Physics-Based Character Animation/Control with Deep Reinforcement Learning

Inria Center At Rennes University

Introduction

Physically capable agents have wide-ranging impacts:

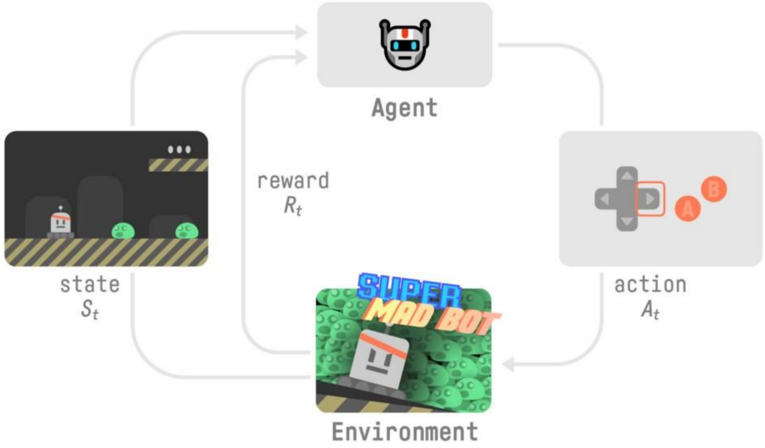
- Expanded Robot Operability: Human-like physical abilities for robots can extend operational domains beyond lab/factory settings into real-world challenging environments.
- Naturalistic Virtual Characters: Enhanced virtual character movements open doors for realistic graphics, eliminating artist intervention, and offering immersive user experiences.
- Biomechanics & Rehab: Advanced models of human motions support biomechanics studies, injury prevention, physiotherapy, customized prosthetics enhancing users' natural ranges of motion.

Peng Xue Bin, Acquiring Motor Skills Through Motion Imitation and Reinforcement Learning 2021

Notions on Reinforcement Learning

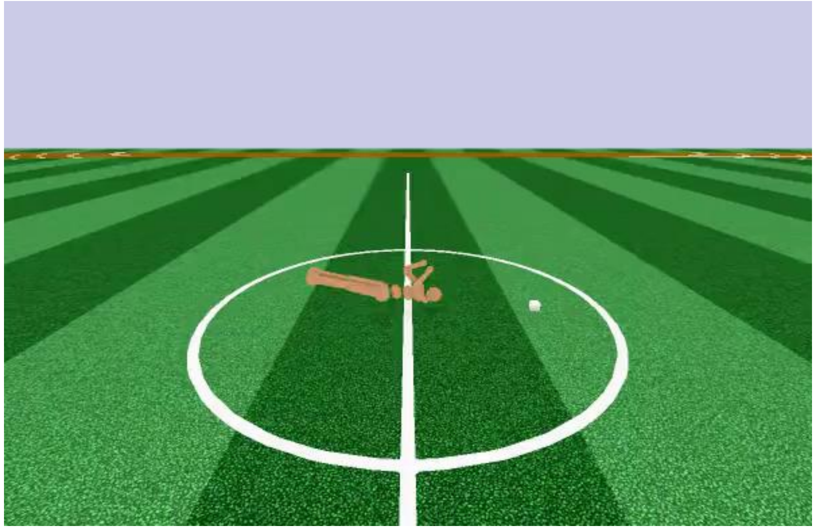
- An agent get observation of the state of the world, decides on an action to take. The environment changes when the agent acts on it.
- The agent perceives a reward signal from the environment, a scalar that tells how good or bad the current world state is after the previous action.

Goal: to maximize its cumulative reward.



Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Use of Reinforcement Learning

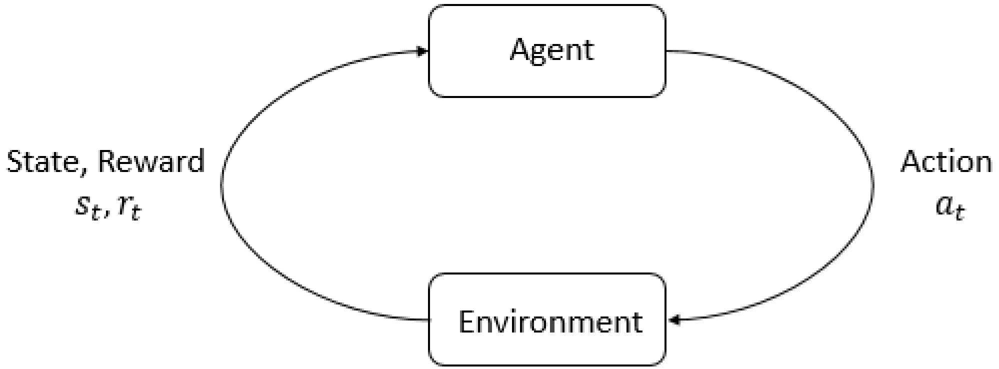


https://spinningup.openai.com/

https://humanoid-transformer.github.io/

Notions on Reinforcement Learning

- States and observations,
- Action spaces,
- Policy
- Trajectories
- Rewards and Return
- RL optimization



Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Notions on Reinforcement Learning

States and observations:

- **State s**: Description of the entire World State, containing absolutely no hidden information. Knowledge of (s) enables the Agent to grasp the full context of the Environment without any uncertainties.
- **Observation o**: Fragmentary portrayal of the actual State. It offers limited insights compared to knowing the whole State.

Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Notions on Reinforcement Learning

Action Spaces:

Various Environments accommodate diverse sets of feasible Actions.

- **Discrete Action Spaces**: In certain classic games, such as Atari or Go, the Agent utilizes a restricted collection of permissible Moves
- **Continuous Action Spaces**: Agent manipulates objects within Physical Worlds, Actions correspond to multidimensional Real-Valued Vectors.

Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Notions on Reinforcement Learning

Policy:

- Guidelines directing an Agent to select Actions
- Two major classes: Deterministic & Stochastic

– **Deterministic Policies**

- Represented as: $\mu(s_t)$
- Generates constant Actions corresponding to a State (s_t)

Example: Automobile driving at fixed speed regardless of road conditions

– **Stochastic Policies**

- Denoted as: π(·| st)
- Yields randomized Actions depending on a State (st)

Example: Autonomously deciding speed limits based on traffic flow probabilities

Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Notions on Reinforcement Learning

Trajectories:

- Sequences of linked States & Actions in the Environment
- Ordered series of alternating States (s_t) & Actions (a_t): $\tau=(s_0, a_0, s_1, a_1, \ldots)$

- First State (s_0) drawn randomly from initial state distribution, denoted as ρ_0:

$$s_0 \sim \rho_0(\cdot)$$

Transition Dynamics:

- Deterministic: st+1=f(st, at)
- Stochastic: st+1~P(·|st, at)

Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Notions on Reinforcement Learning

Reward and Return:

- Reward function r_t:

- Measure of the goodness of the performed action
- Depends on Present State, Recent Action, and/or Following State
- Common Simplification: Current State or State-Action Pair

- Return R:

- Simple Summation/Accumulation of Rewards In a Defined/ Infinite Timeline
- Finite-horizon undiscounted return: $$R(\tau) = \sum_{t=0}^{T} r_t.$$
- Infinite-horizon discounted return $$R(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t.$$

Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Notions on Reinforcement Learning

RL optimization:

- Identifying an Optimal Policy to Maximize Anticipated Return
- In the case of a stochastic environment and policy, the probability of a T -step trajectory is:

$$P(\tau|\pi) = \rho_0(s_0) \prod_{t=0}^{T-1} P(s_{t+1}|s_t, a_t)\pi(a_t|s_t).$$

- The expected return is then: $J(\pi) = \int_\tau P(\tau|\pi)R(\tau) = \mathop{\mathrm{E}}_{\tau\sim\pi}[R(\tau)].$

- The central optimization problem in RL can then be expressed by:

$$\pi^* = \arg\max_\pi J(\pi),$$

Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Notions on Reinforcement Learning

Formal Definition:

- Markov Decision Process which is a 5-Tuple $\langle S, A, R, P, \rho_0 \rangle$

- S: the set of all valid states
- A: the set of all valid actions
- R: Reward Function mapping State × Action × State → Real numbers
- P: Transition Probability Function associating State × Action with Probability Distribution over Successor State (P(s'|s,a))
- ρ_0: Starting State Distribution

The System complies with **Markov Property** implying dependency strictly on latest State and Action discarding past History

Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Imitation Learning

- Creation of advanced agents with lesser efforts by mimicking human behaviors.

- Excels for challenging tasks requiring complex control strategies or obscure objectives.

