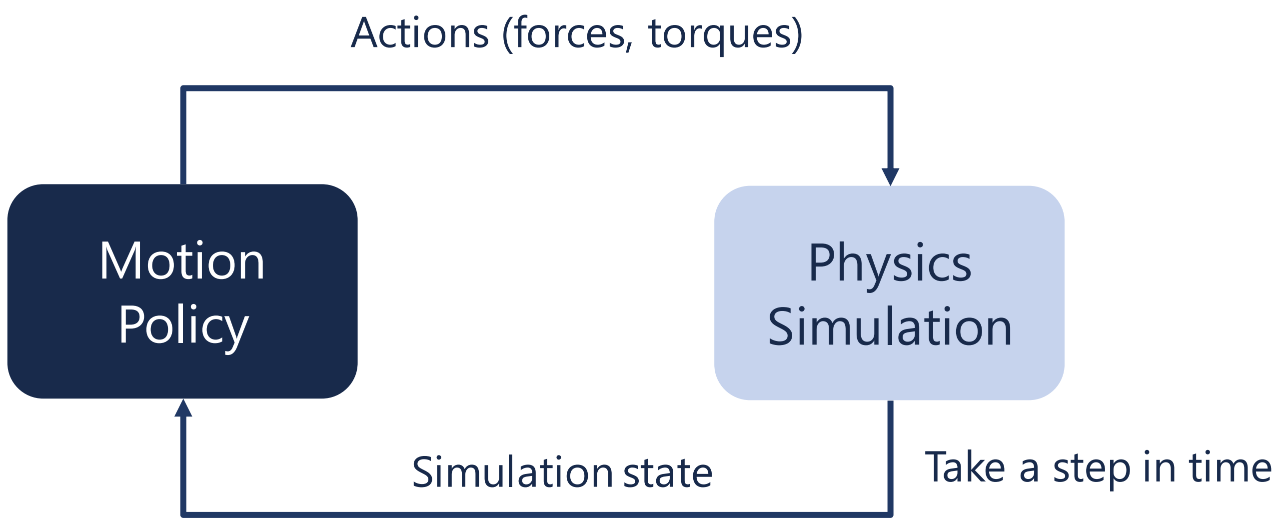


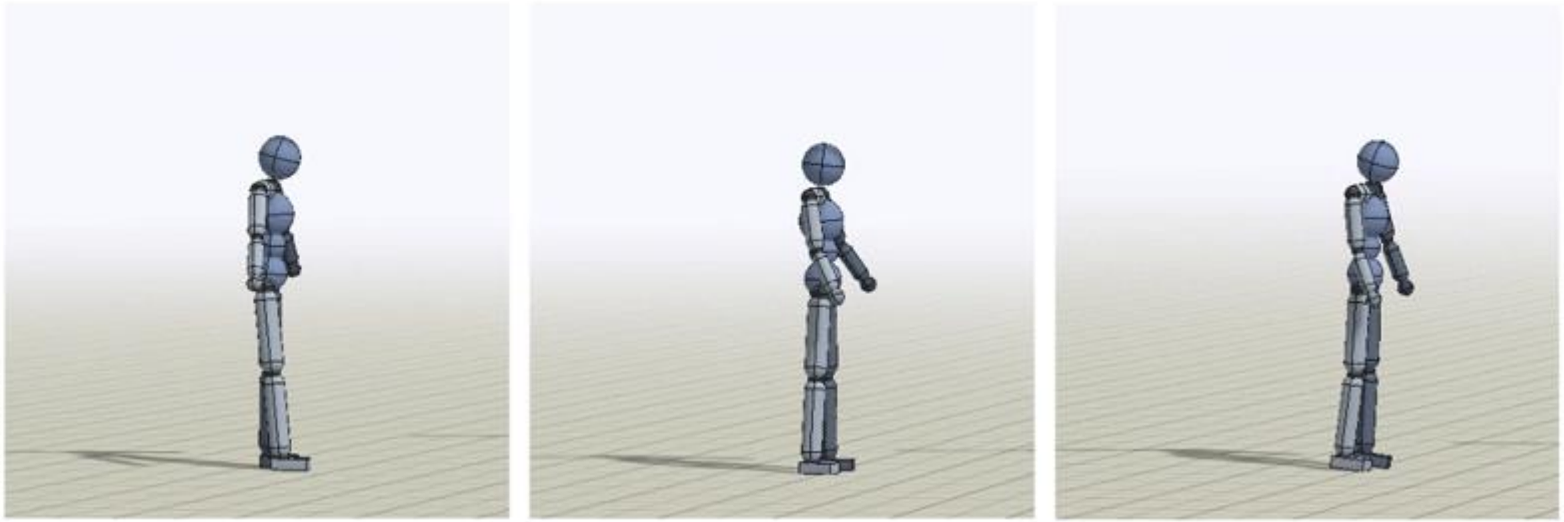
Scalable Muscle-Actuated Human Simulation and Control
Lee et al., 2019



Simulating humanoid movement

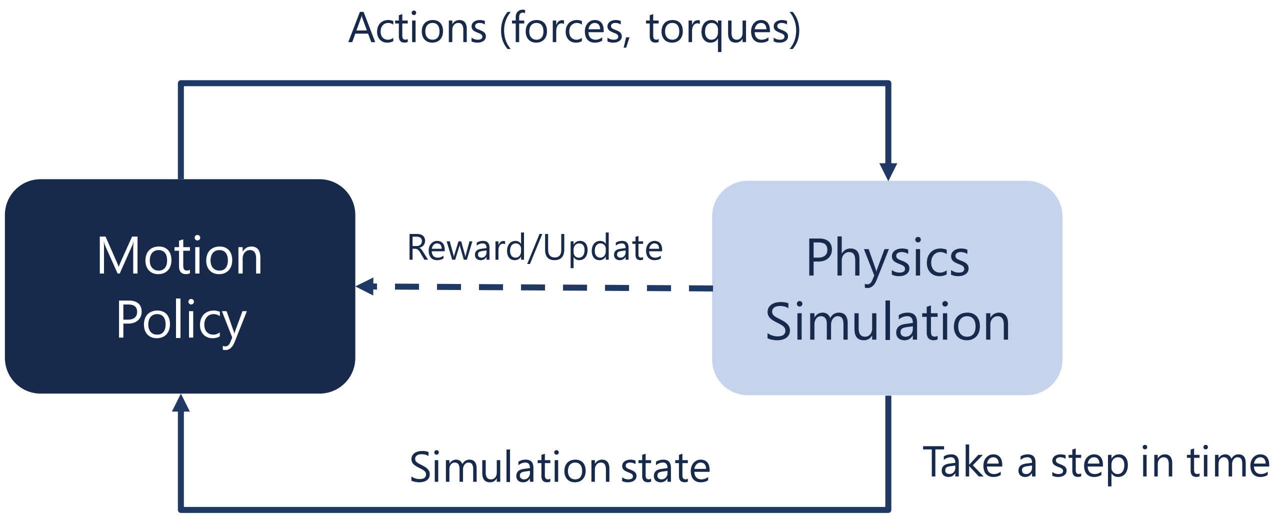
- Decisions in structure, physics, and control

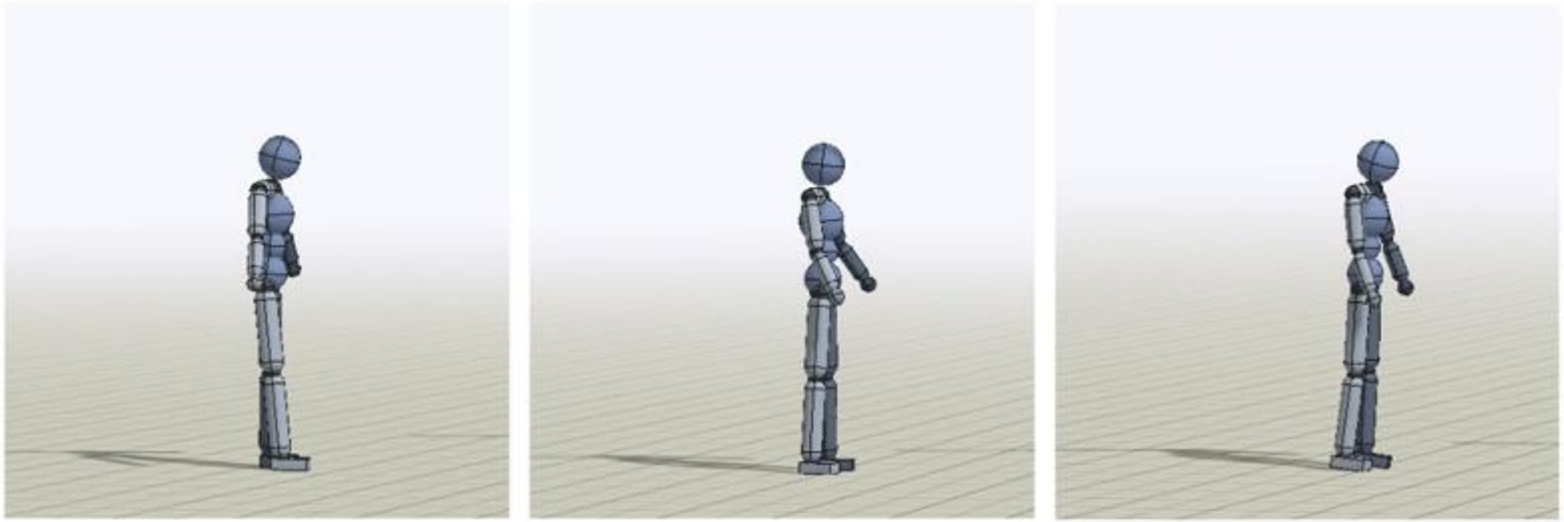




Simulating humanoid movement

- Decisions in structure, physics, and control

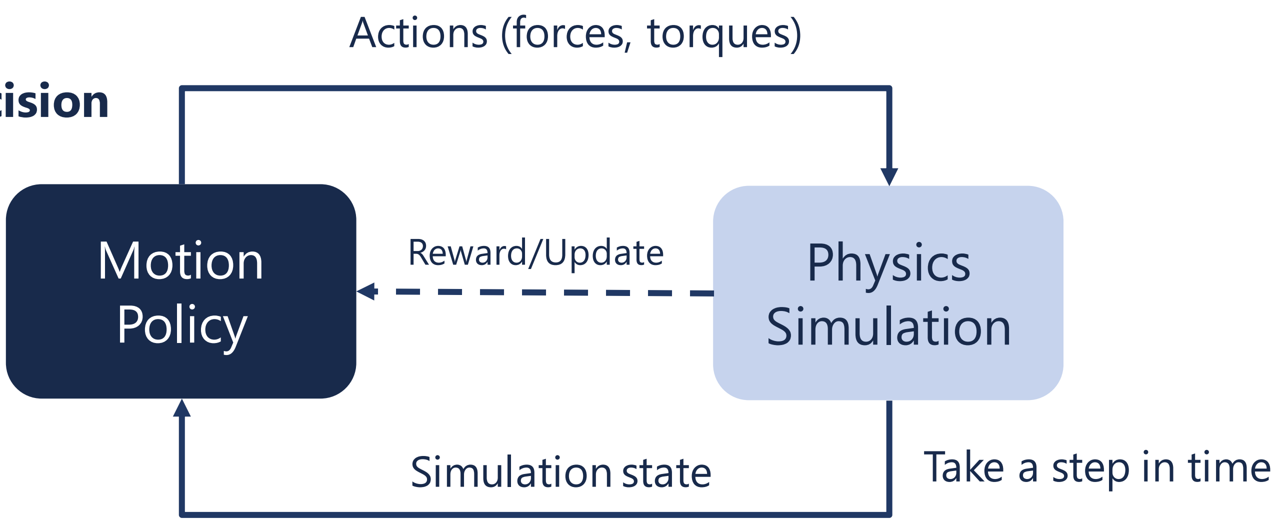


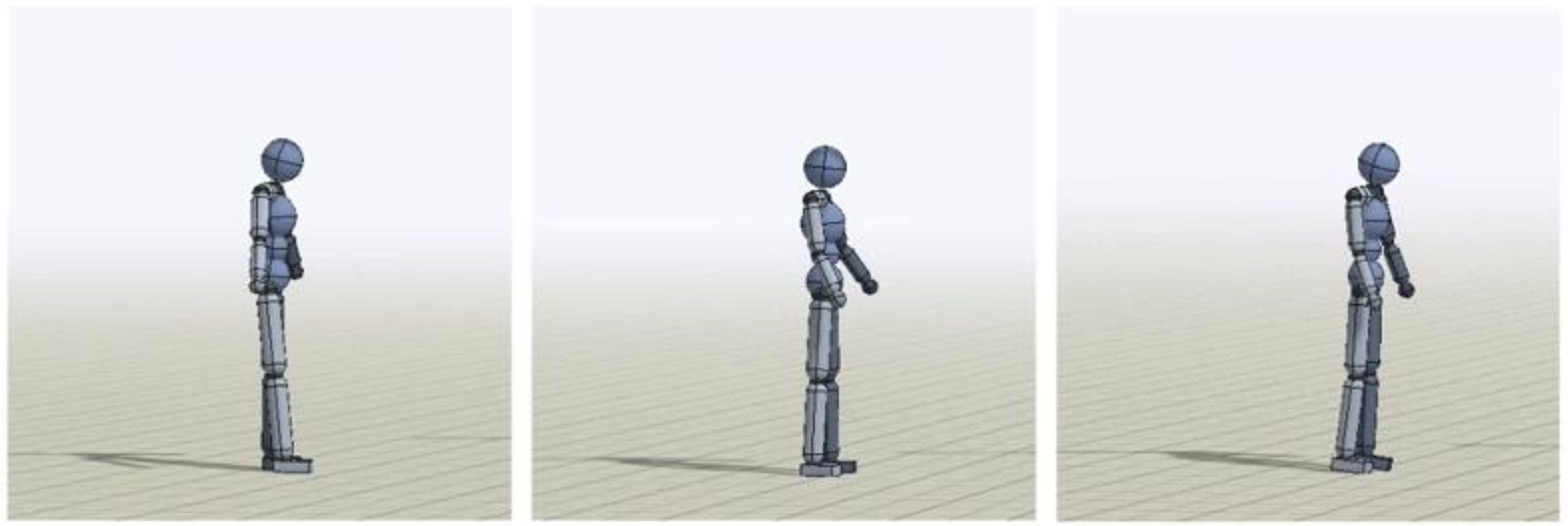


Simulating humanoid movement

- Decisions in structure, physics, and control

Markov Decision Process

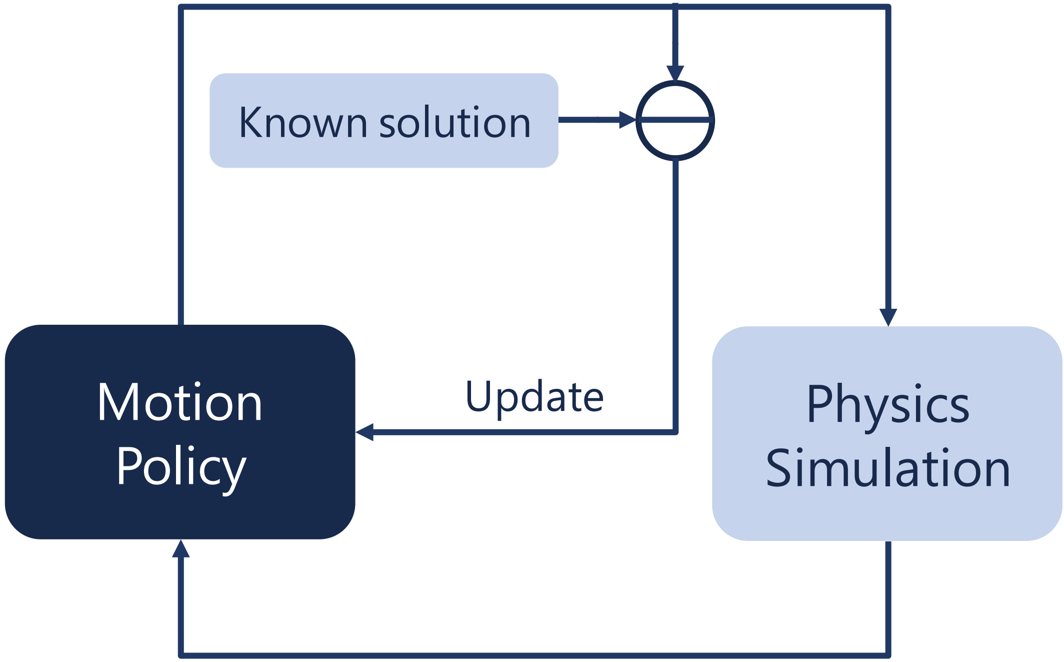




Learning a motion policy

Different approaches to teaching controller

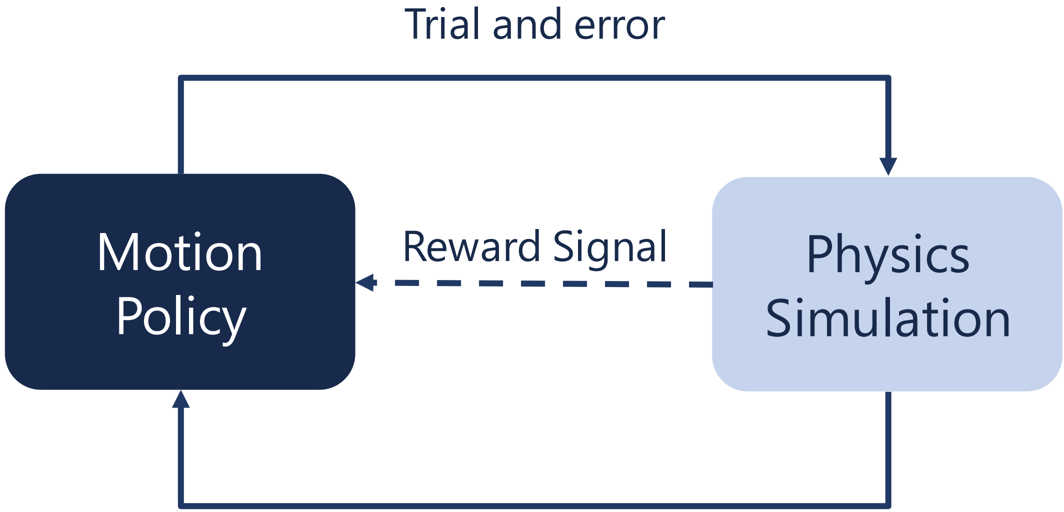
Supervised learning:



