Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

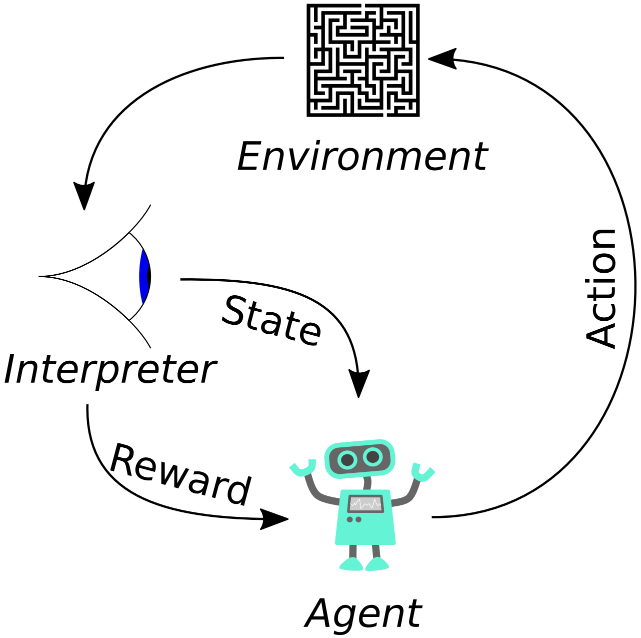
Timon Engelke

Learning to Kick from Demonstration with Deep Reinforcement Learning

Motivation

- Kick is required in Humanoid Soccer League
- Current approaches:
  - Keyframe Animations
  - Kick Engines
- New approach:
  - Learning from Demonstration to improve existing solutions