Two Classes of Methods:

- Supervised Learning-Based Methods - e.g., Behavioral Cloning

- Reinforcement Learning-Based Methods

Imitation Learning

Supervised Learning-Based Methods - e.g., Behavioral Cloning

- Utilizes demonstration data as direct supervision to train policies
- Reduces imitation learning issue to conventional supervised learning problem
- Effective when ample data is available and recording actions is viable
- Limited for motor control tasks, as accurately logging human actions and tackling embodiment discrepancies pose difficulties

Imitation Learning

Reinforcement Learning-Based Methods

- Uses demonstrations to shape an objective function measuring conformance to the demonstrator's behavior

- Optimization-based methods (like RL or trajectory optimization) derive a controller by improving said objective

- Can define objective function independent of demonstrator's actions, making it adaptable to cases where actions aren't readily accessible

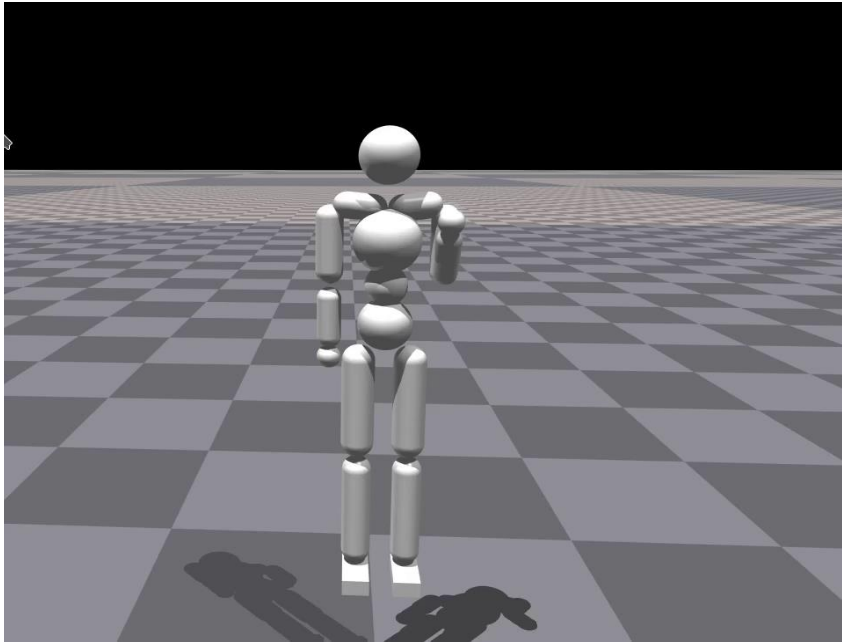
- Capable of abstracting differences in embodiment between the demonstrator and the agent

- More data-efficient than Supervised methods, potentially learning complex skills with merely one demonstration

Imitation Learning and Physics-Based Character Animation

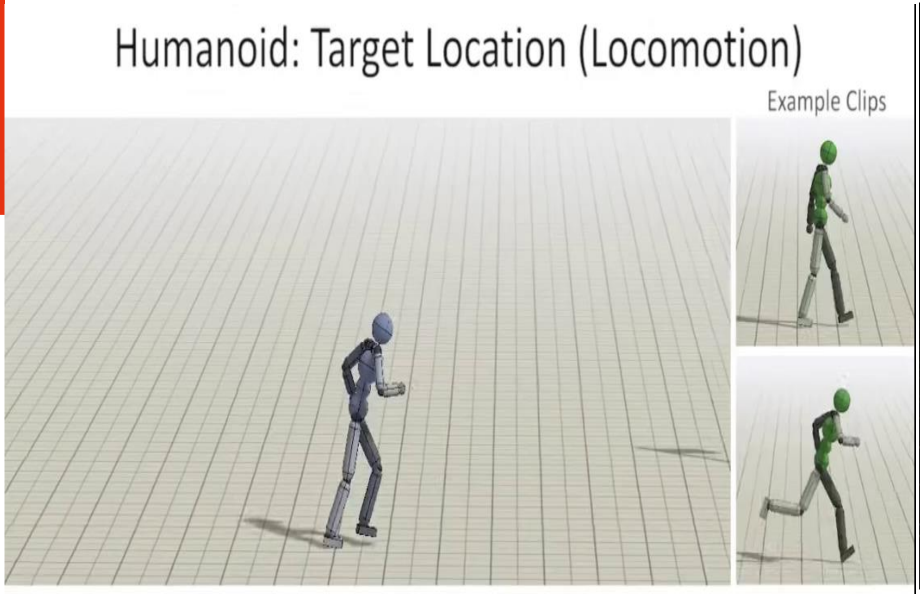
- Agent gets observation:
    - Current humanoid's configuration.
- Decides an action to perform:
    - Torques applied to joints
- The environment transitions to a new state.
- Reward: How similar the generated motion is to the reference motion capture data.