Learning a motion policy

Different approaches to teaching controller

Reinforcement learning:



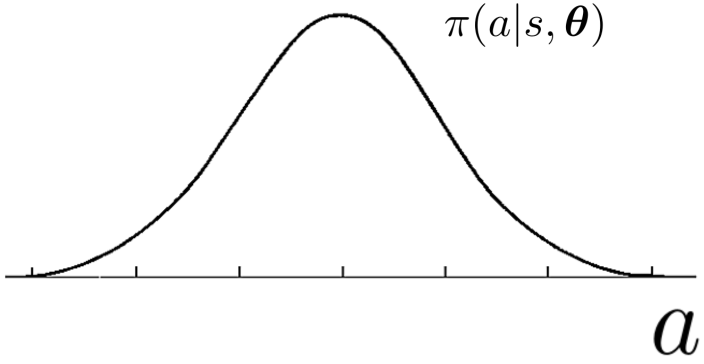
One approach to learning a policy

- Of the Proximal Policy Optimization flavour

Stochastic policy

$$\pi(a|s, \boldsymbol{\theta})$$

For a given $s, \boldsymbol{\theta}$



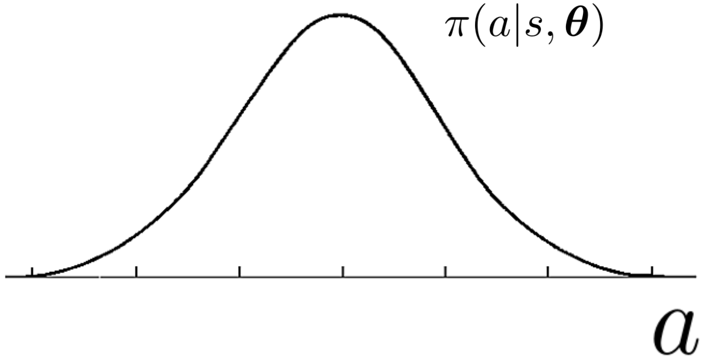
One approach to learning a policy

- Of the Proximal Policy Optimization flavour

Stochastic policy

$$\pi(a|s, \boldsymbol{\theta})$$

For a given $s, \boldsymbol{\theta}$



Sample random action A_t from the random variable a

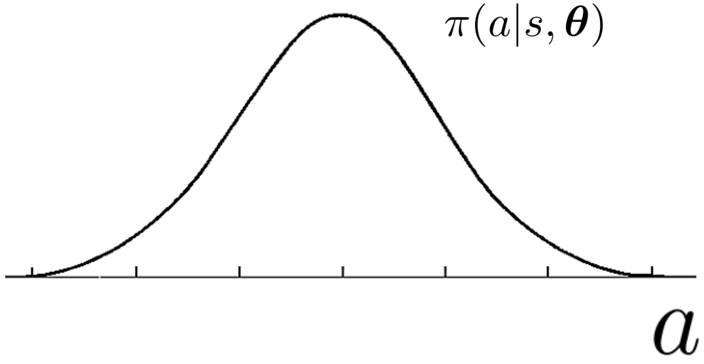
One approach to learning a policy

- Of the Proximal Policy Optimization flavour

Stochastic policy

$$\pi(a|s, \boldsymbol{\theta})$$

For a given $s, \boldsymbol{\theta}$



Perform episode following policy, collect rewards, summed in G_t .
Have some expected baseline performance $b(S_t)$.

Sample random action A_t from the random variable a

One approach to learning a policy

- Of the Proximal Policy Optimization flavour

Direction to change parameters θ to make this action more likely

Scaling factor

$$\boldsymbol{\theta}_{t+1} \doteq \boldsymbol{\theta}_t + \alpha\Big(G_t - b(S_t)\Big)\frac{\nabla\pi(A_t|S_t,\boldsymbol{\theta}_t)}{\pi(A_t|S_t,\boldsymbol{\theta}_t)}$$

Original parameters

How much better we performed than we expected

How likely this action was

Perform episode following policy, collect rewards, summed in G_t .
Have some expected baseline performance $b(S_t)$.

Sample random action A_t from the random variable a

One approach to learning a policy

- Of the Proximal Policy Optimization flavour

Direction to change parameters θ to make this action more likely

Scaling factor

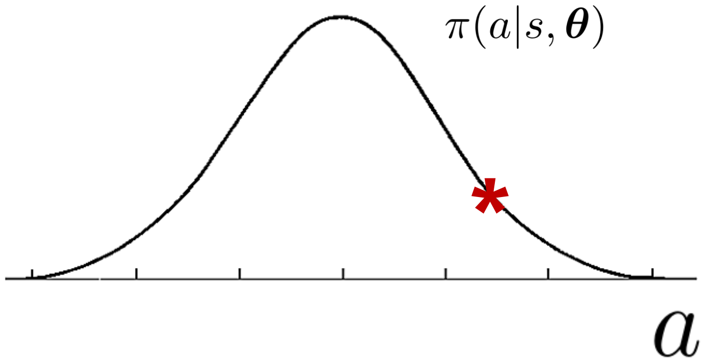
$$\boldsymbol{\theta}_{t+1} \doteq \boldsymbol{\theta}_t + \alpha\Big(G_t - b(S_t)\Big)\frac{\nabla\pi(A_t|S_t,\boldsymbol{\theta}_t)}{\pi(A_t|S_t,\boldsymbol{\theta}_t)}$$

Original parameters

How much better we performed than we expected (advantage)

How likely this action was

Example:



Took unlikely action, but more reward than expected, and reached a state we think is better!

One approach to learning a policy

- Of the Proximal Policy Optimization flavour

Direction to change parameters θ to make this action more likely

Scaling factor

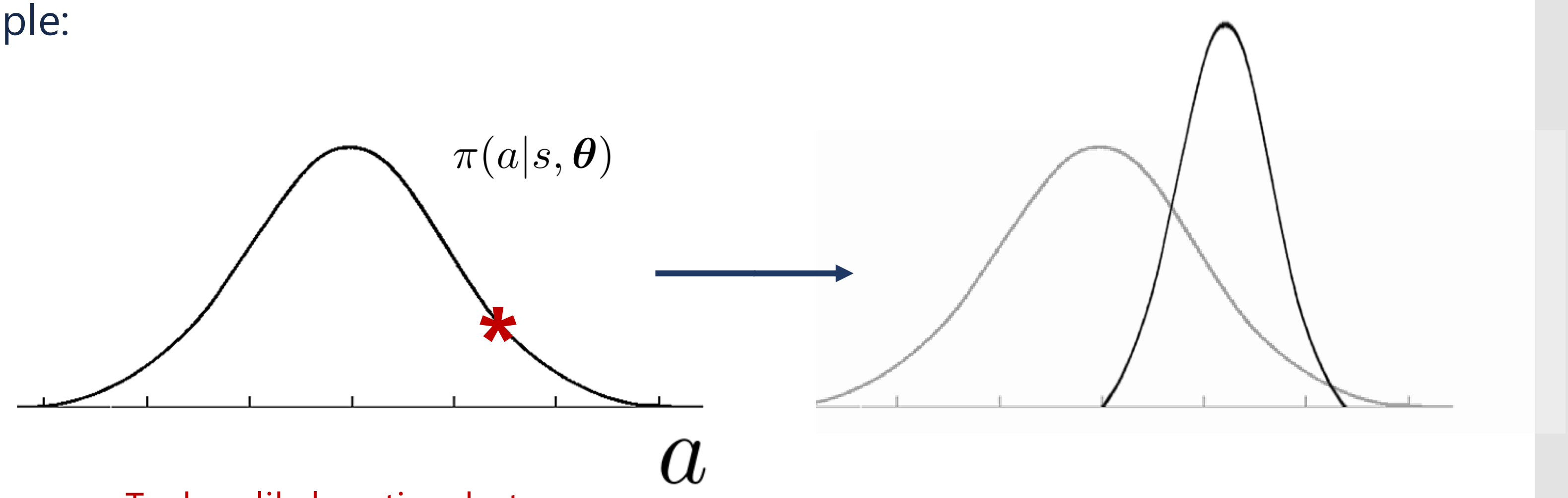
$$\boldsymbol{\theta}_{t+1} \doteq \boldsymbol{\theta}_t + \alpha\Big(G_t - b(S_t)\Big)\frac{\nabla\pi(A_t|S_t,\boldsymbol{\theta}_t)}{\pi(A_t|S_t,\boldsymbol{\theta}_t)}$$

Original parameters

How much better we performed than we expected (advantage)

How likely this action was

Example:



Took unlikely action, but more reward than expected, and reached a state we think is better!

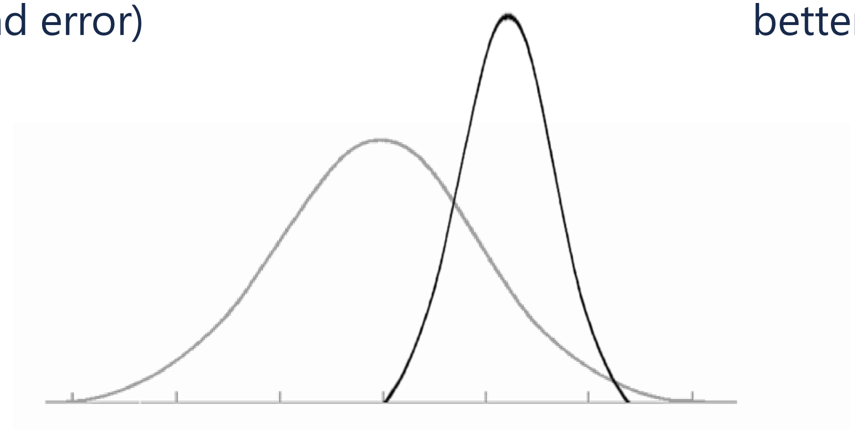
Adjust parameters to make this action more likely in the future!

One approach to learning a policy

- Of the Proximal Policy Optimization flavour

Wide distribution: Explore, take new actions (learn by trial and error)

Narrow distribution: Exploit, take actions currently thought better



One approach to learning a policy

- Of the Proximal Policy Optimization flavour

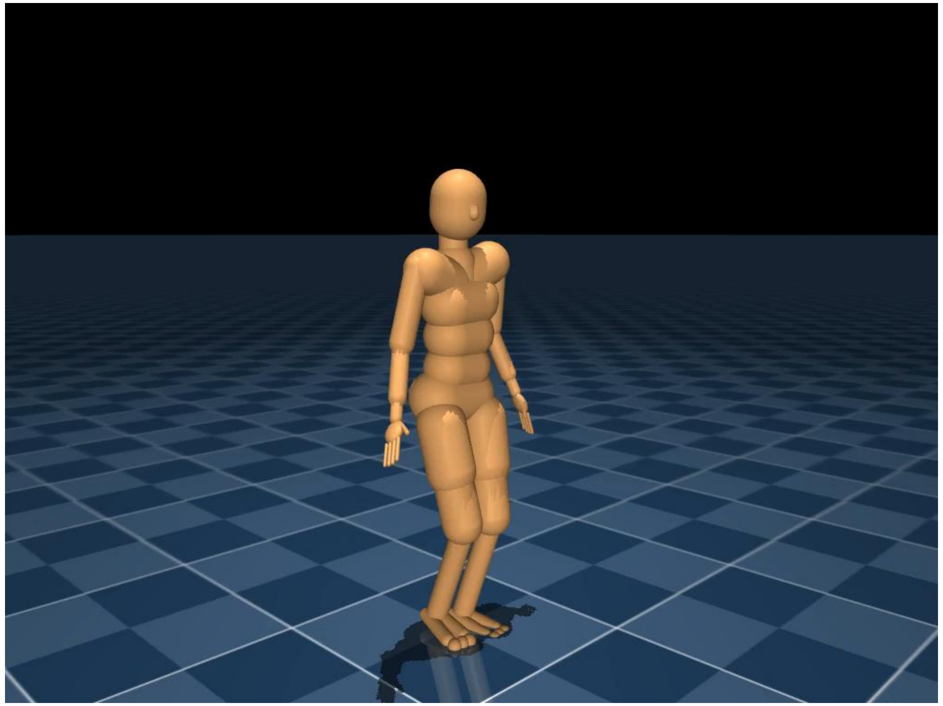
Mapping states to action distributions and states to value/advantage can be performed with function estimators.

If these function estimators are deep learning estimators (like with an ANN) then you are doing deep reinforcement learning.

Learning a motion policy

Different approaches to teaching controller

Reinforcement learning:

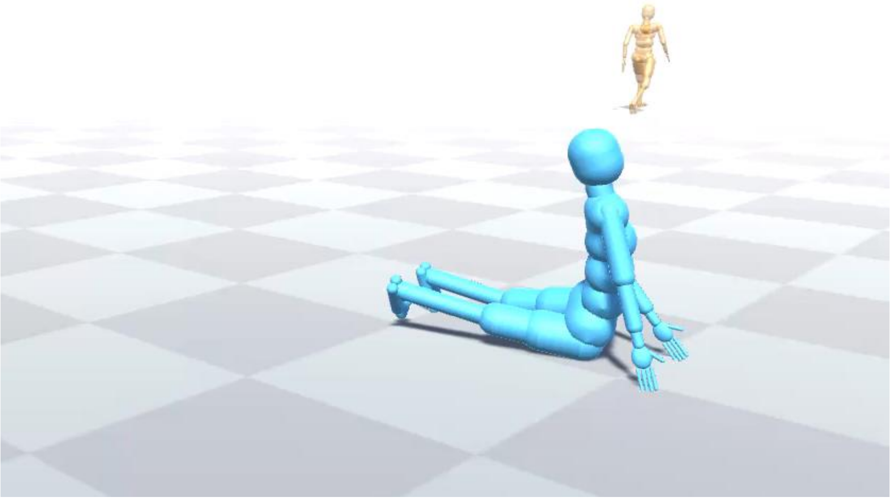


Wagener et al., 2022

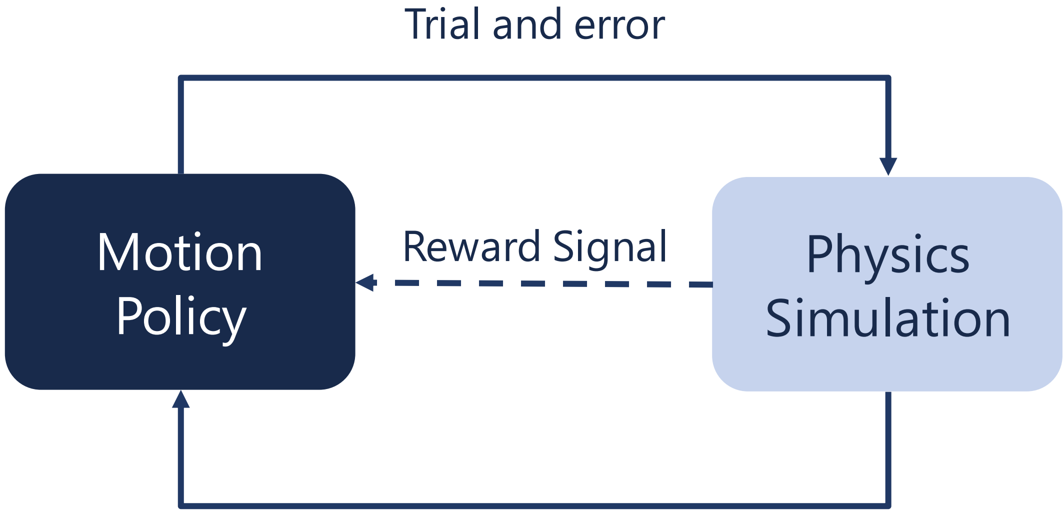
Learning a motion policy

Different approaches to teaching controller

Reinforcement learning:



"Imitate movement, but don't let your head touch the ground at all cost!"