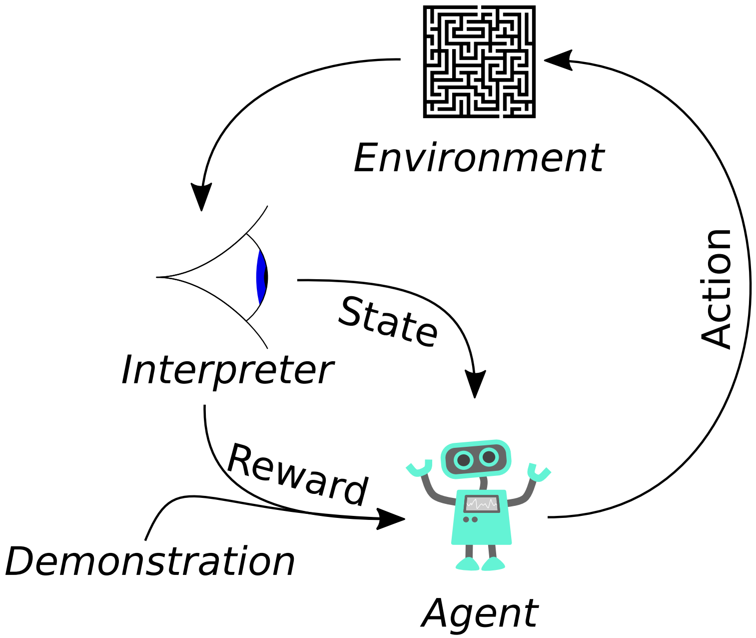
Reinforcement Learning



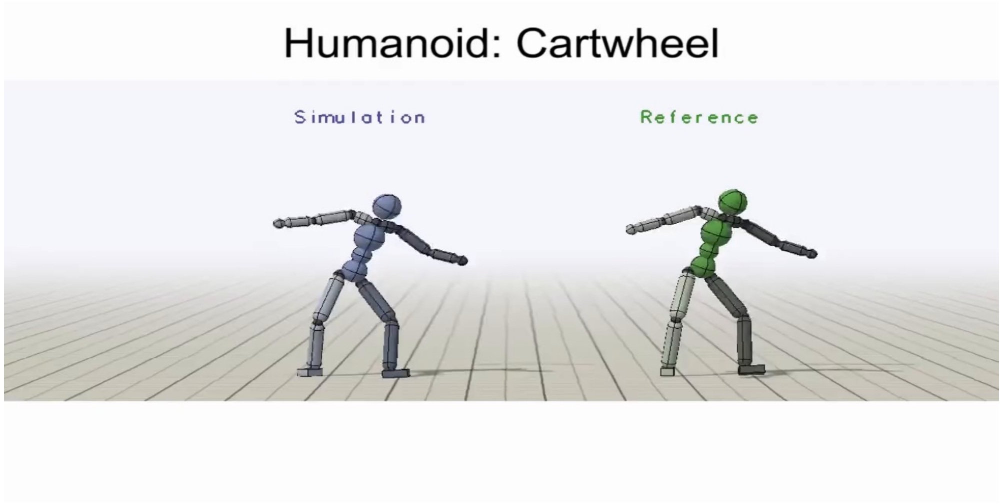
Source: Wikimedia Commons, CC-0

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Learning from Demonstration



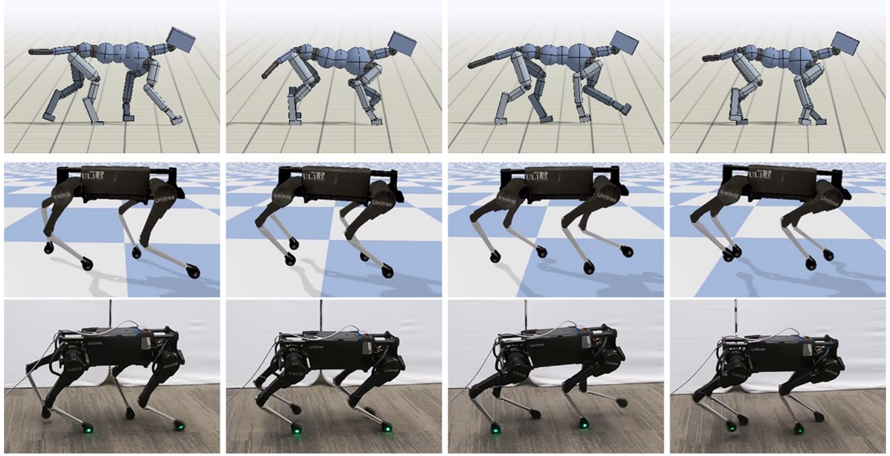
DeepMimic (Peng et al., 2018)



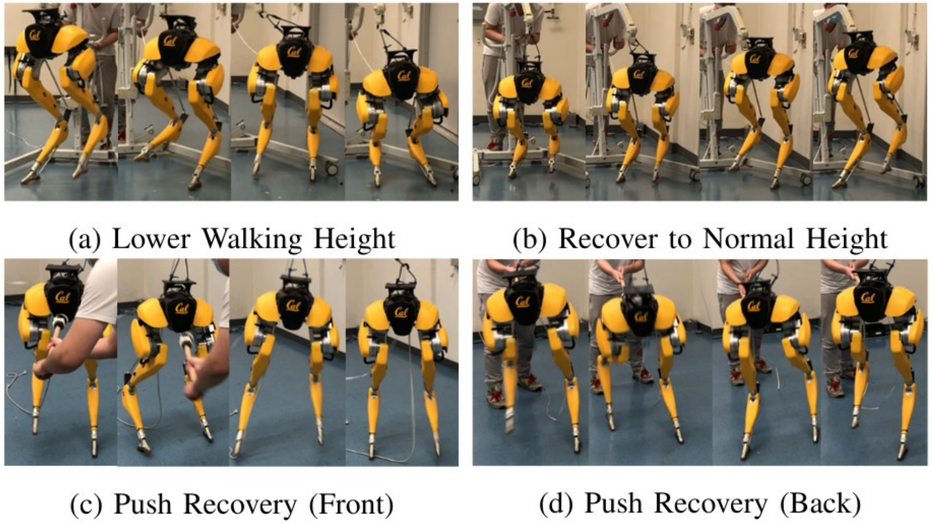
Learning from Demonstration for various motions

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Learning Agile Robotic Locomotion Skills by Imitating Animals (2020)

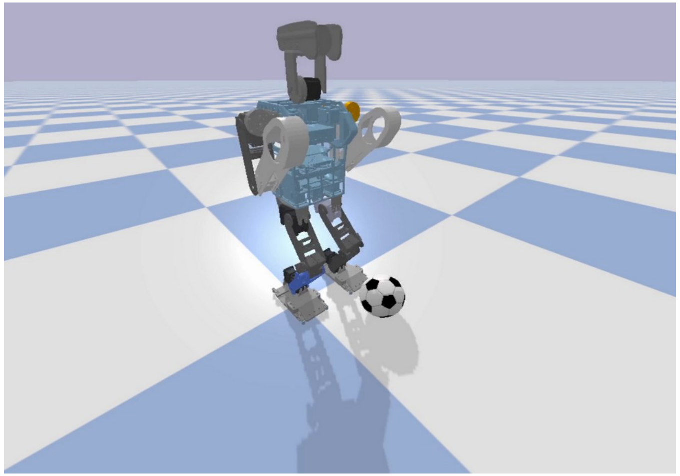


Reinforcement Learning for Robust Parameterized Locomotion Control of Bipedal Robots (2021)