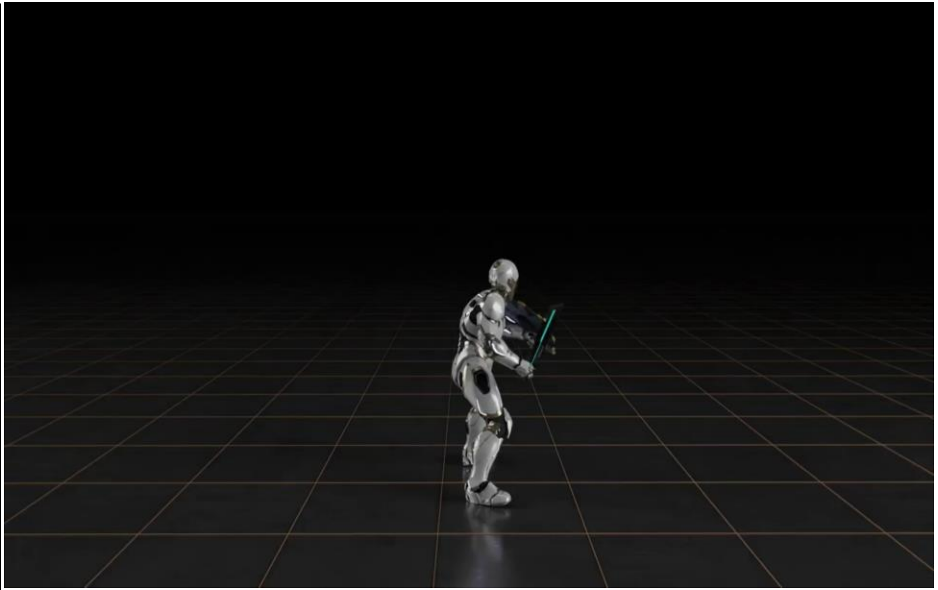
Goal: Maximize its cumulative reward



Recent Work

Controllable and Realistic Character motion





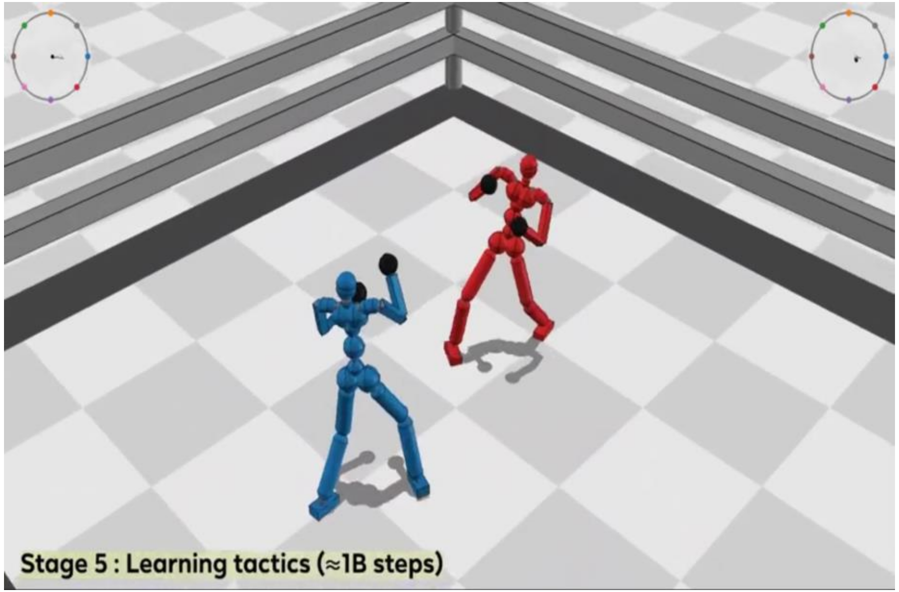
Peng et Al, Adversarial Motion Priors for stylized physics-based character control 2021

NVIDIA: AI-Driven, Physics-Based Character Animation https://youtu.be/8oIQy6fxfCA