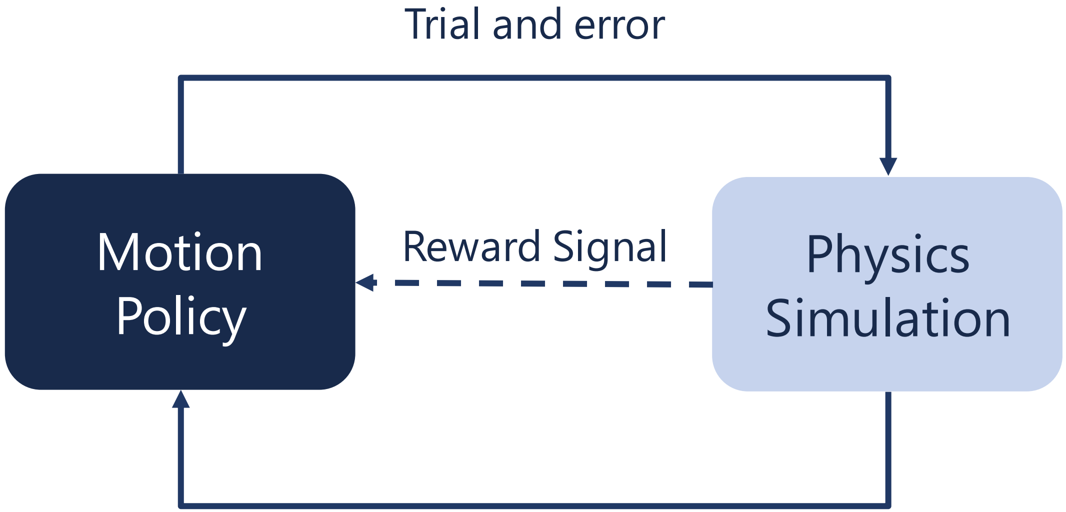


Learning a motion policy

Different approaches to teaching controller

Key decisions:

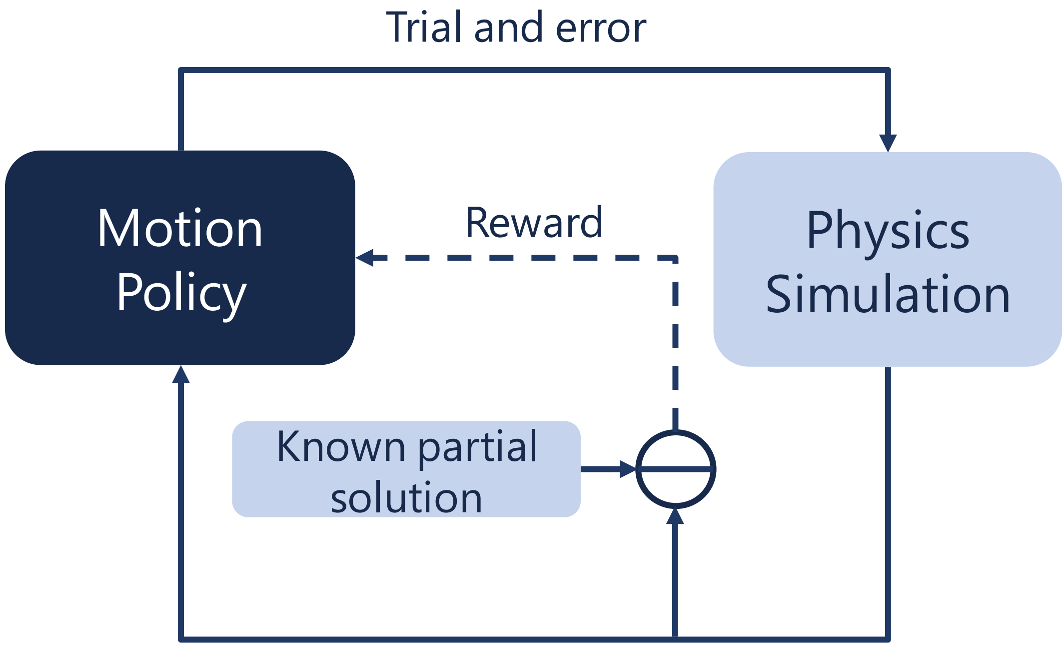
- What are the observations input to the policy?
- How are its actions interpreted?
- How can we shape the reward function to capture our intent?
- How are episodes terminated and started?
- How/when are actions sampled?



Learning a motion policy

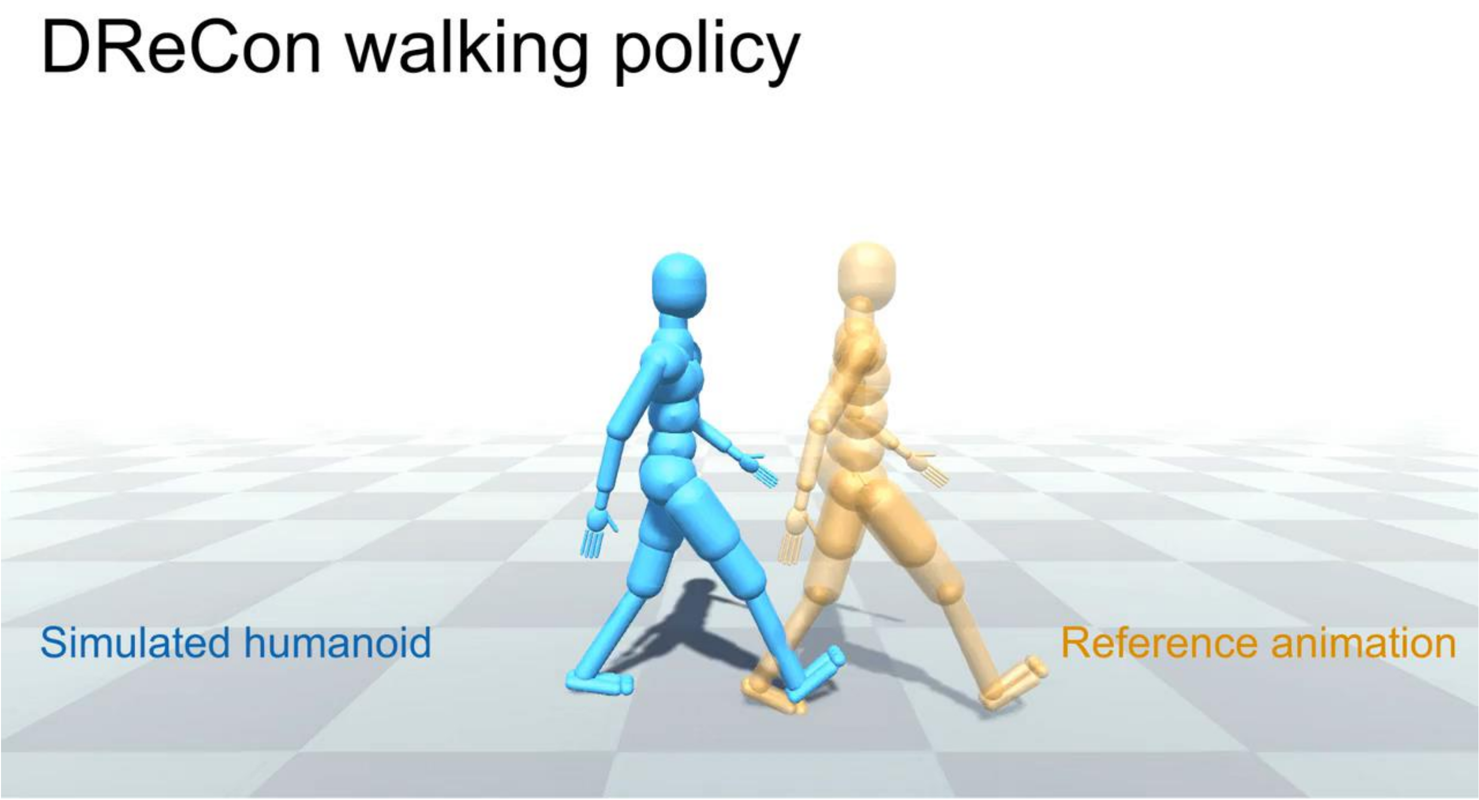
Different approaches to teaching controller

Reinforcement learning with **Motion Tracking**:



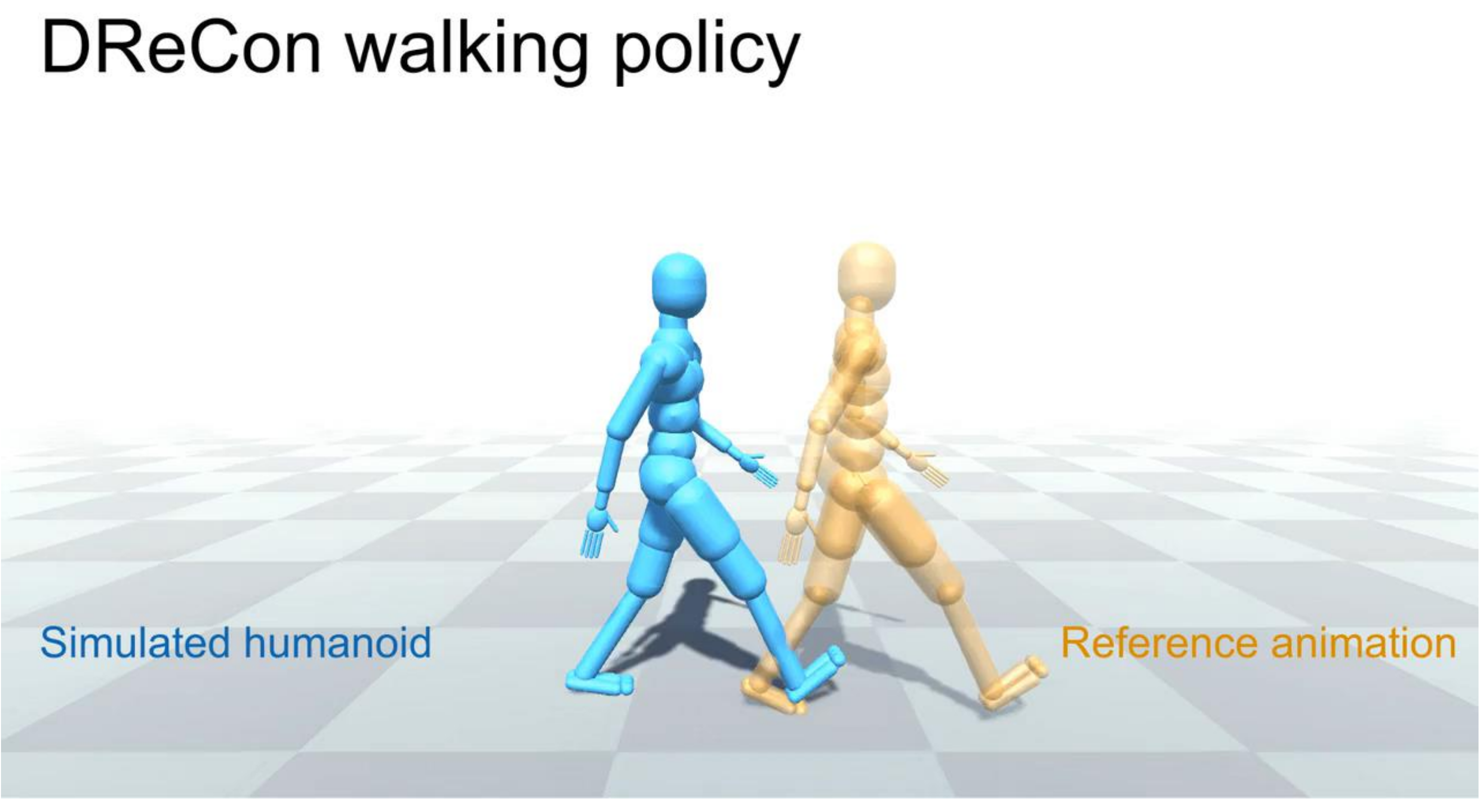
Motion tracking with RL

- Don’t need to know exact solution
- Still need synchronised reference motion



Motion tracking with RL

- Don’t need to know exact solution
- Still need synchronised reference motion
- May provide **proprioception** or **phase** information for the agent to know the state of the reference motion

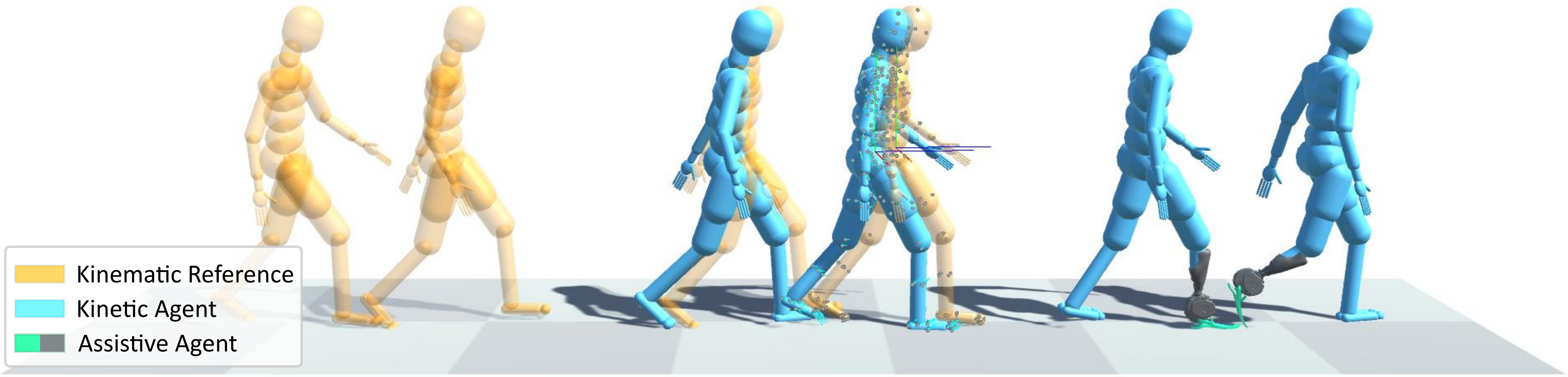


Virtual P&O controller testbed

Motion Capture
↪ Synthesized Reference Animation

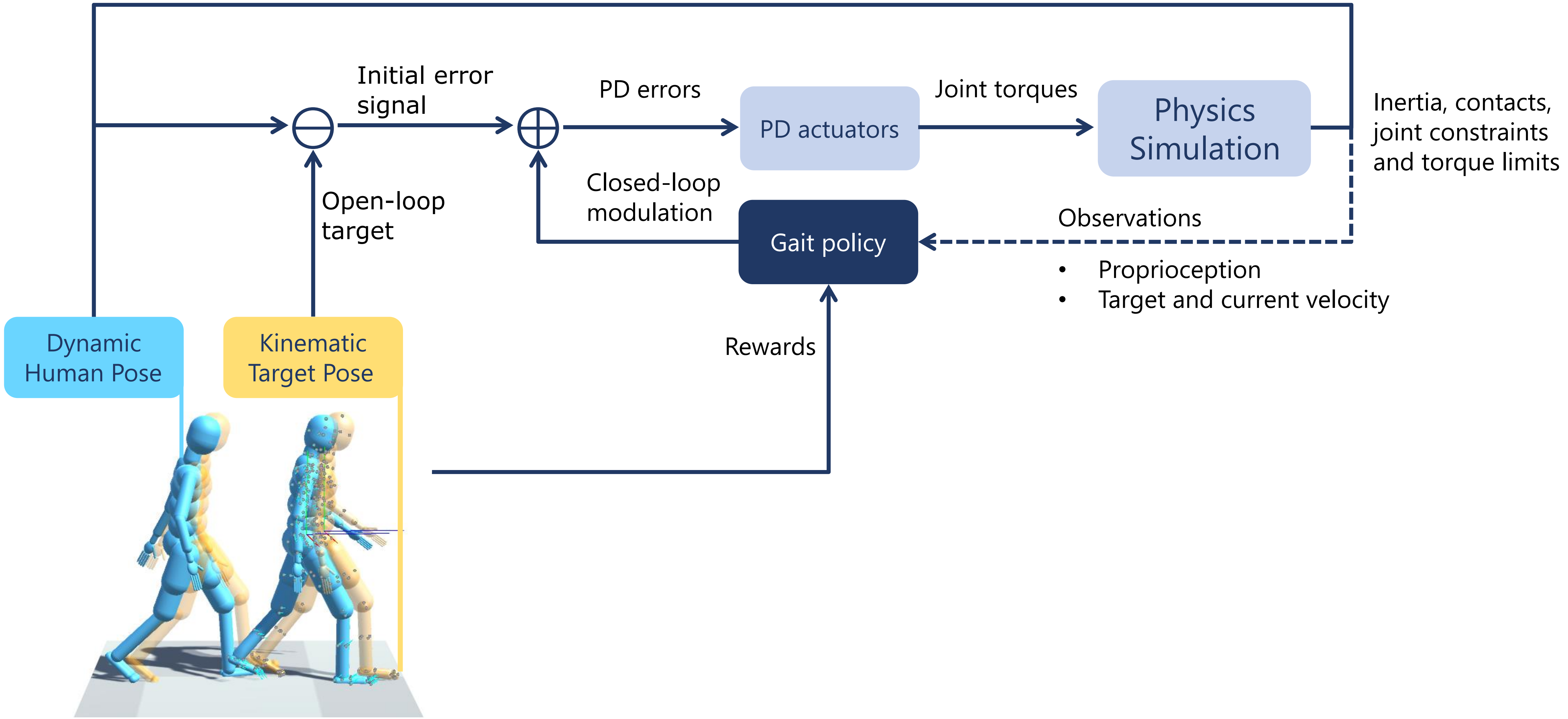
Reference ⊖ Simulation
↪ Reward Signal → Gait Control Policy

Gait perturbation and Low-DoF Assistance
↪Device Control Policy

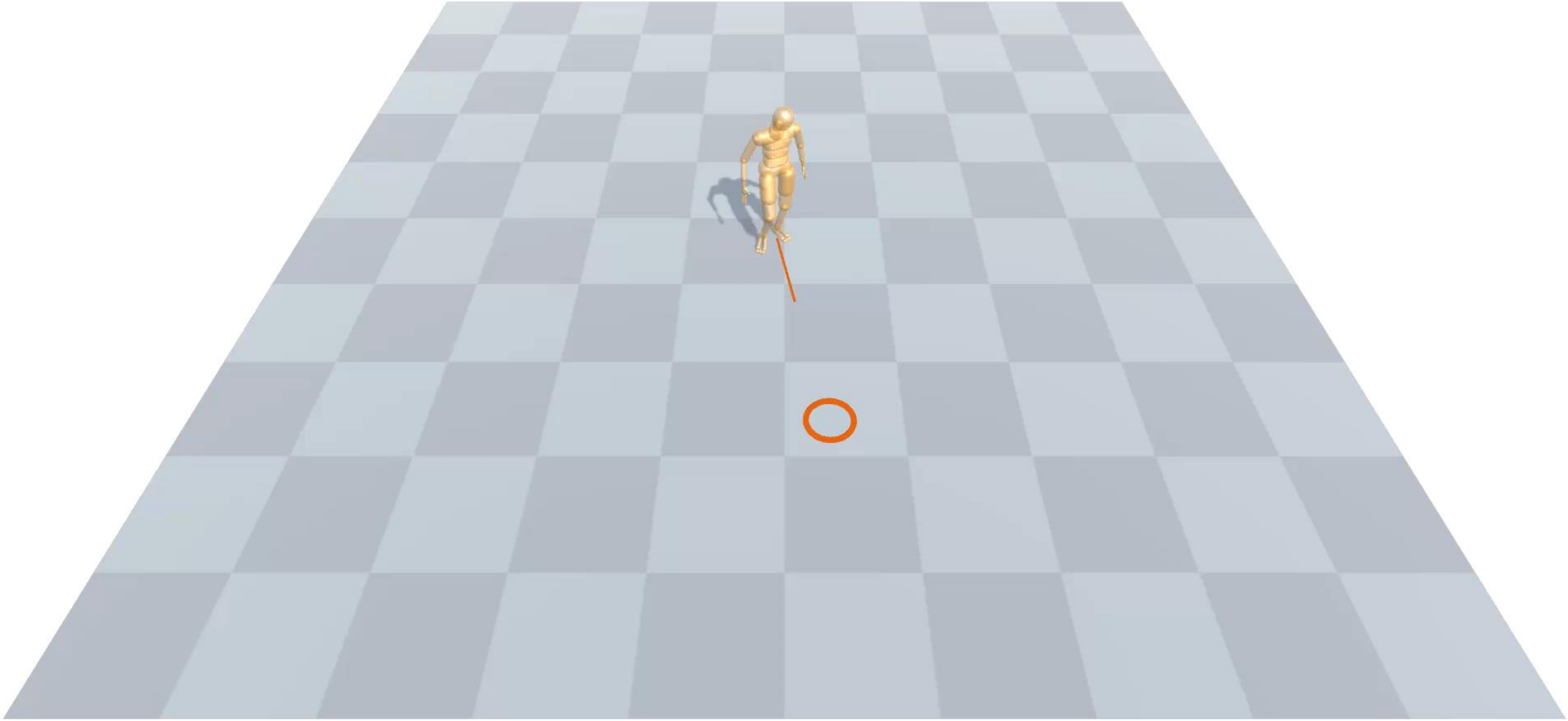


Hodossy, B.K. and Farina, D., 2023.
Shared Autonomy Locomotion Synthesis with a Virtual Powered Prosthetic Ankle.