(a) Lower Walking Height
(b) Recover to Normal Height
(c) Push Recovery (Front)
(d) Push Recovery (Back)

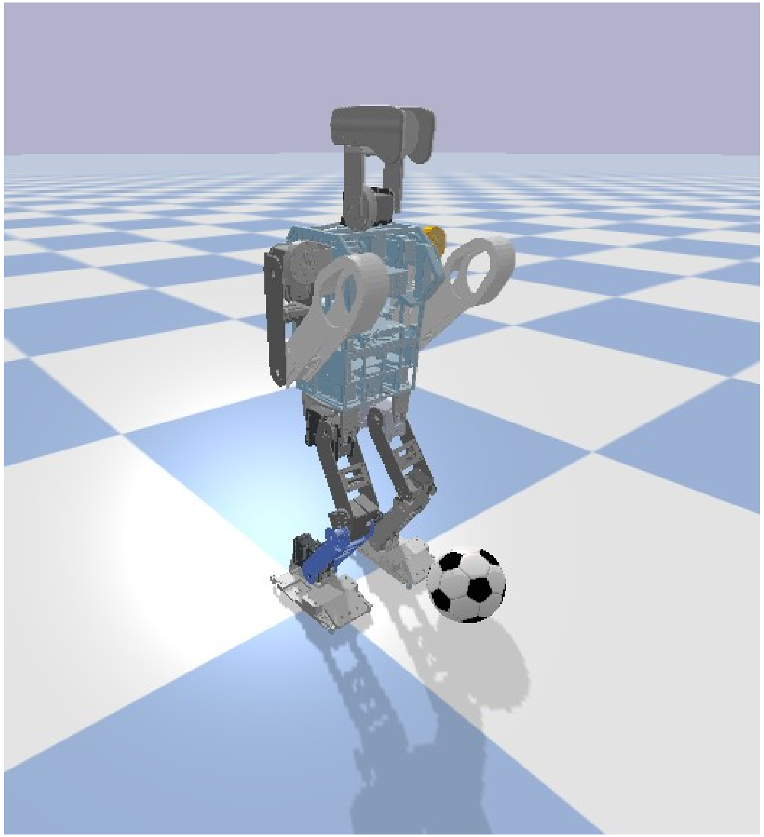
Demonstration used in the training



- Kick Engine currently used by Bit-Bots
- Only one motion is used
- Parameters were optimized for most effective and reliable results
- Multi-objective tree-parzen estimator was used for optimization

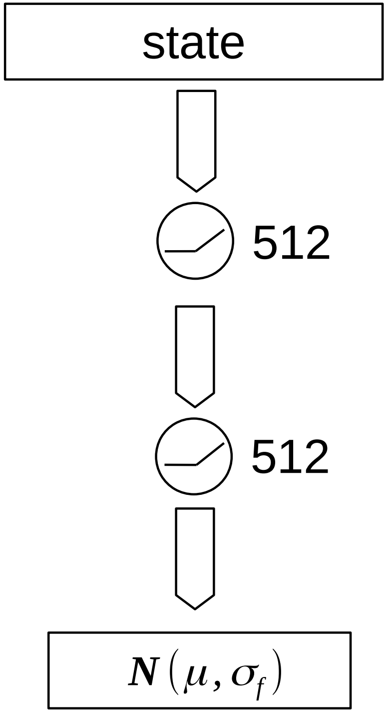
Training Setup

- Environment:
  - PyBullet Simulator
  - Wolfgang Robot and FIFA Size 1 Ball
- Training:
  - Stable Baselines 3
  - Proximal Policy Optimization
  - 30 Hz
  - 10 million timesteps (~150.000 episodes)



Network Architecture

- Separate Networks for Policy and Value function
- Two fully connected hidden layers with 512 neurons
- ReLU activation function
- Gaussian distribution with fixed variance in output layer
- Normalized input and output

