Hierarchical Control for Bipedal Locomotion using Central Pattern Generators and Neural Networks

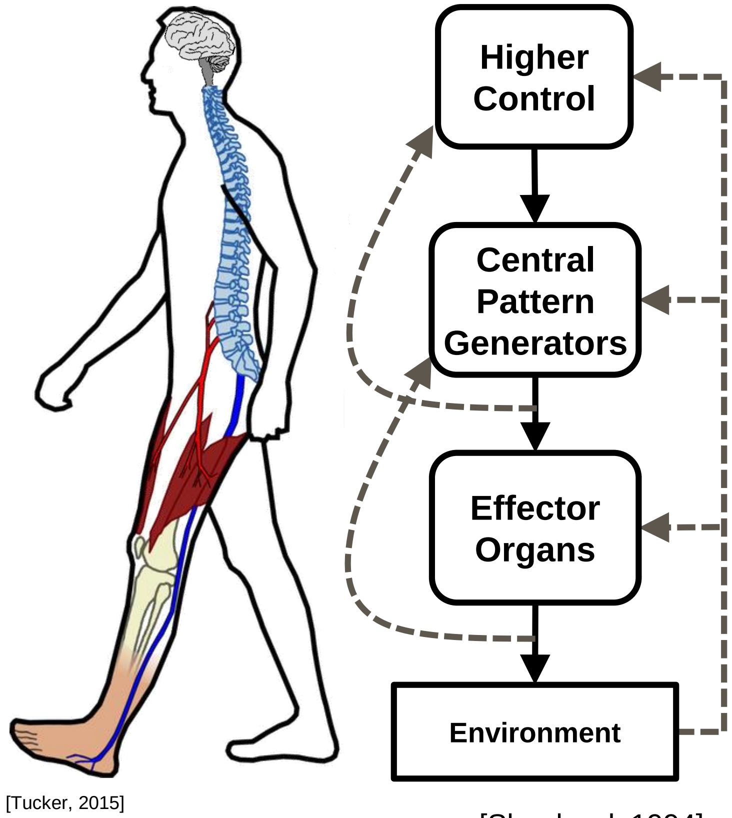
Sayantan Auddy

Adviser: Dr. Sven Magg

http://www.informatik.uni-hamburg.de/WTM/

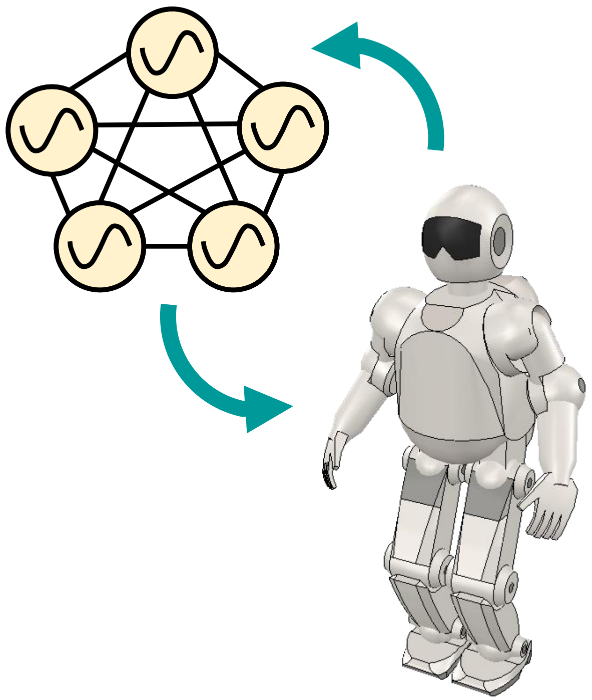
Motivation

Biological Inspiration

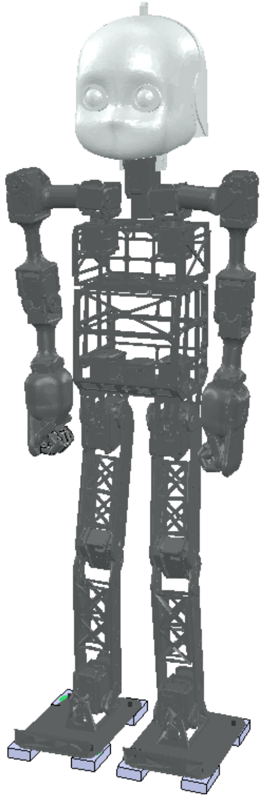


[Shepherd, 1994]