Motion tracking with RL



RL environment – learning a policy



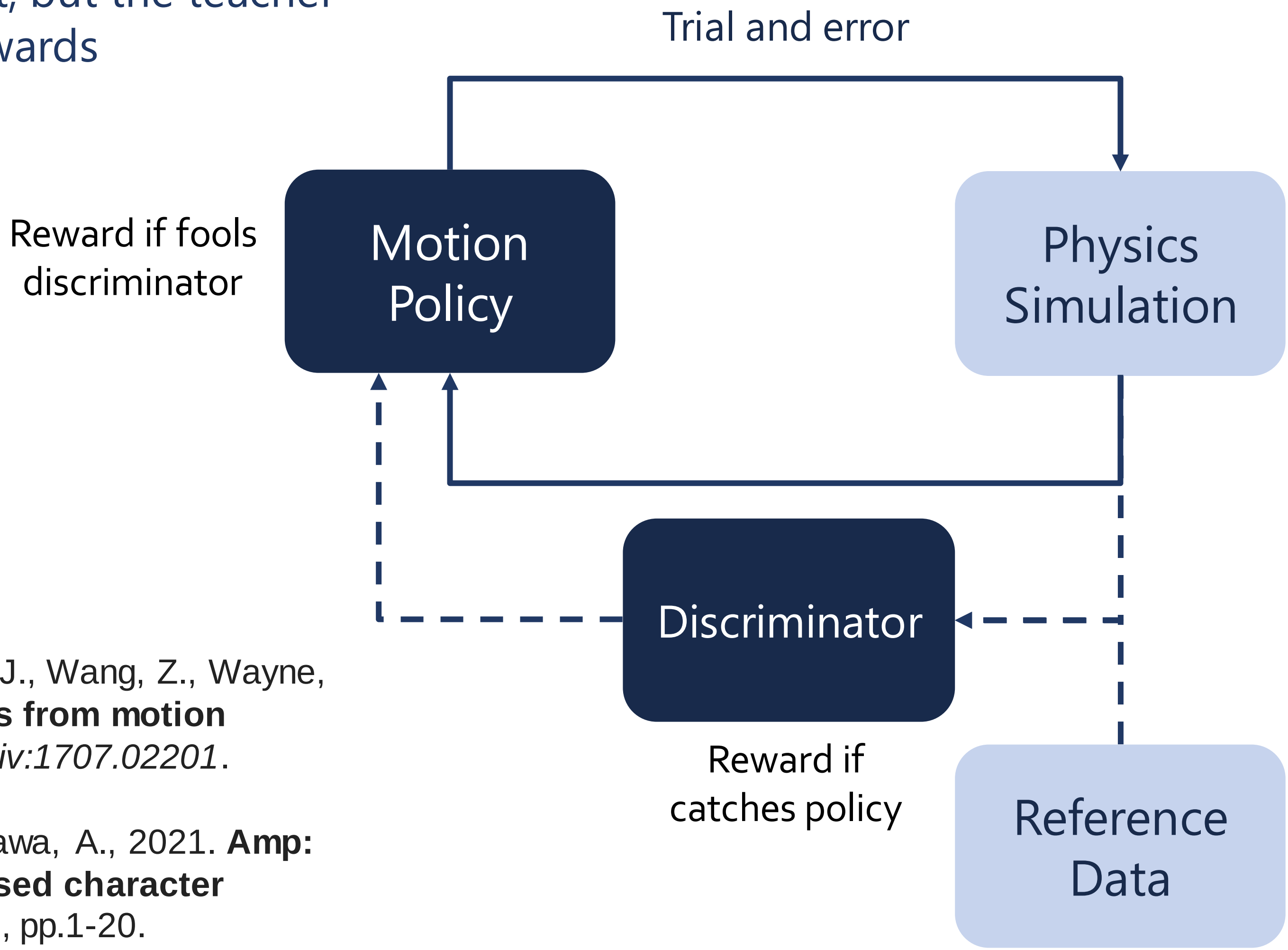
Motion tracking with RL

Limitations:

- Generalizing to challenging movement?
    - Non-cyclic, freeform
    - Socially aware?
- Avoid needing to define and tune cost function?
- Avoid needing kinematic controllers?

Adversarial imitation learning

Simultaneously train teacher and student, but the teacher doesn’t give specific instructions, just rewards



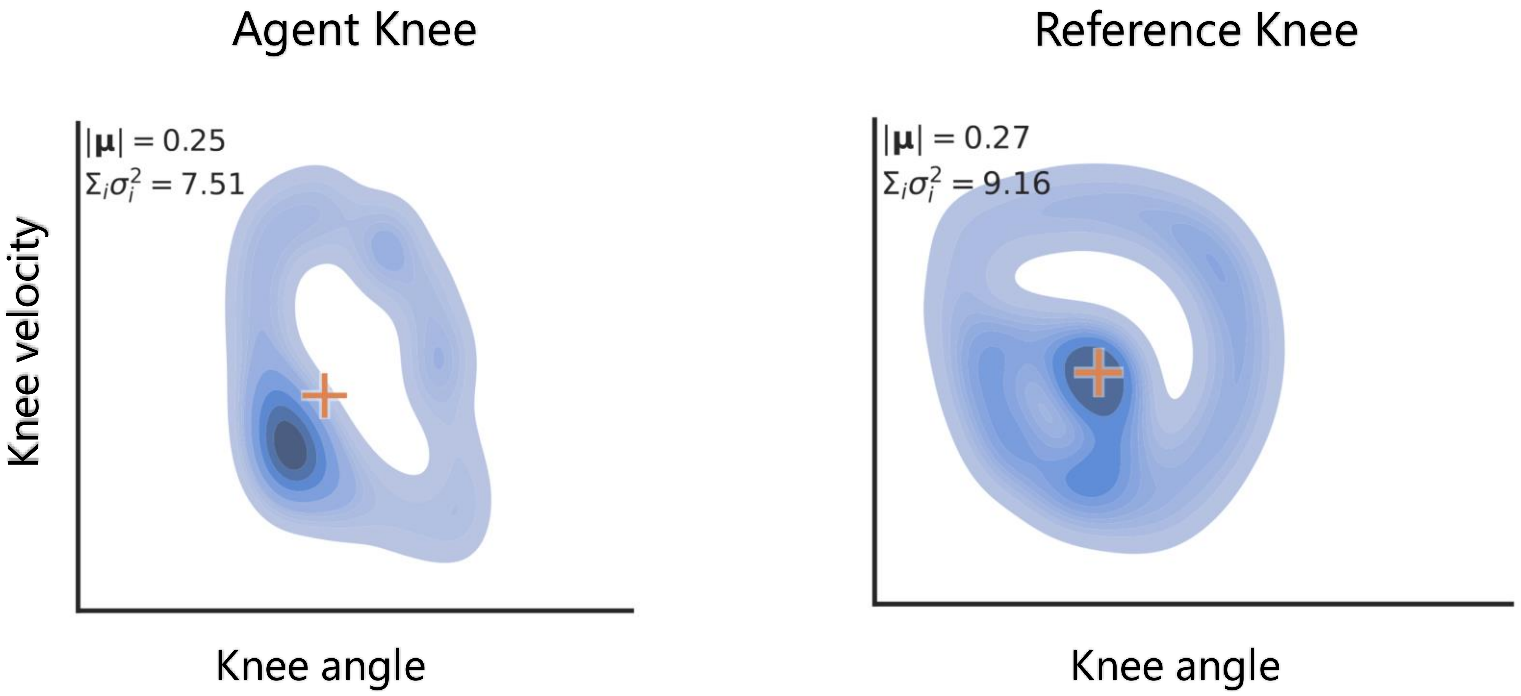
Examples in:

Merel, J., Tassa, Y., TB, D., Srinivasan, S., Lemmon, J., Wang, Z., Wayne, G. and Heess, N., 2017. **Learning human behaviors from motion capture by adversarial imitation**. *arXiv preprint arXiv:1707.02201*.

Peng, X.B., Ma, Z., Abbeel, P., Levine, S. and Kanazawa, A., 2021. **Amp: Adversarial motion priors for stylized physics-based character control**. *ACM Transactions on Graphics (ToG)*, *40*(4), pp.1-20.