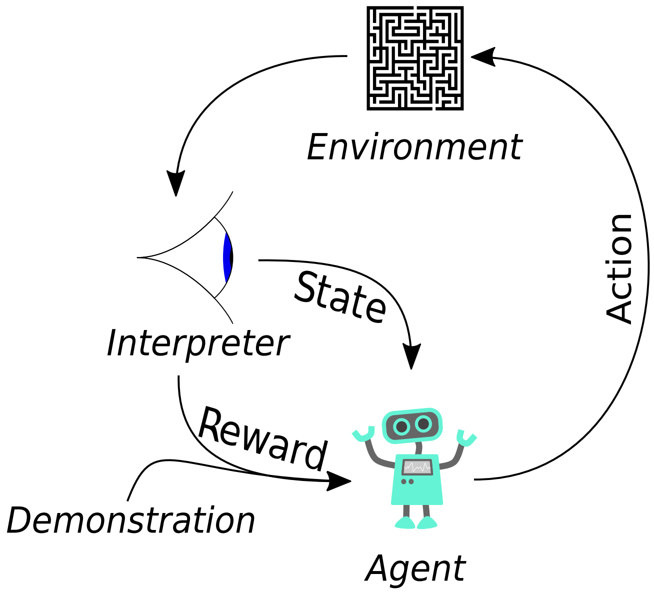


Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Reward

$$r = 0.7\, r_I + 0.3\, r_T$$

- Imitation Reward
  - Root position
  - End effector positions
  - Joint positions
  - Joint velocities
- Task Reward
  - Ball velocity



Imitation Reward

- Rewards closeness to the demonstration
- Same as in DeepMimic

$$r_I = 0.1\, r_R + 0.15\, r_E + 0.65\, r_P + 0.1\, r_V$$

Root position

$$r_R = \exp\left(-10 \cdot \|R - \hat{R}\|_2^2\right)$$

End effector positions

$$r_E = \exp\left(-40 \sum_{e \in E} \|p_e - \hat{p}_e\|_2^2\right)$$

Joint positions

$$r_P = \exp\left(-2 \sum_{j \in J} \|p_j - \hat{p}_j\|_2^2\right)$$

Joint velocities

$$r_V = \exp\left(-0.1 \sum_{j \in J} \|v_j - \hat{v}_j\|_2^2\right)$$

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Task Reward

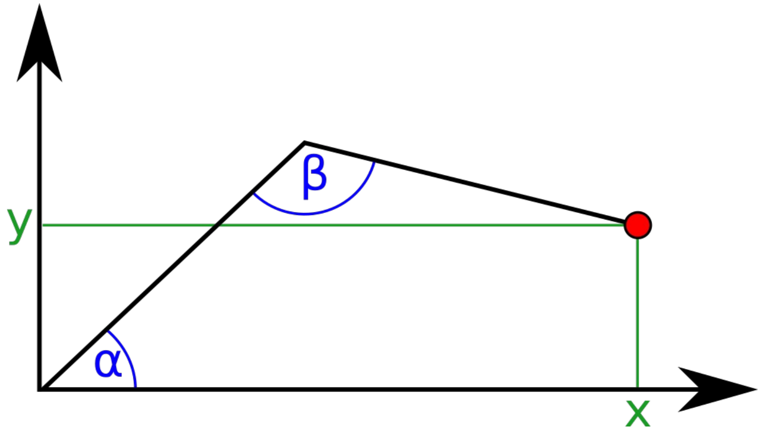
- Rewards strong ball movement at the correct time

Ball Velocity

$$r_T = \begin{cases} 1 - \exp(-2 \cdot v_B) & if\ t_k \leq t \leq (t_k + 0.5) \\ 0 & else \end{cases}$$

Actions

- Ways of controlling the robot’s legs
- Two different types:
  - Joint action (motor goals)
  - Cartesian action (foot positions)



States

- Representation of the robot
- Available information:
  - Phase: increasing number marking the progress in the kick
  - Proprioception: current position and velocity of feet
  - IMU readings: roll, pitch, and angular velocities
  - Pressure sensor readings