Recent Work

Interaction between multiple characters / Scene Objects

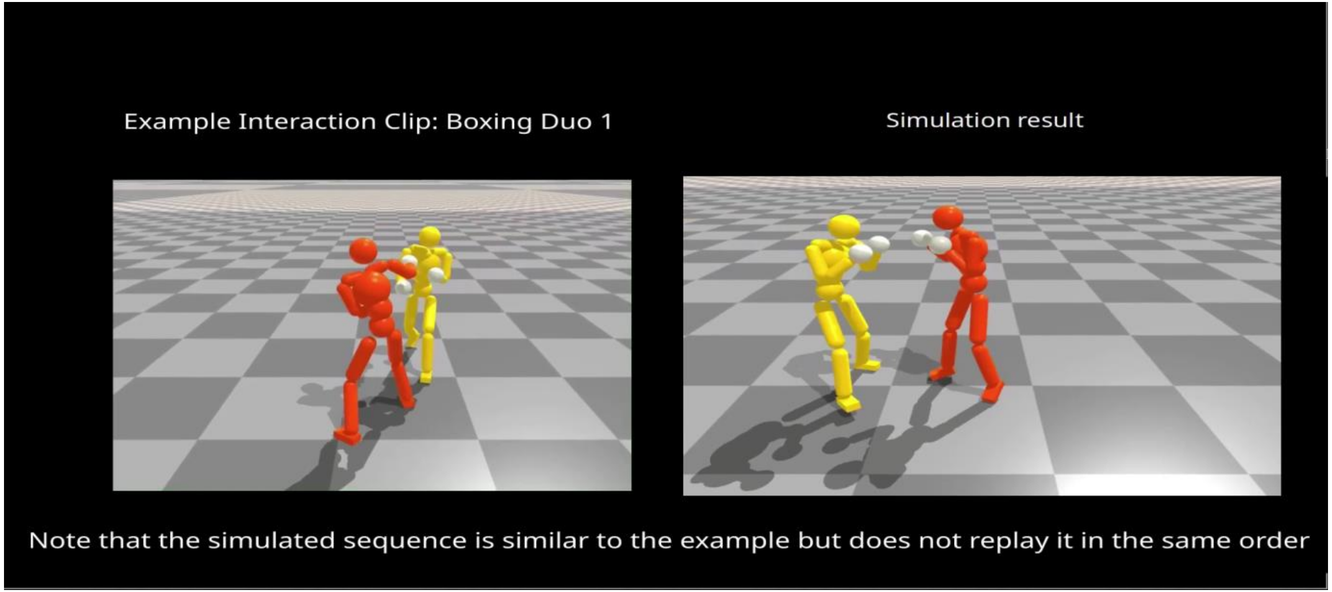
Hassan et Al, Synthesizing Physical Character-Scene Interactions 2023



Won et Al, Control strategies for physically simulated characters performing two-player competitive sports 2021

Example: Physics Simulation for Fighting Imitation

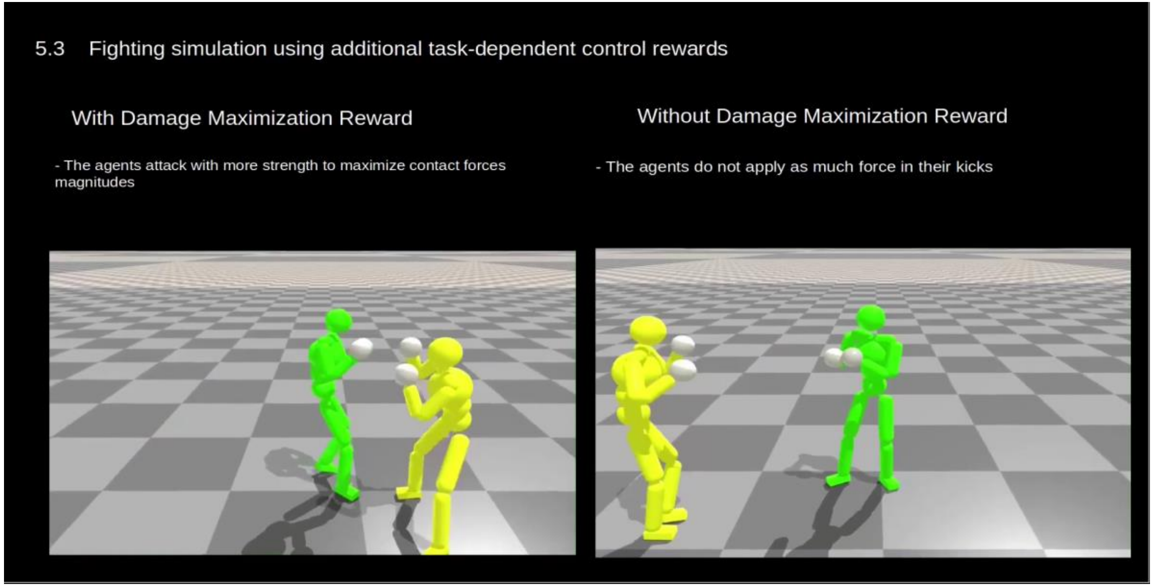
Imitation of demonstration data



Younes et Al, Multi-Agent Adversarial Interaction Priors for imitation from fighting demonstrations for physics-based characters 2023

Example: Physics Simulation for Fighting Imitation

Adding constraint to the imitation



Younes et Al, Multi-Agent Adversarial Interaction Priors for imitation from fighting demonstrations for physics-based characters 2023

Thank you for your attention!