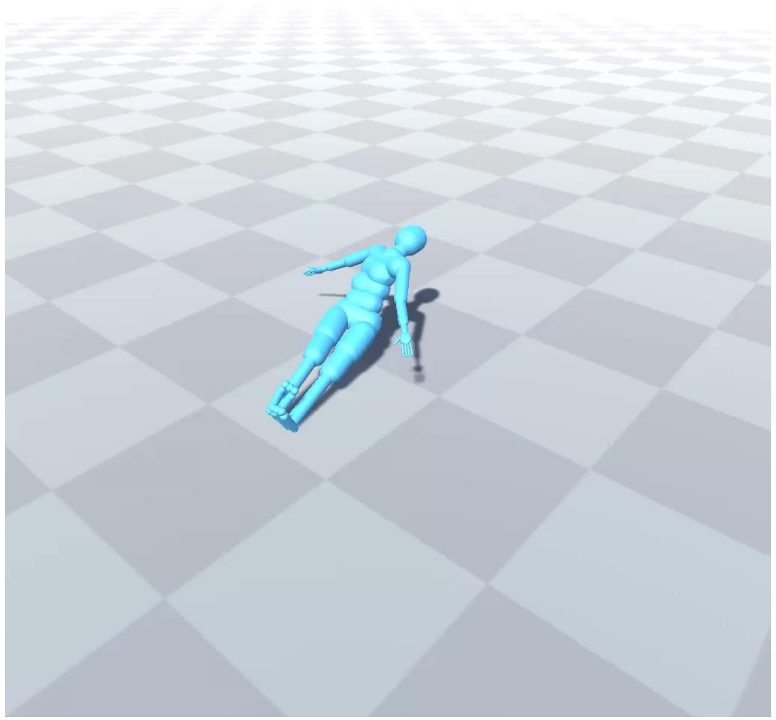
Adversarial Motion Priors

- No state transitions



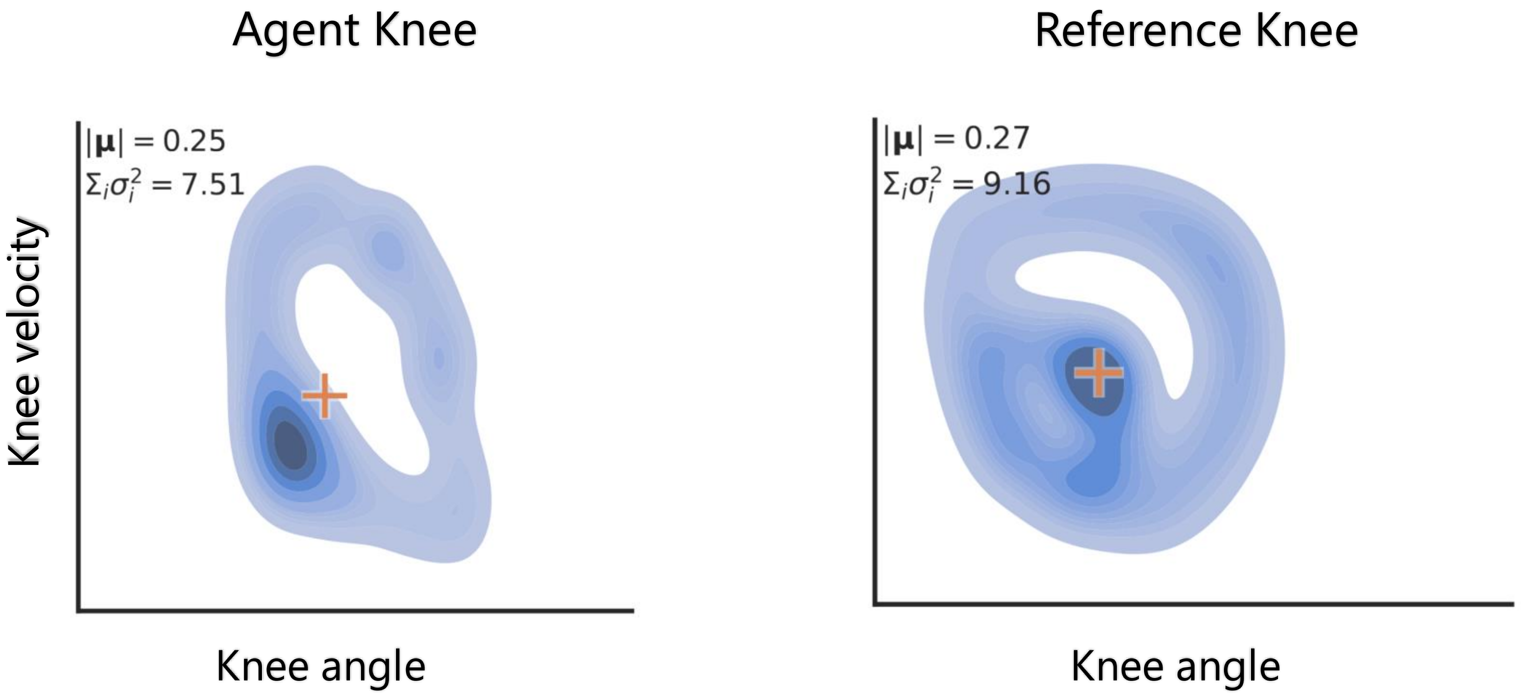
Adversarial Motion Priors

- No state transitions



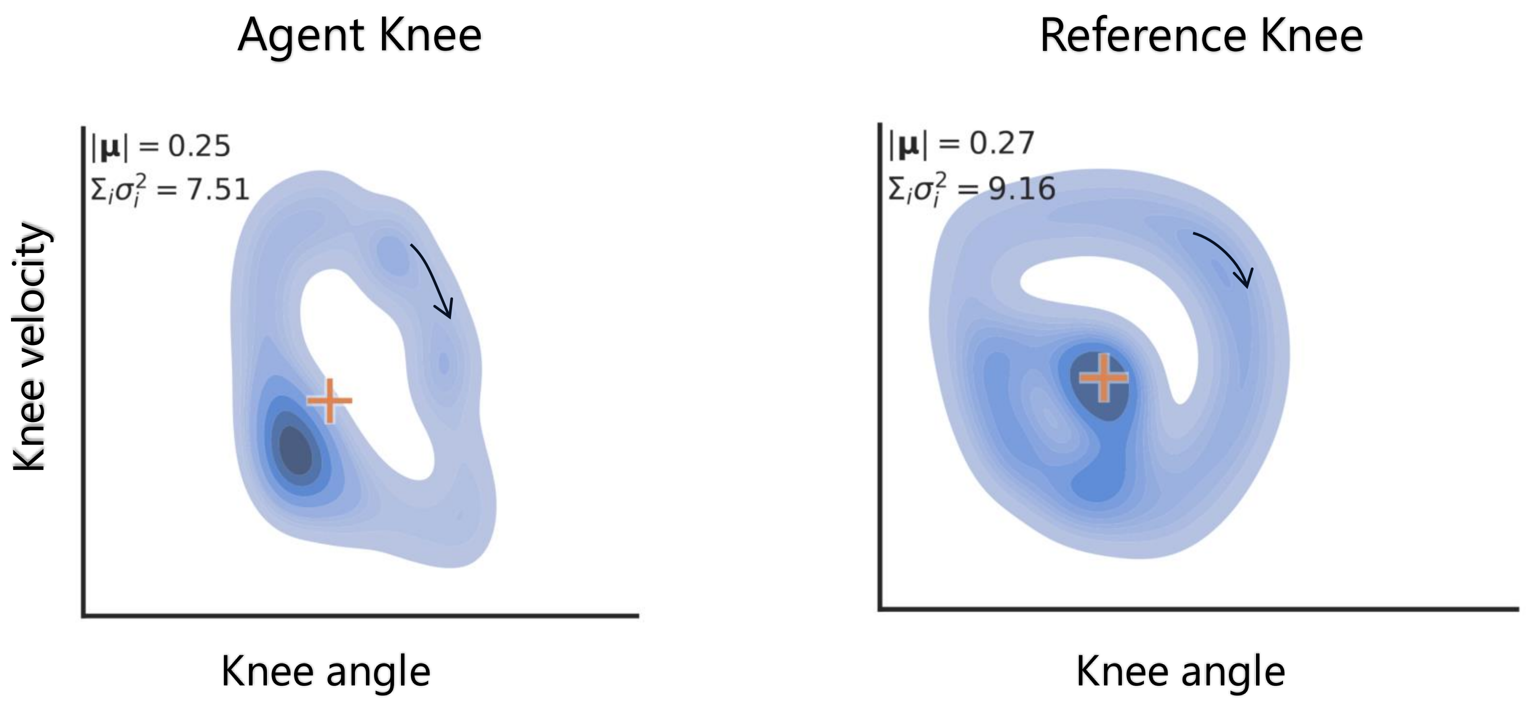
Adversarial Motion Priors

- No state transitions



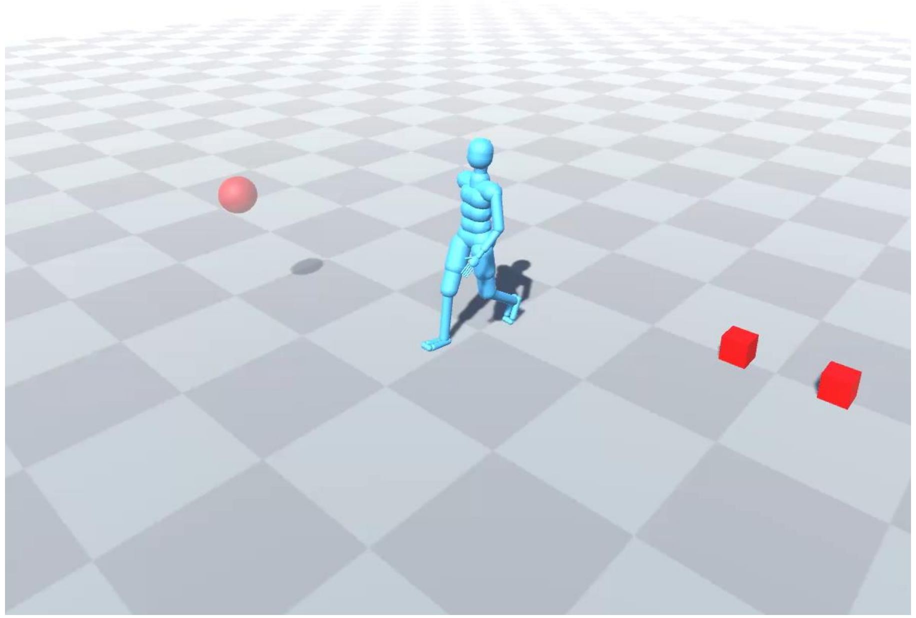
Adversarial Motion Priors

- With state transitions



Adversarial Motion Priors

- With state transitions



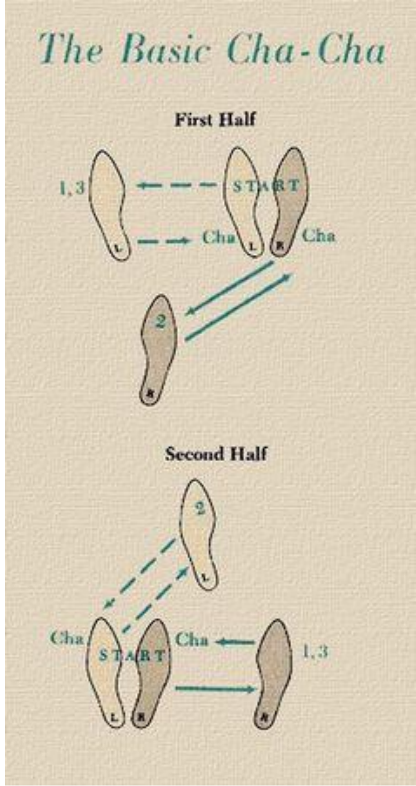
Adversarial Imitation RL

Limitations:

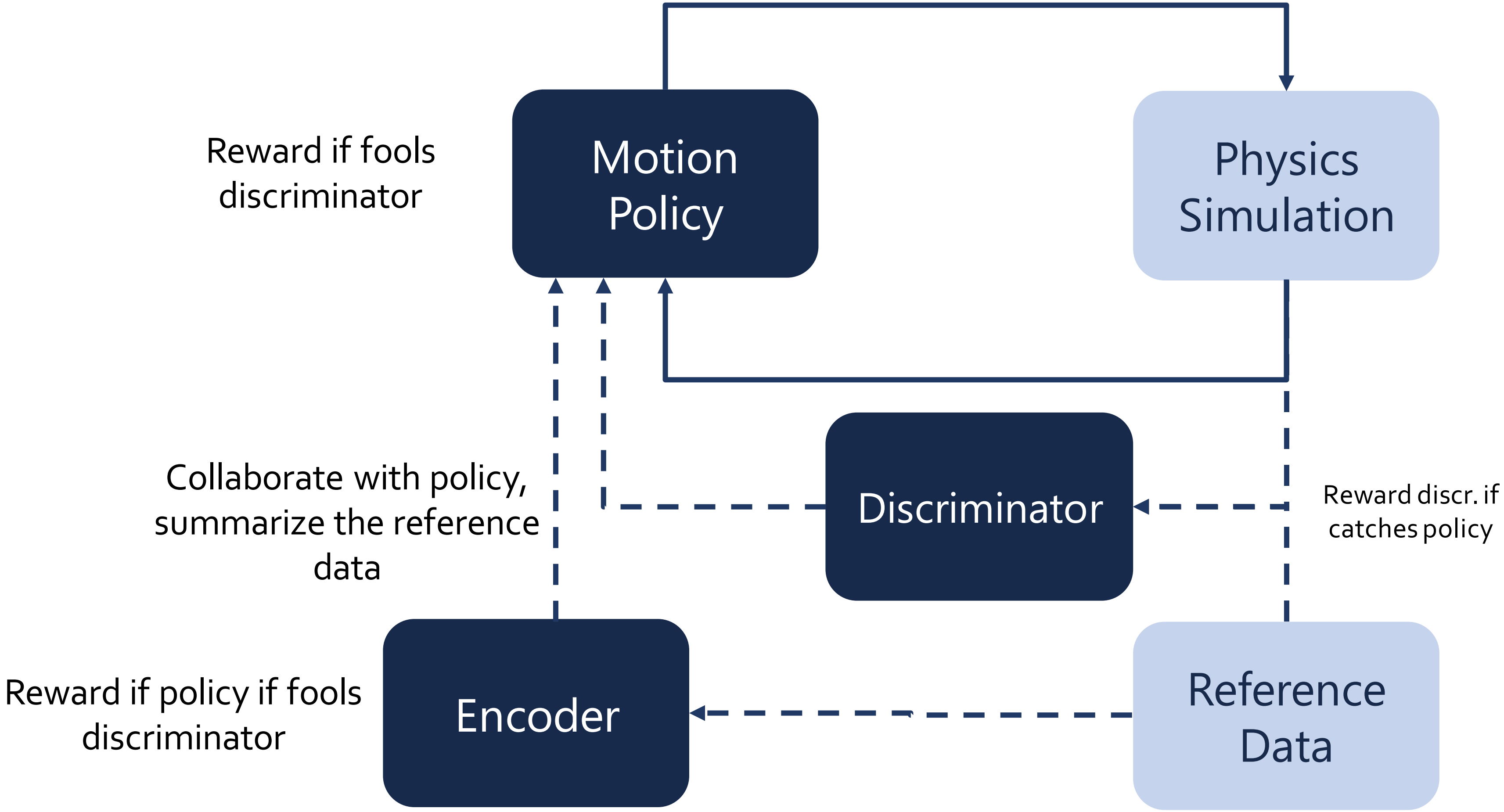
- No need to precisely define imitation reward, but still need reference data
- Need to carefully balance the learning rates and capabilities of the policy vs. the discriminator
- Much slower to learn then motion tracking
- Hard to generalize to lots of motions, and to achieve a range of tasks

Learned hierarchical control

We can learn how to efficiently compress a description of the desired movement for the policy.



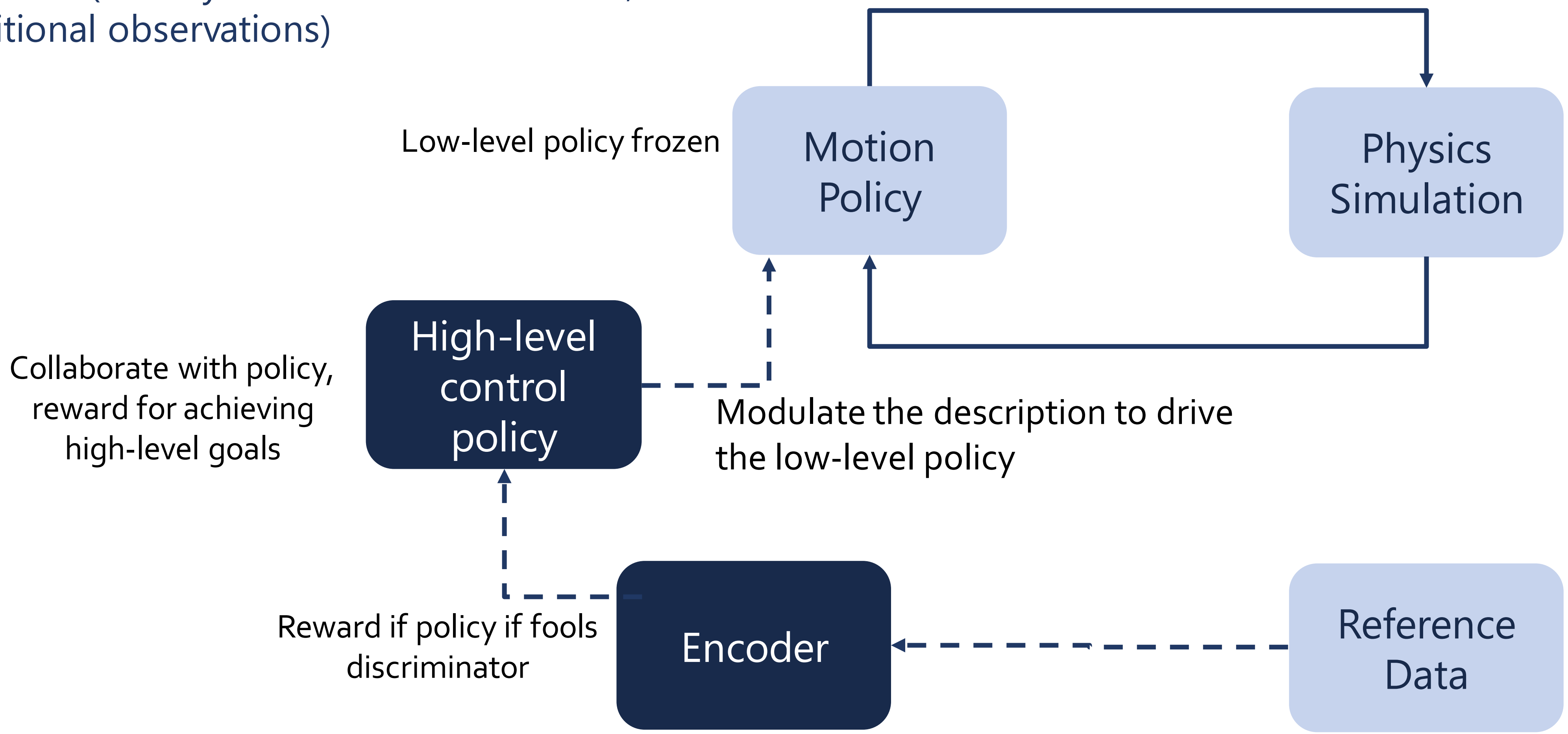
Pretrain:



Learned hierarchical control

Generalize over a range of tasks by learning latent task representations (already learned reward function, also learns additional observations)

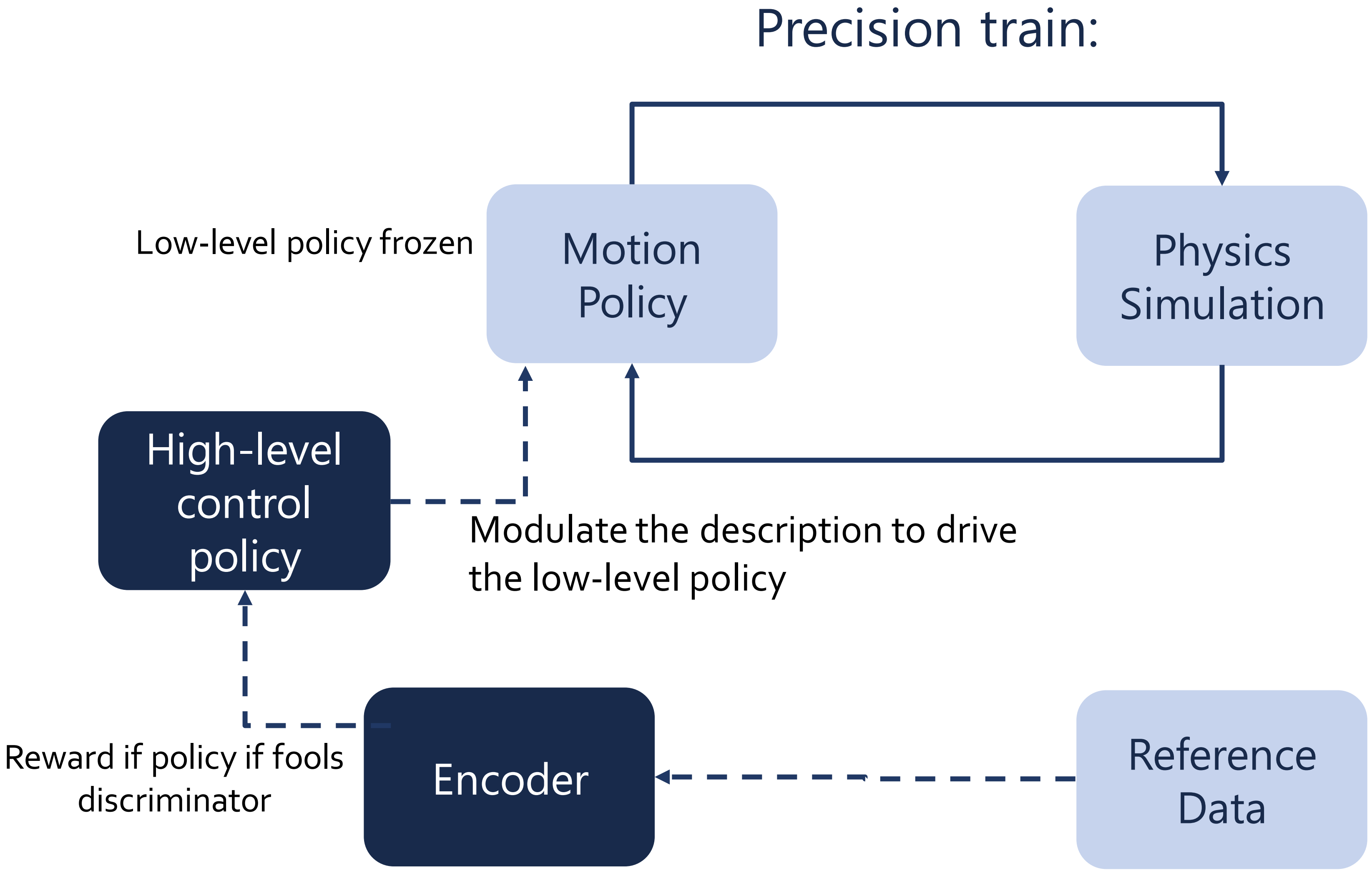
Precision train:



Learned hierarchical control

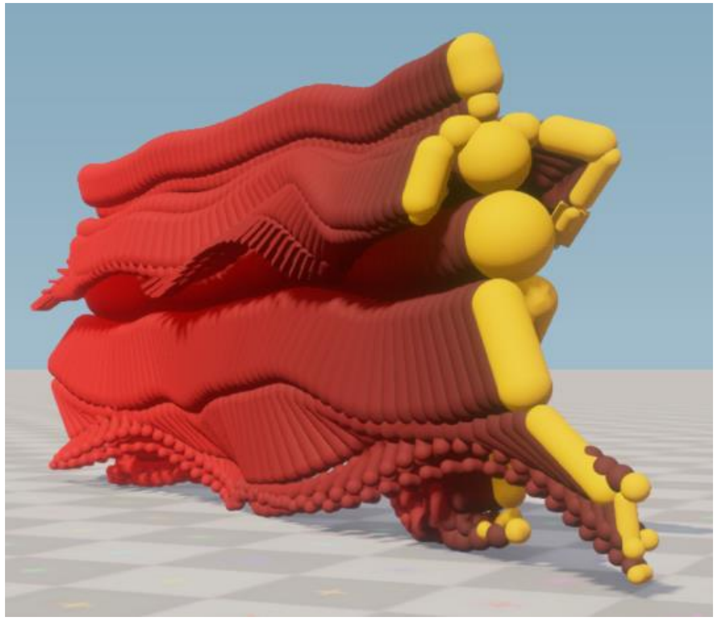
Described in:

Tessler, C., Kasten, Y., Guo, Y., Mannor, S., Chechik, G. and Peng, X.B., 2023, July. **Calm: Conditional adversarial latent models for directable virtual characters.** In *ACM SIGGRAPH 2023 Conference Proceedings* (pp. 1-9).