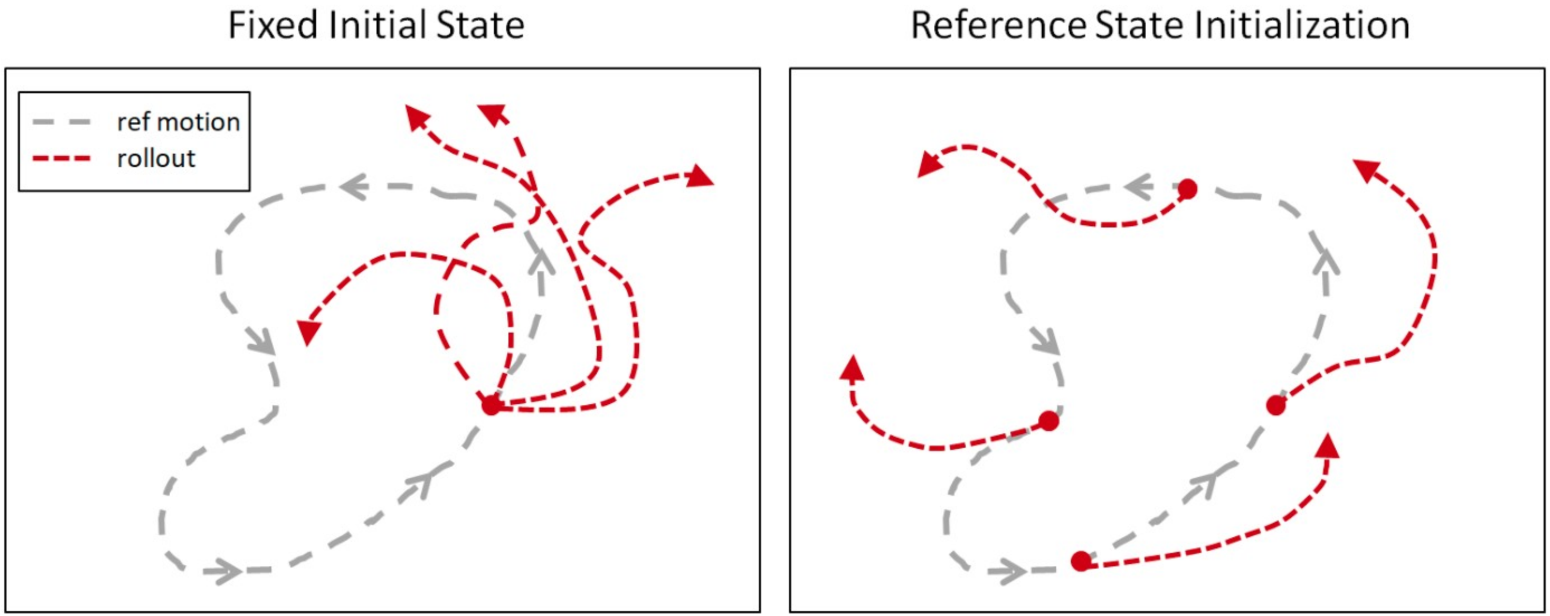
States

	Phase	Orient.	Ang. Vel.	Foot Pos.	Foot Vel.	Pressures
PhaseState	✓	–	–	–	–	–
OrientationState	✓	✓	–	–	–	–
GyroState	✓	✓	✓	–	–	–
FootState	✓	–	–	✓	–	–
FootVelocityState	✓	–	–	✓	✓	–
OrientationFootState	✓	✓	–	✓	–	–
PressureState	✓	–	–	–	–	✓
PressureFootState	✓	–	–	✓	–	✓
ComprehensiveState	✓	✓	✓	✓	✓	✓

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Training Additions

- Early Termination
  - Reset the robot when it falls
- Reference State Initialization
  - Start the robot at random positions of the demonstration
- Initial Bias
  - Set the bias of the output layer to obtain a stable position