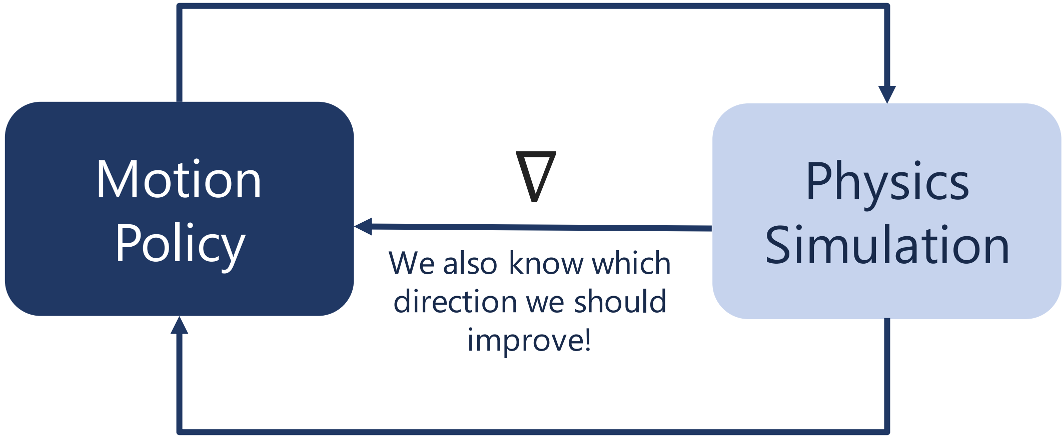
Physics inspired, semi-supervised?

Can exploit the fact that the motion happens in a physics engine, mix in model predictive control



Fussell, L., Bergamin, K. and Holden, D., 2021. **Supertrack: Motion tracking for physically simulated characters using supervised learning**. *ACM Transactions on Graphics (TOG)*, *40*(6), pp.1-13.

Another example:
Ren, J., Yu, C., Chen, S., Ma, X., Pan, L. and Liu, Z., 2023. **Diffmimic: Efficient motion mimicking with differentiable physics**. *arXiv preprint arXiv:2304.03274*.



Not just reward signal, we can get the specific directions.

"Constructive criticism", instead of thumbs up/down.

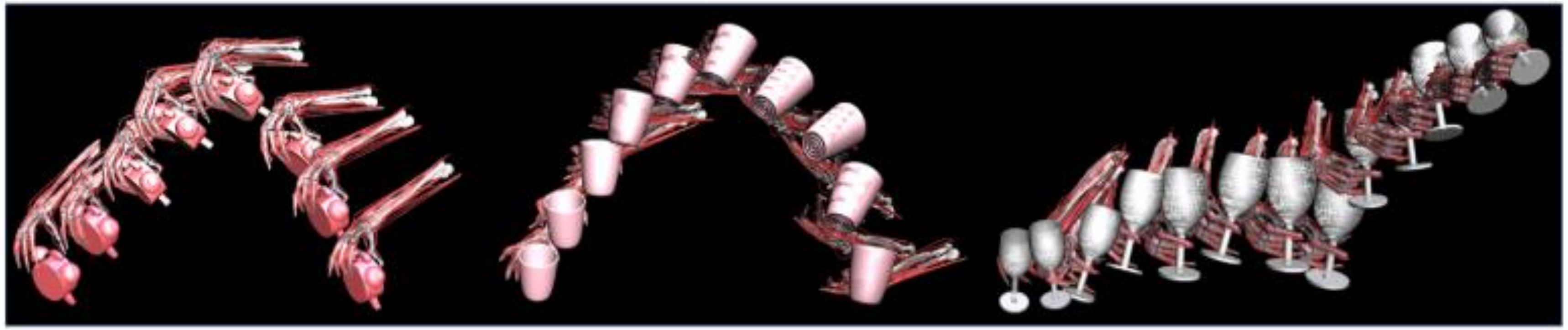
RL in Musculoskeletal models

- Naively applying what worked for joint-torque models fails spectacularly.
- Huge action-state spaces, and redundant systems are the bane of trial-and-error
- Need good human MSk models, that can run fast (RL is usually not sample efficient!)
- Example: MyoSuite (OpenSim models translated to MuJoCo for RL)

RL in Musculoskeletal models

Caggiano, V., Dasari, S. and Kumar, V., 2023, July. **MyoDex: a generalizable prior for dexterous manipulation.** In *International Conference on Machine Learning* (pp. 3327-3346). PMLR.

Example: MyoDex (manipulation agent from the MyoSuite team)

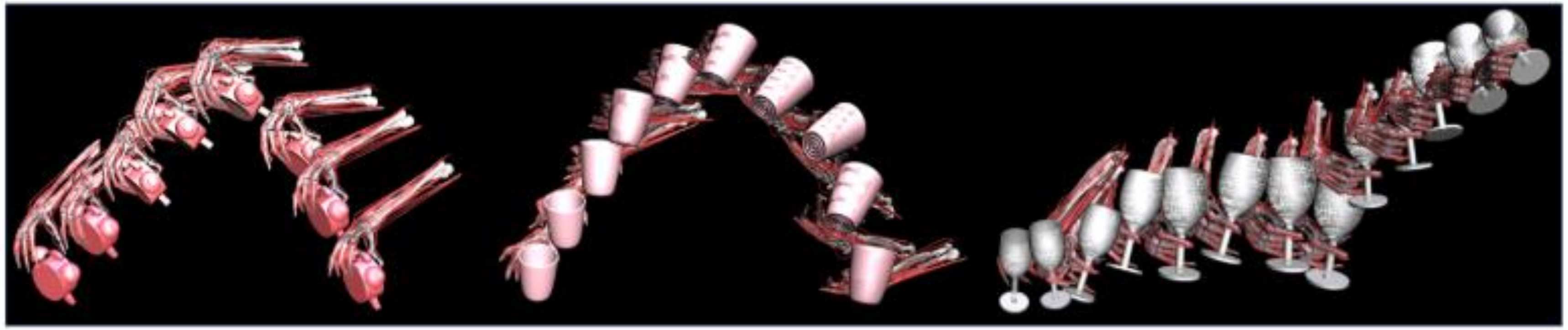


RL in Musculoskeletal models

Caggiano, V., Dasari, S. and Kumar, V., 2023, July. **MyoDex: a generalizable prior for dexterous manipulation.** In *International Conference on Machine Learning* (pp. 3327-3346). PMLR.

Example: MyoDex (manipulation agent from the MyoSuite team)

Observations: Joint space kinematics, target object orientation
Actions: Muscle activations
Rewards: Negative muscle activation vector magnitude,
match object trajectory

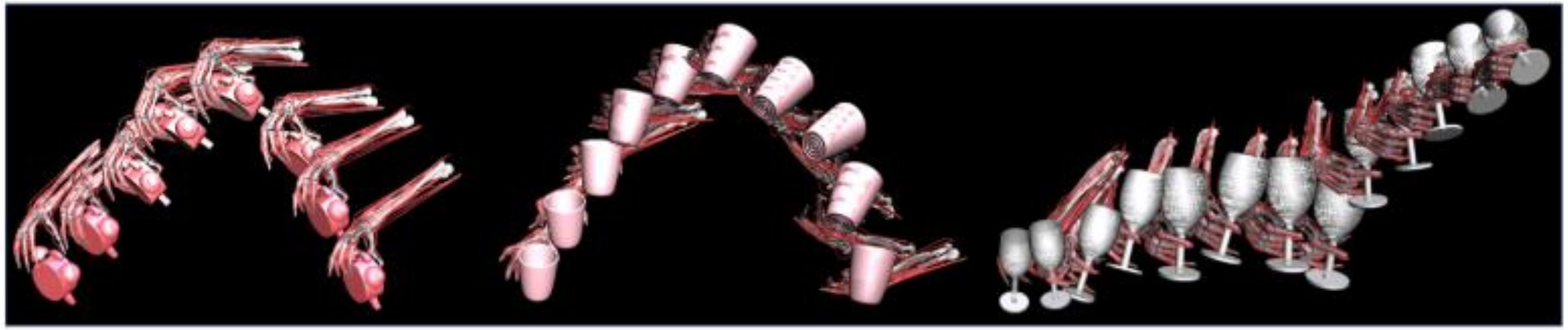


RL in Musculoskeletal models

Caggiano, V., Dasari, S. and Kumar, V., 2023, July. **MyoDex: a generalizable prior for dexterous manipulation.** In *International Conference on Machine Learning* (pp. 3327-3346). PMLR.

Example: MyoDex (manipulation agent from the MyoSuite team)

Observations: Joint space kinematics, target object orientation
Actions: Muscle activations
Rewards: Negative muscle activation vector magnitude,
match object trajectory



Idea to tackle challenges: pretrain policy on a subset of smaller/easier tasks simultaneously, then the generalist model can be specialized to harder tasks.

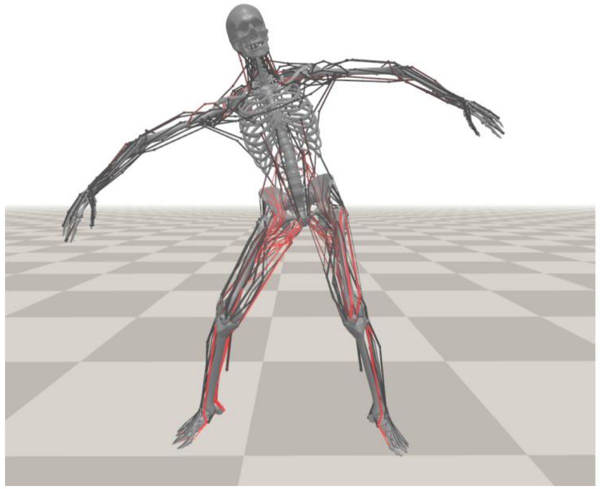
(A bit like curriculum learning)

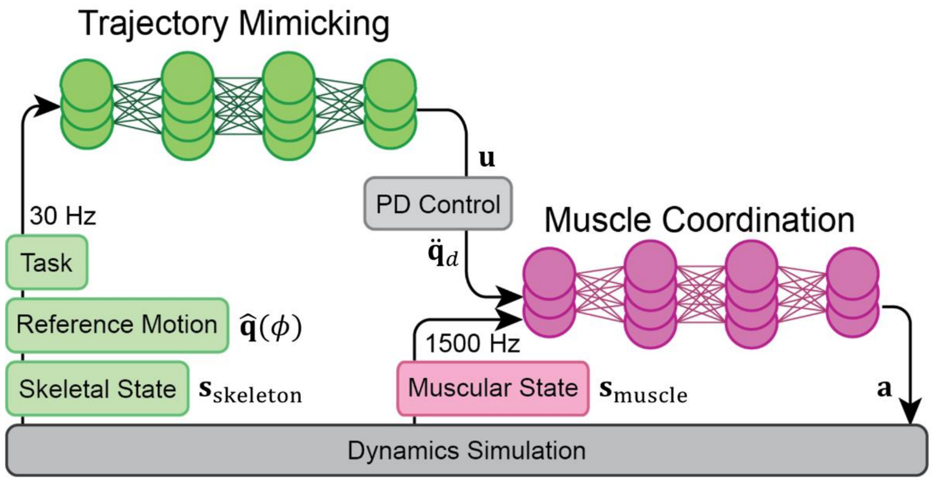
RL in Musculoskeletal models

Park, J., Min, S., Chang, P.S., Lee, J., Park, M.S. and Lee, J., 2022, July. **Generative gaitnet**. In *ACM SIGGRAPH 2022 Conference Proceedings* (pp. 1-9).

Lee, S., Park, M., Lee, K. and Lee, J., 2019. **Scalable muscle-actuated human simulation and control.** *ACM Transactions On Graphics (TOG)*, *38*(4), pp.1-13.

Example: GaitNet





Idea to tackle challenges: Hierarchical control, same type of PD-style joint torque controller that outputs target joint torques, and submodule that translates desired joint torque to activation.

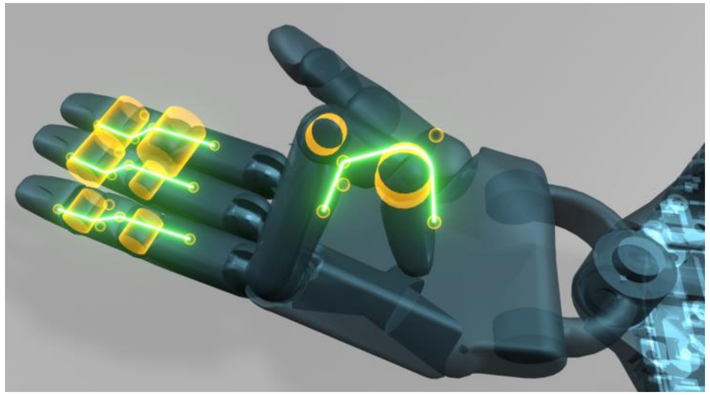
Parametrize MSk parameters, condition policy on it! One policy good for multiple subjects.

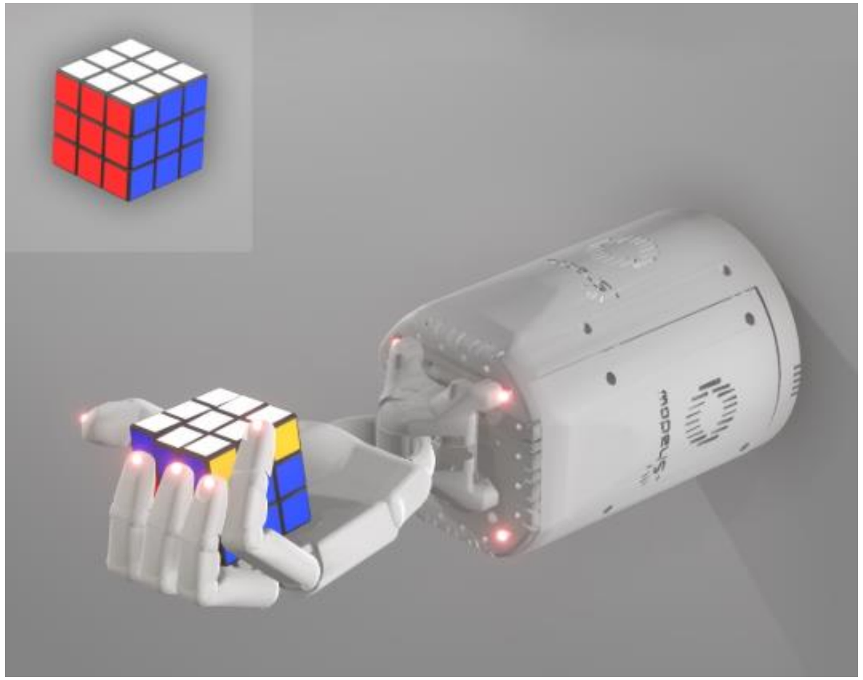
Disentangle the high-level goal and the way to get there

Generalization and Sim2Real transfer

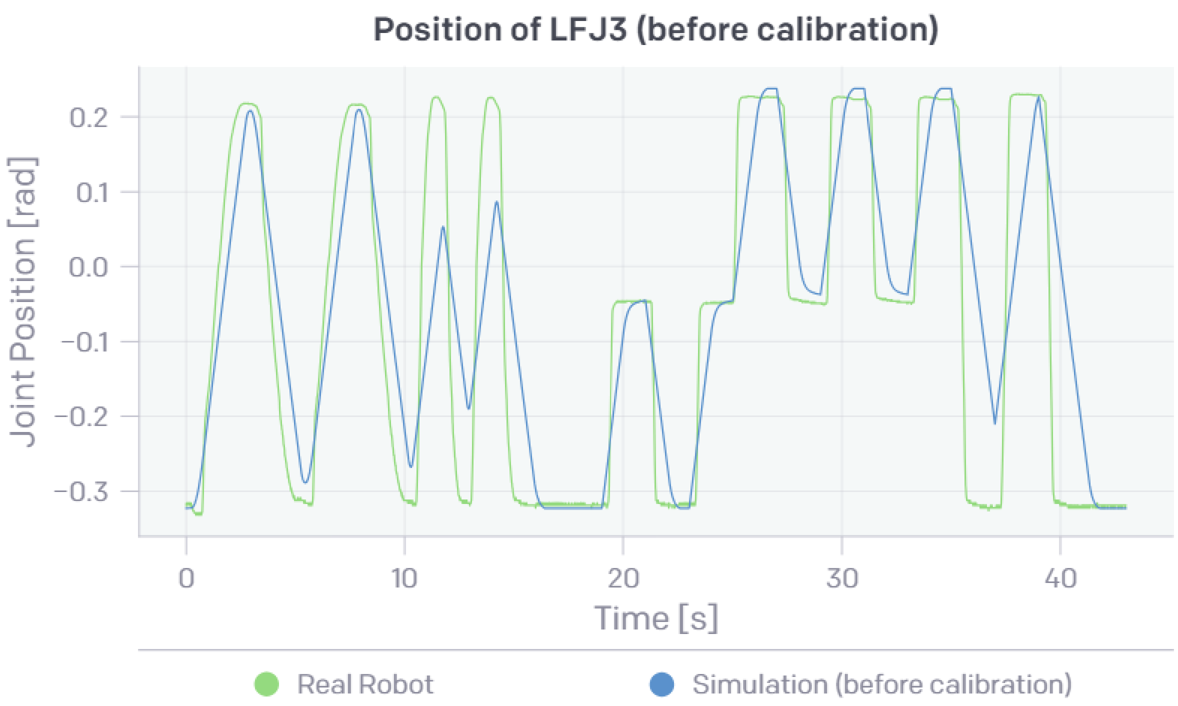
Akkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., Petron, A., Paino, A., Plappert, M., Powell, G., Ribas, R. and Schneider, J., 2019. **Solving rubik's cube with a robot hand.** arXiv preprint arXiv:1910.07113.

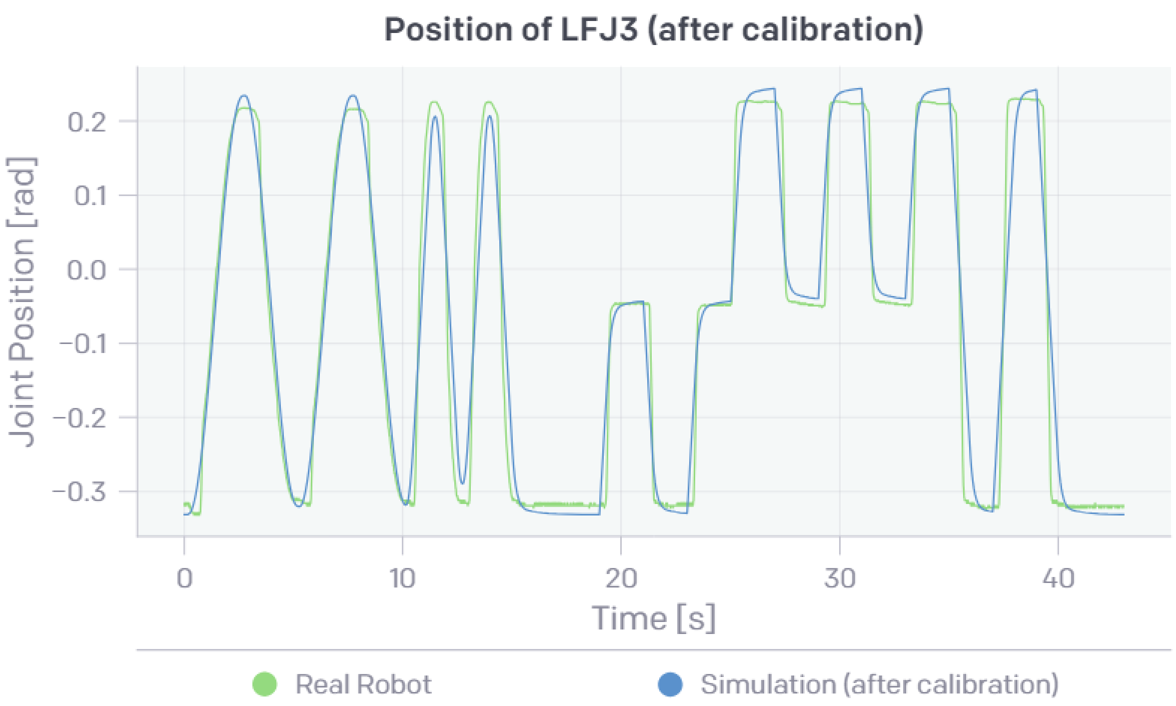
Example: OpenAI + Shadow hand





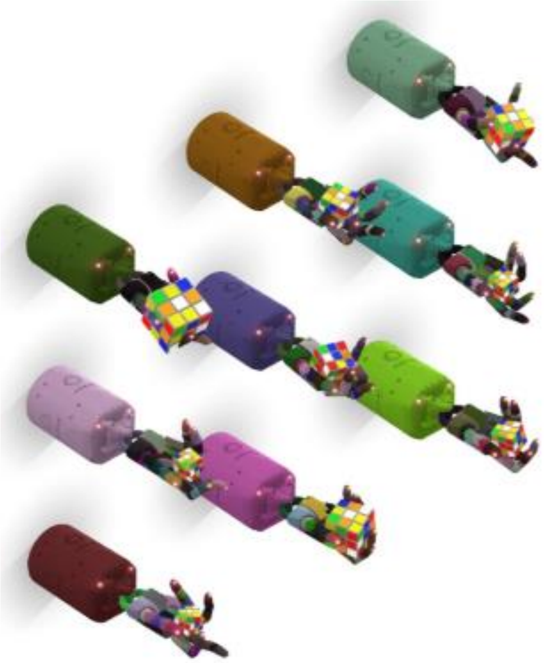
Generalization and Sim2Real transfer

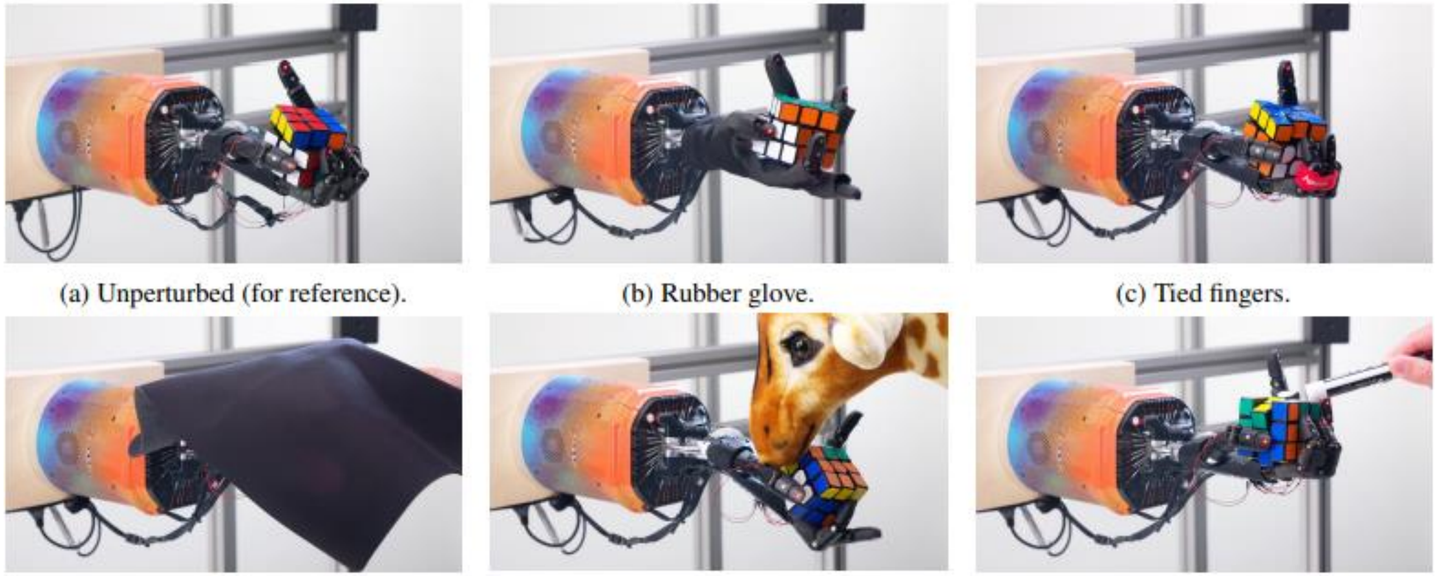




System Identification, reproduce experimental data

Generalization and Sim2Real transfer





(a) Unperturbed (for reference). (b) Rubber glove. (c) Tied fingers.

(d) Blanket occlusion and perturbation. (e) Plush giraffe perturbation.[17] (f) Pen perturbation.

Automatic domain randomization + systematic consistent random perturbations (not white noise!)

Thank you for your attention!

Recommended literature:

Song, S., Kidziński, Ł., Peng, X.B., Ong, C., Hicks, J., Levine, S., Atkeson, C.G. and Delp, S.L., 2021. Deep reinforcement learning for modeling human locomotion control in neuromechanical simulation. *Journal of neuroengineering and rehabilitation*, *18*, pp.1-17.

Thanks to funding sources:

EPSRC Centre for DOCTORAL TRAINING PROSTHETICS & ORTHOTICS

NATURAL BIONICS

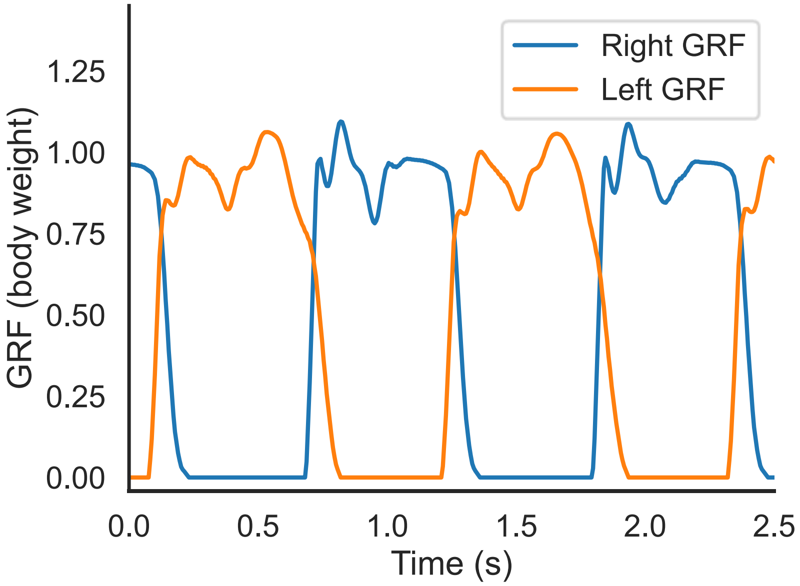
Project Aims and Contributions

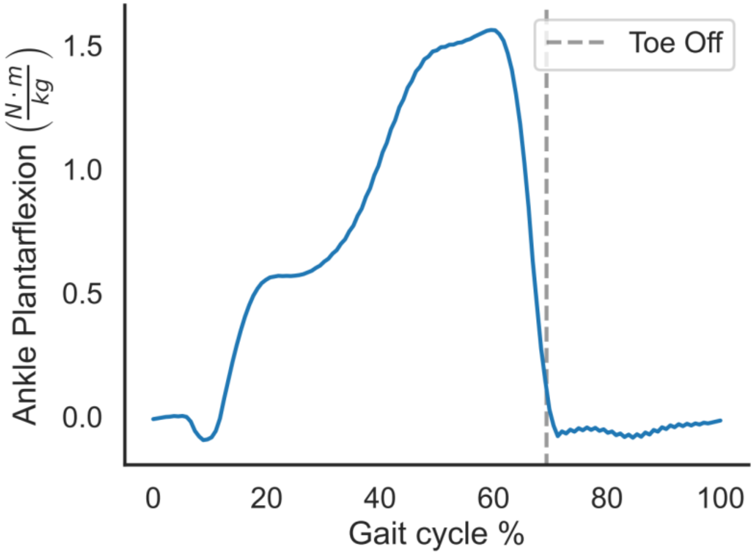
- Investigate simulated assistive technology in non-steady-state locomotion settings

- Explore the benefit of intent-driven devices and control-strategies

Gait characteristics

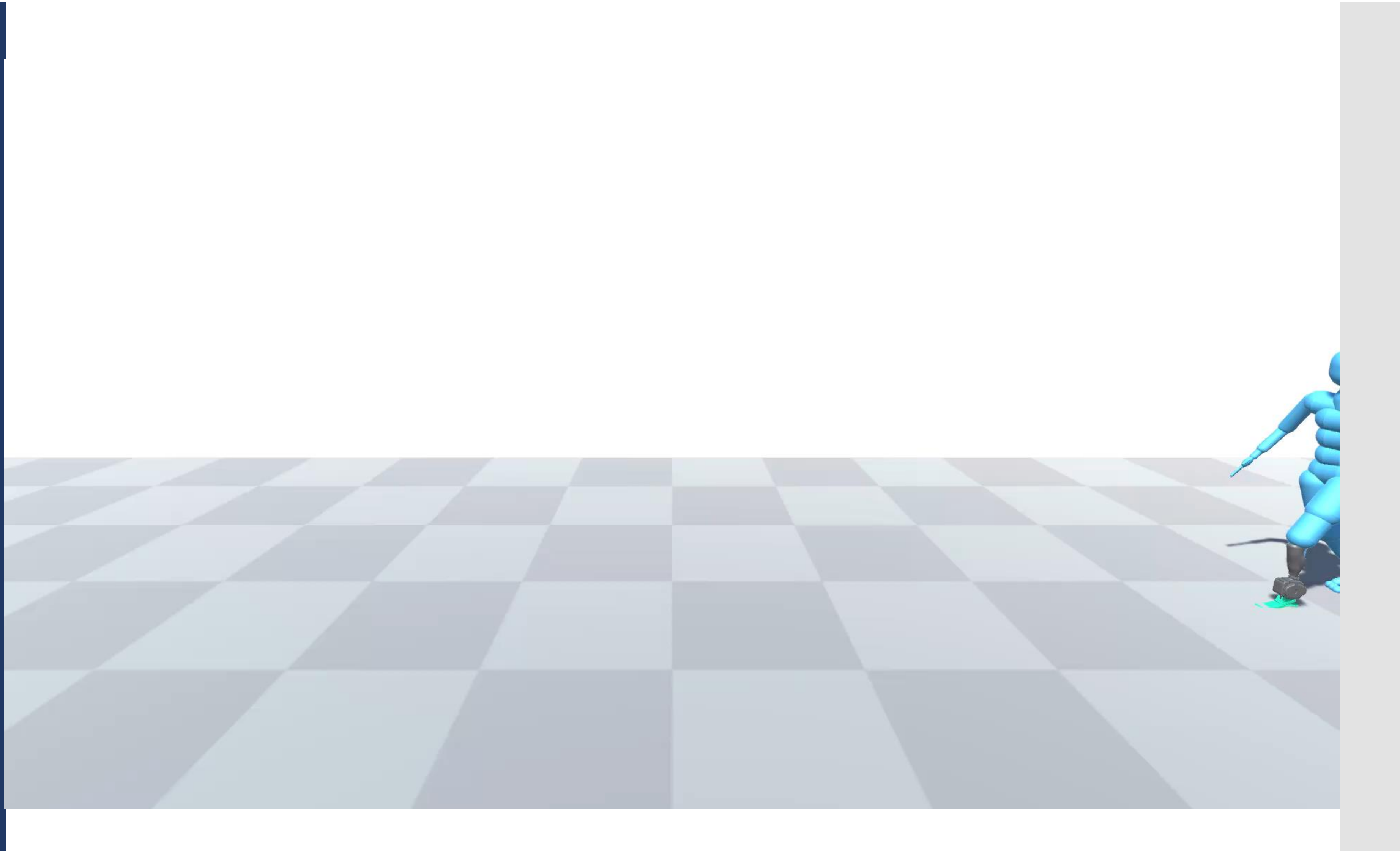
- Key aspects of unperturbed gait emerge, despite not constraining for them





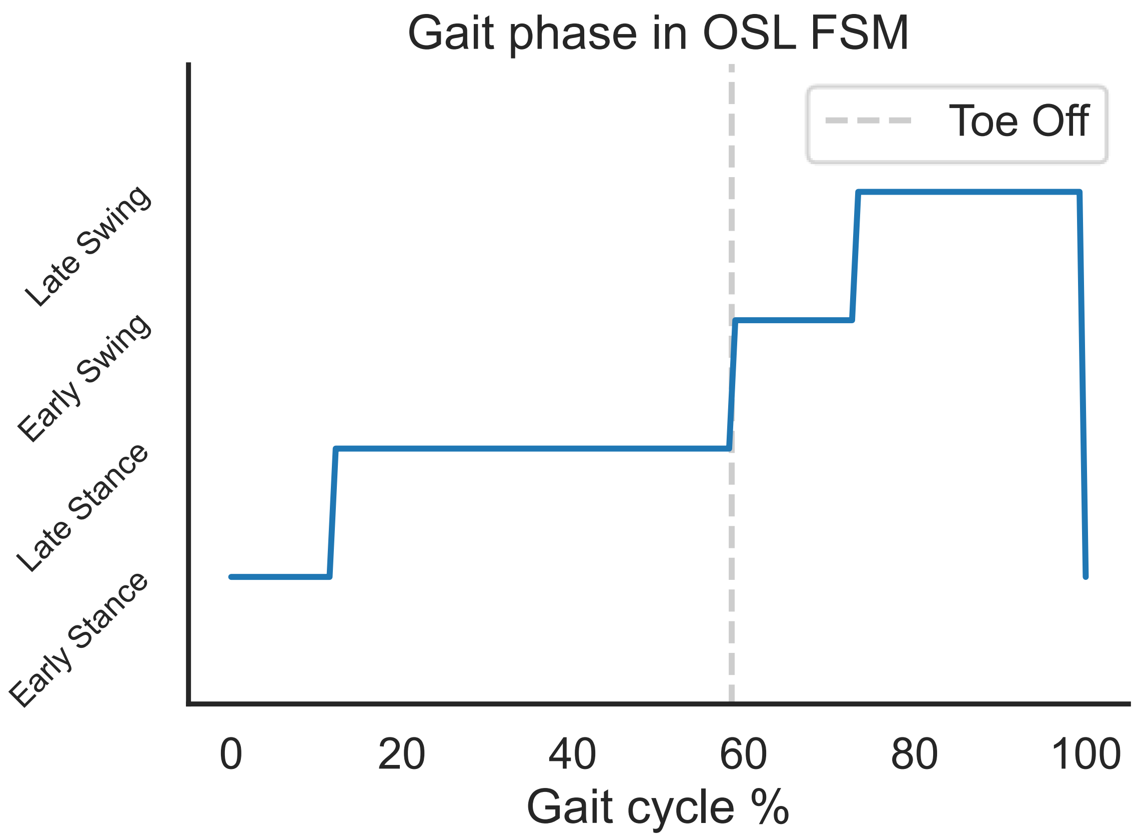
OSL FSM based controller

- Replicate and explore non - ML models for comparisons and iterations on the controller