Source: DeepMimic, Peng et al., 2018

Training Additions

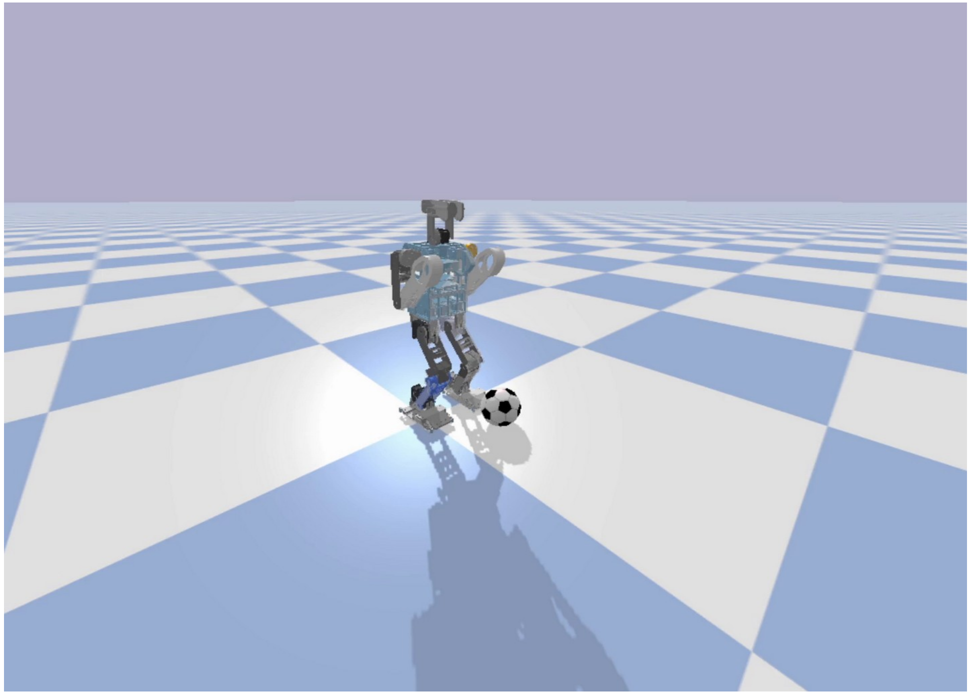
- Early Termination
  - Reset the robot when it falls
- Reference State Initialization
  - Start the robot at random positions of the demonstration
- Initial Bias
  - Set the bias of the output layer to obtain a stable position

Experiments

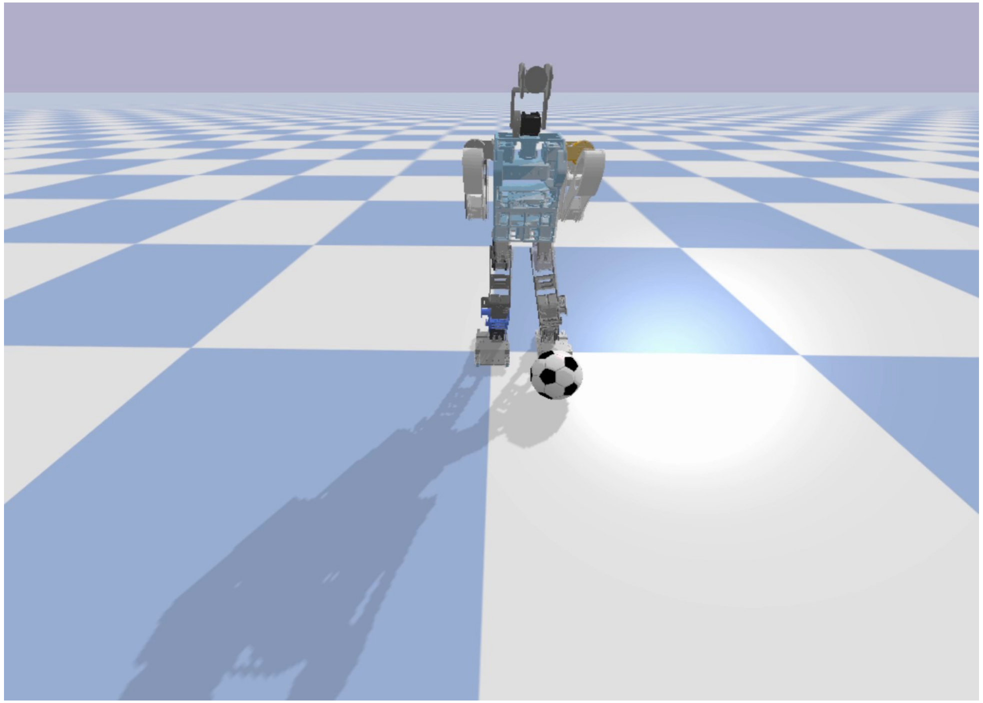
- Ablation Study on states
  - Which parts of the input improve the kick?
  - Which parts worsen it?
- Differences in Cartesian / Joint actions
  - Does the action representation improve the kick?
  - Does it influence the sample efficiency?