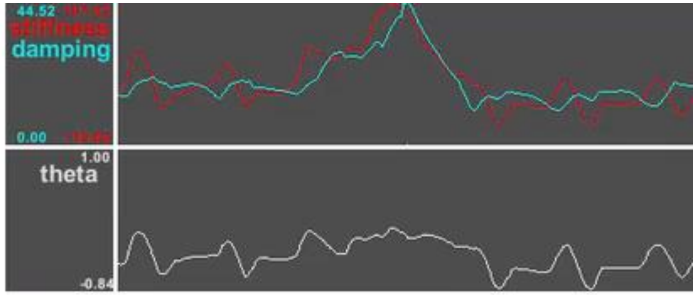


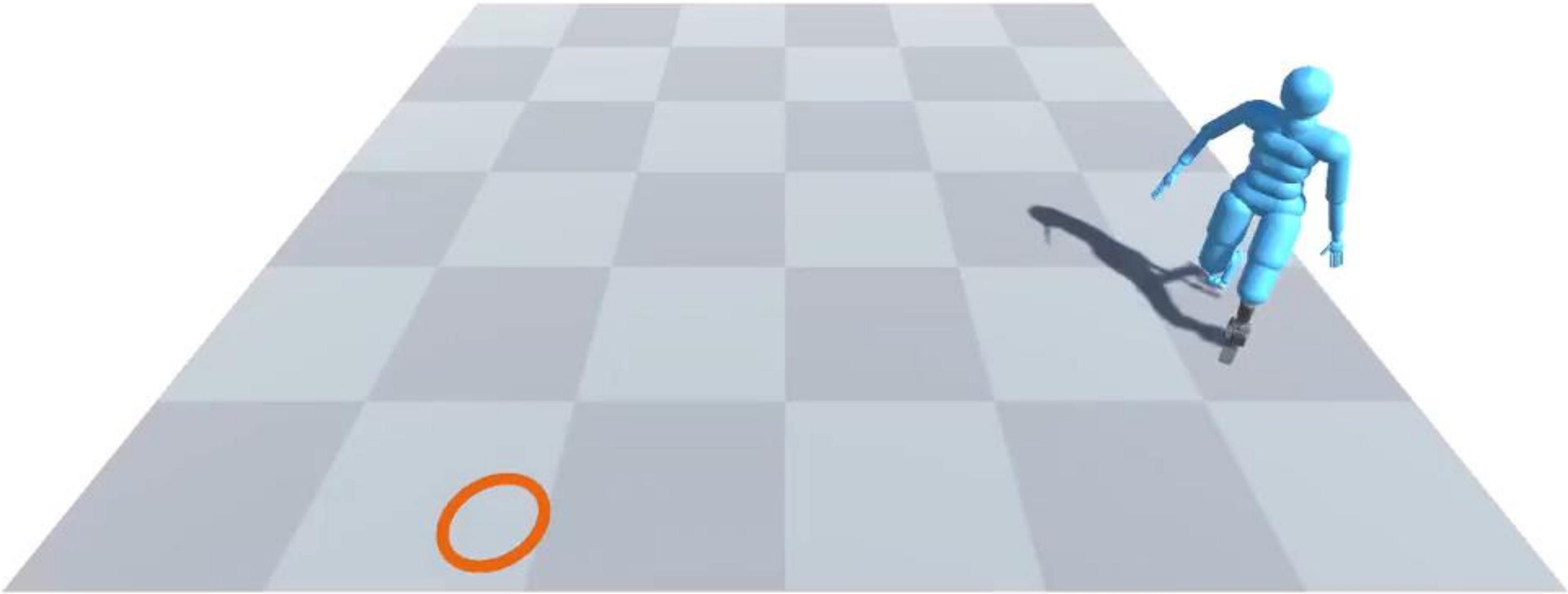
OSL FSM based controller

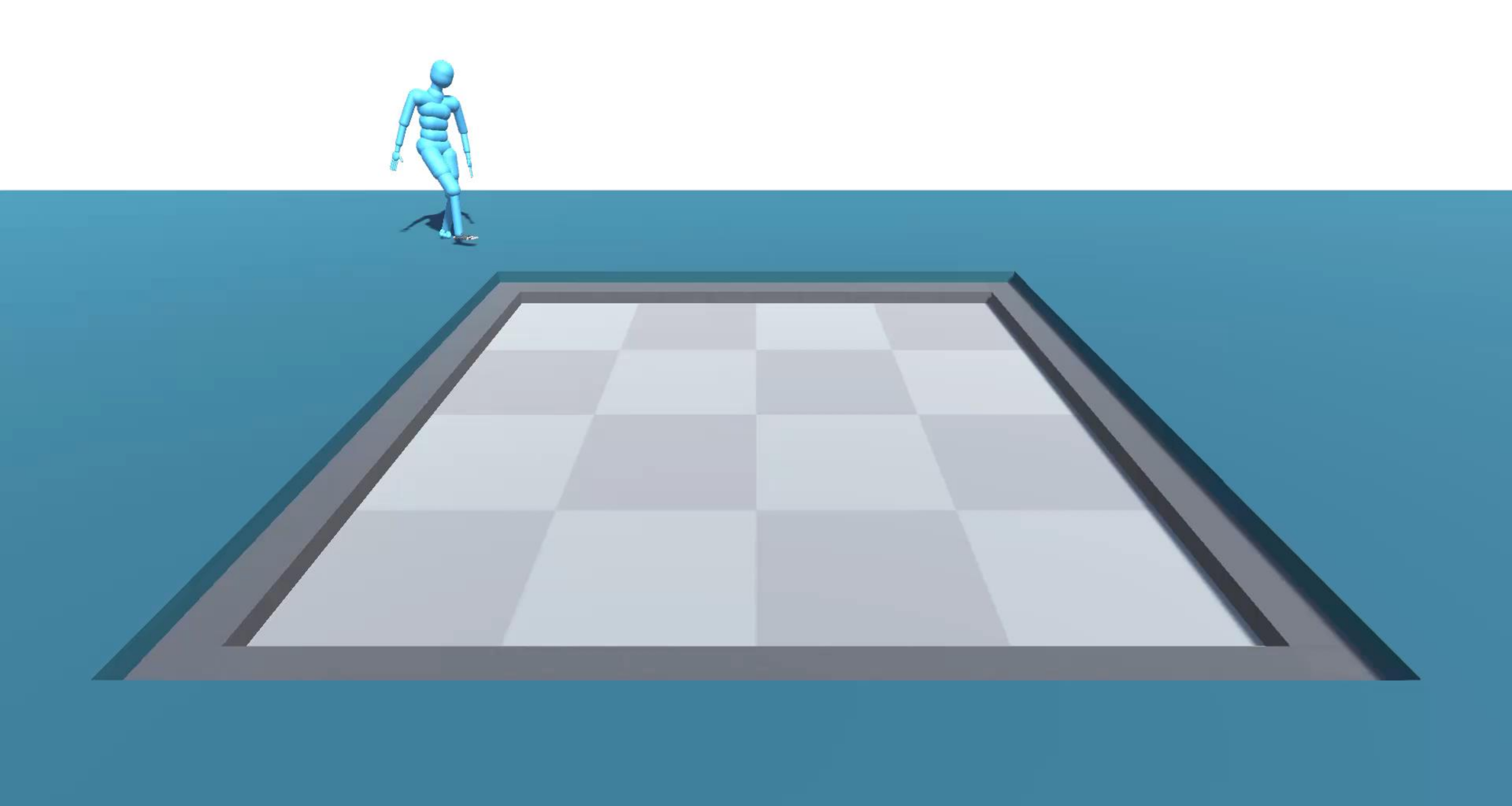
- Use as validation for kinematic and kinetic context of the virtual prosthesis



Prosthesis use in non-steady-state locomotion







Model of the IIT SoftFoot developed with Matteo Crotti

Comparing designs in simulation

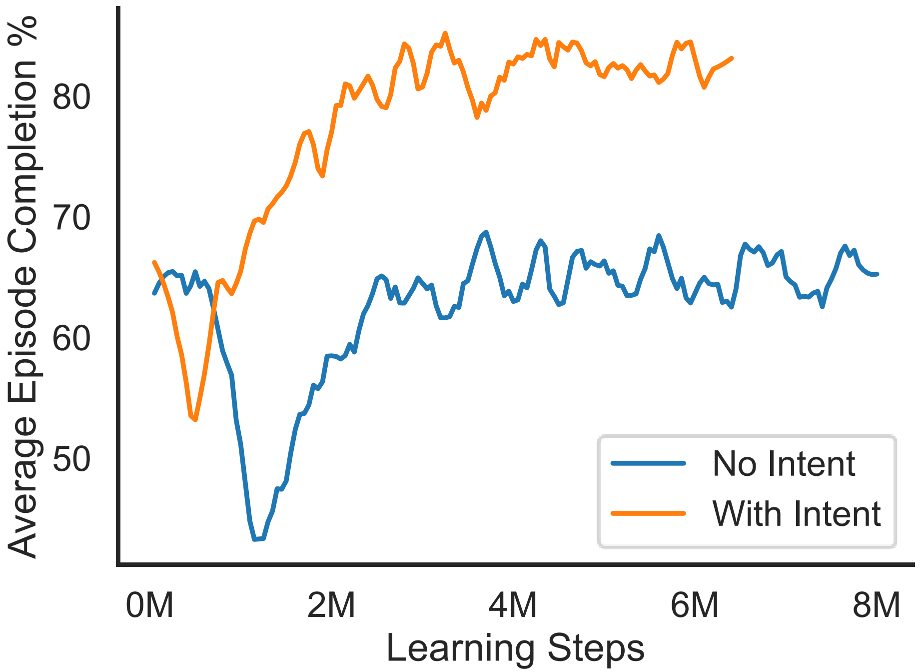
Assist perturbed gait:

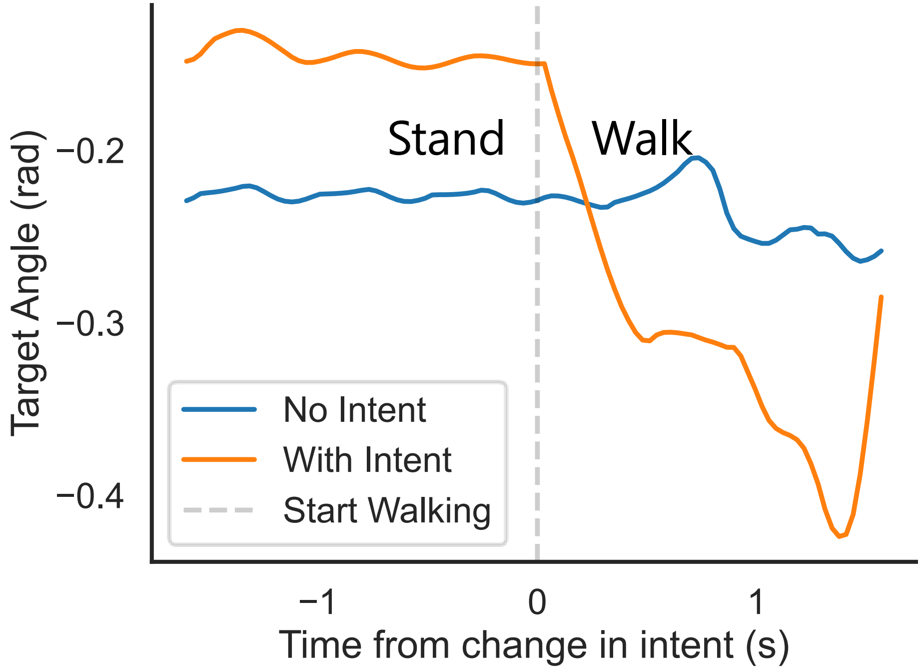
- Passive prosthesis
  ↪ Possible with compensatory movement

Active assistance:

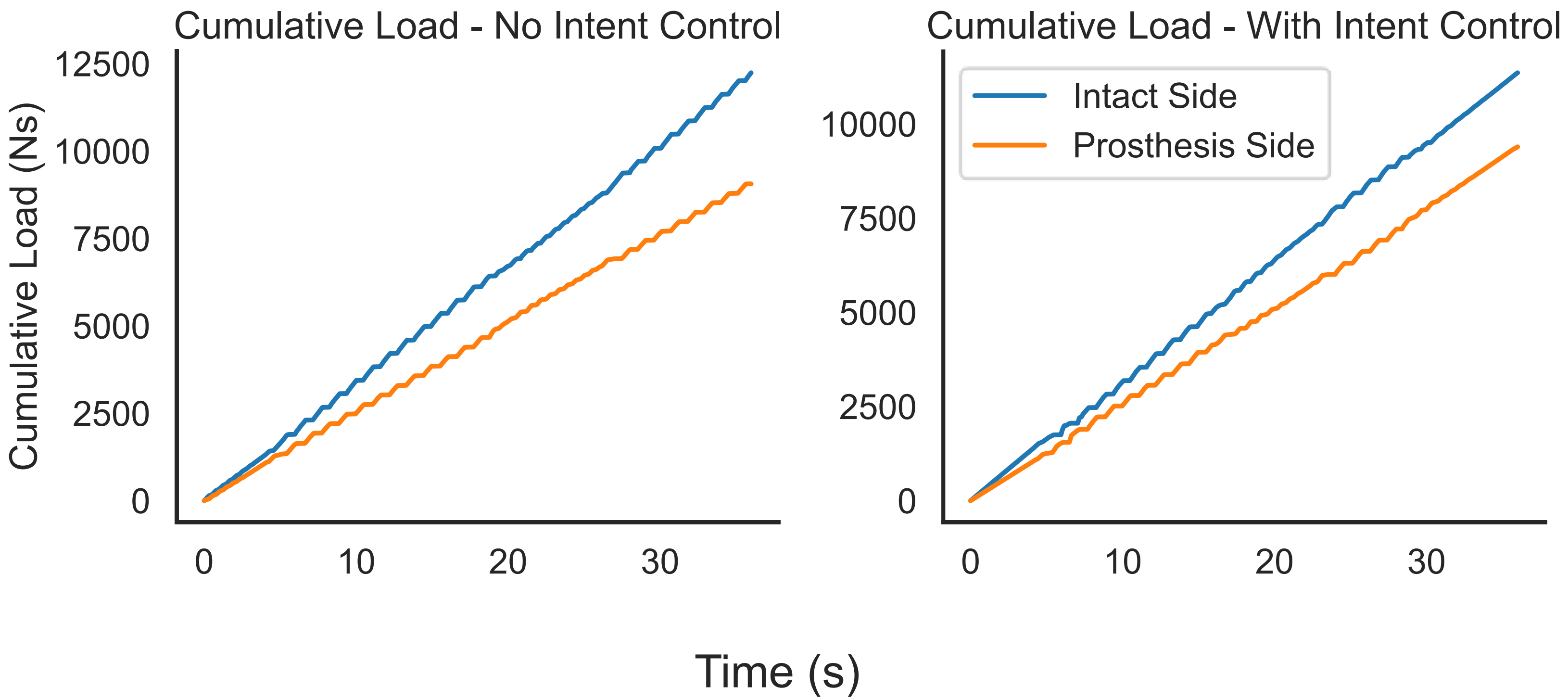
- Impedance control
- Impedance control with high-level intent

Effects of actuation and intent-based control

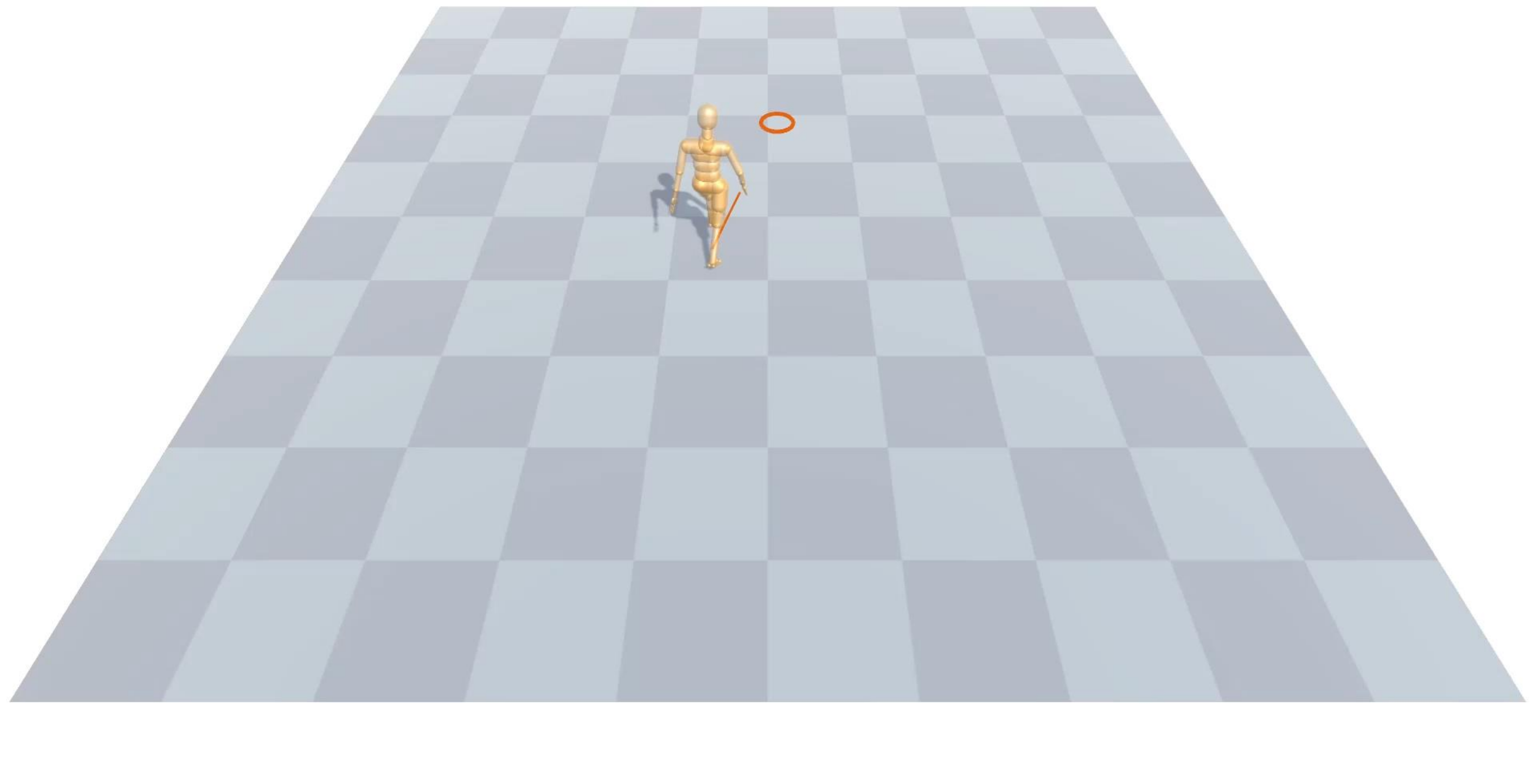




Cumulative Load



Non-steady-state locomotion



Future work

- Limiting proprioception on one side leads to increased foot clearance and reduced stability
  ↳ Test different kinds of sensory feedback on foot clearance to see which can restore gait