➔ 18 different training runs

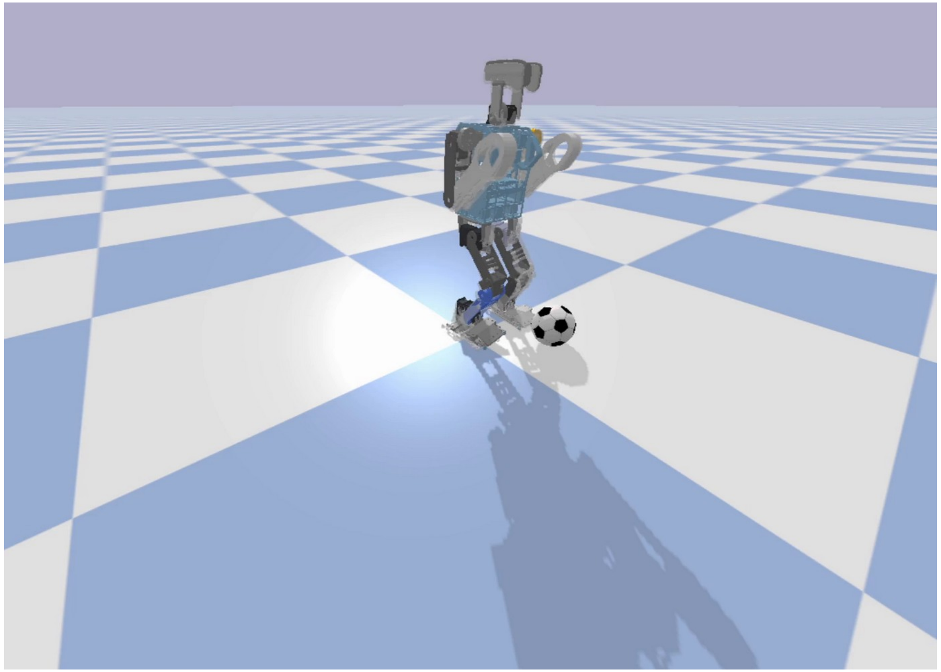
- Resistance against pushing



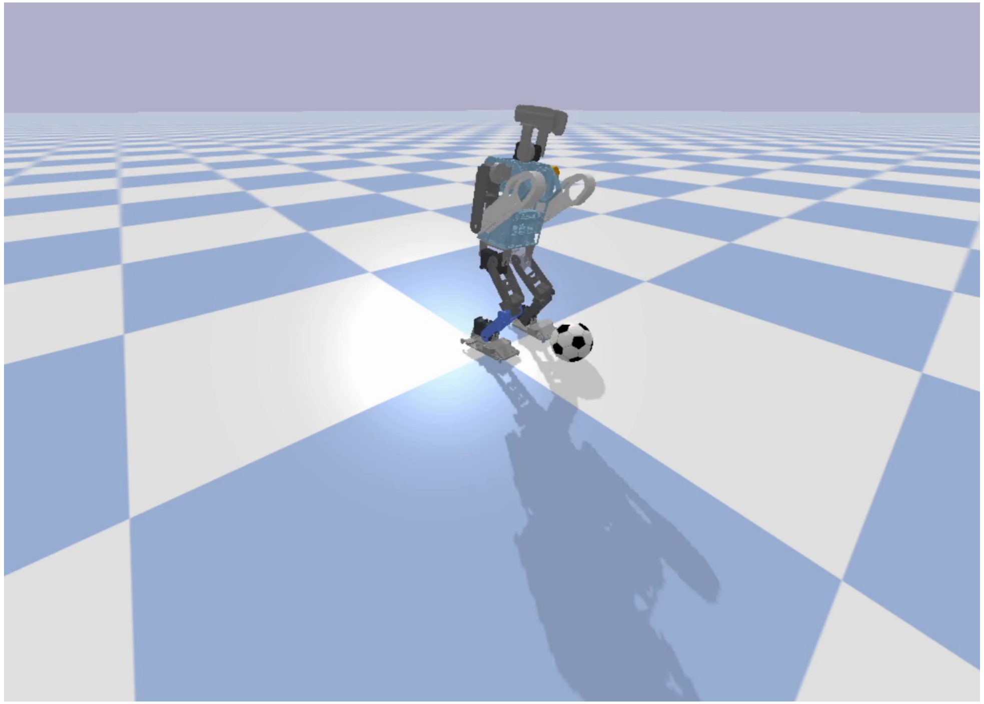
PhaseState with JointAction



PhaseState with JointAction



OrientationState with CartesianAction



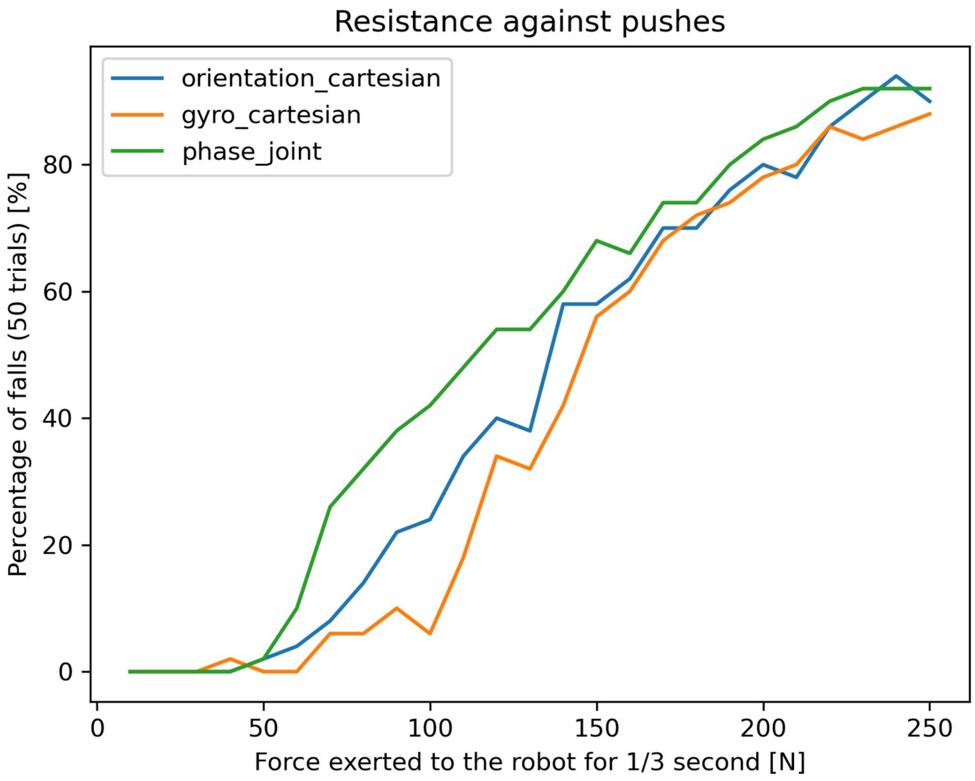
PressureFootState with CartesianAction

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Results

State	Action	Time to Stability	Distance	Fallen	Timesteps ($r = 35$)	Visual
PhaseState	Cartesian	**2.6**	0.246	40%	2 210 000	✓
PhaseState	Joint	**1.4**	**0.358**	**0%**	1 050 000	✓
OrientationState	Cartesian	**2.127**	**0.331**	**0%**	790 000	✓
OrientationState	Joint	3.0	0.199	**0%**	930 000	○
GyroState	Cartesian	3.0	**0.303**	**0%**	2 000 000	✓
GyroState	Joint	3.0	0.199	**0%**	–	○
FootState	Cartesian	3.0	0.245	70%	–	○
FootState	Joint	–	0.213	100%	–	✗
FootVelocityState	Cartesian	–	0.006	100%	2 810 000	✗
FootVelocityState	Joint	3.0	0.03	10%	–	✗
OrientationFootState	Cartesian	2.997	0.161	**0%**	870 000	○
OrientationFootState	Joint	3.0	0.225	**0%**	–	✗
PressureState	Cartesian	–	0.289	100%	–	✗
PressureState	Joint	–	0.002	100%	–	✗
PressureFootState	Cartesian	3.0	0.181	40%	–	✗
PressureFootState	Joint	3.0	0.067	80%	–	✗
Comprehensive	Cartesian	2.99	0.231	**0%**	1 830 000	✓
Comprehensive	Joint	3.0	0.062	**0%**	–	✗
Demonstration		1.433	0.327	0%		

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG



Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Results

- Best results: PhaseState, OrientationState, GyroState
  - Stable kick
  - Low number of timesteps (< 3M)
  - Kick distance higher than demonstration
- Pressure sensors or foot velocities lead to unstable results
- Cartesian action might lead to better results

Discussion

- Open-Loop Approach (PhaseState) performs best
  - Simulation is mostly deterministic
  - Performance is likely to be worse on real robot
- Pressures and Foot Velocities worsen the result
  - Relatively noisy or jumpy inputs
  - Might disturb gradient updates

Possible problems

- Hyperparameters are not optimized for each problem
- Network architecture might not be adequate
- Task reward function can probably be improved

Future Work

- Tweak reward function
- Hyperparameter optimization
- Sim-to-real transfer
- Hierarchical approach for different kick directions

Questions?