Central Pattern Generator based approaches



Walking algorithm for the NICO robot



Outline

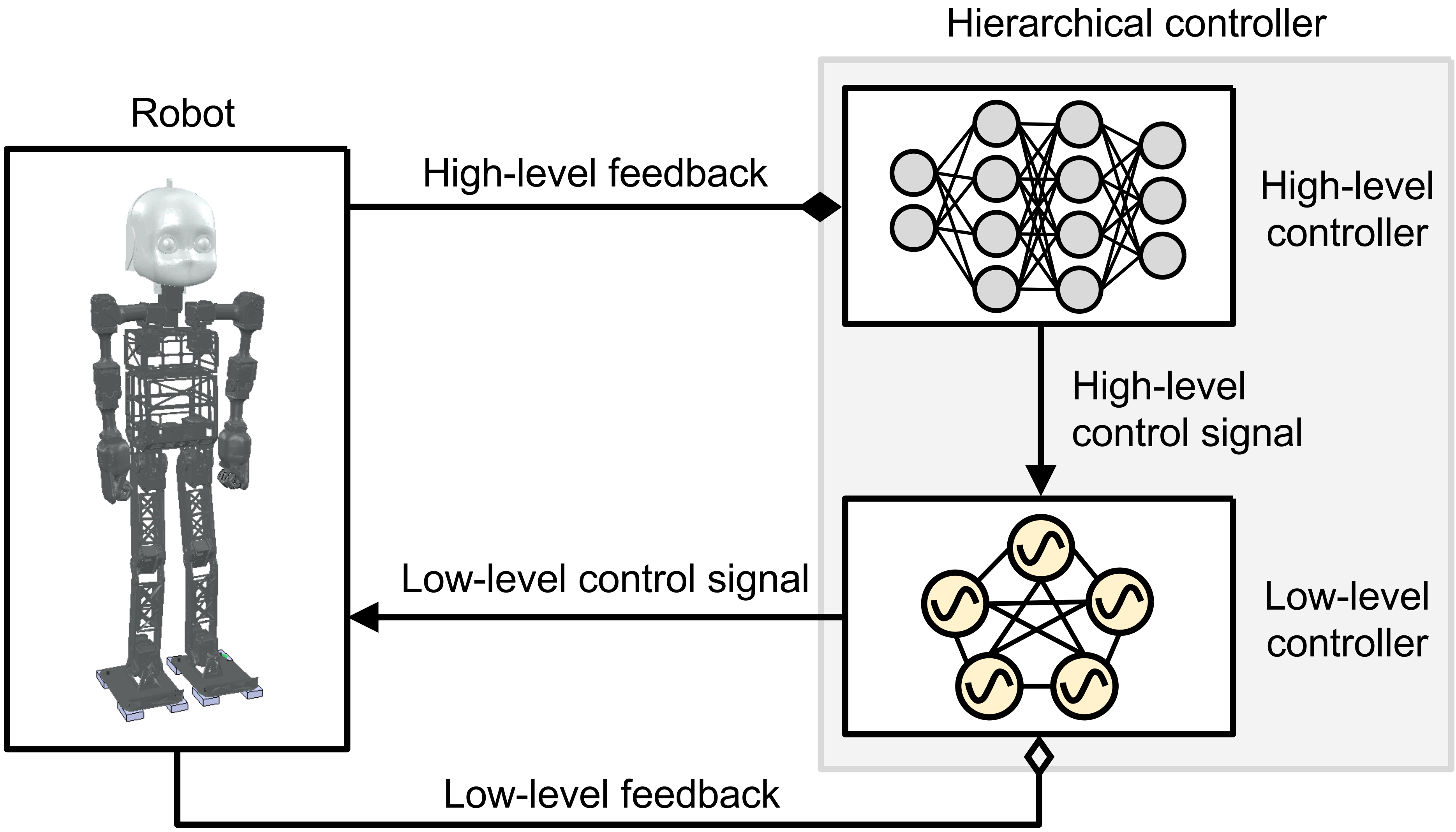
- Motivation
- Research Questions
- Architecture
- Low-level Control Experiments
- High-level Control Experiments
- Conclusion and Future Work

Research Questions

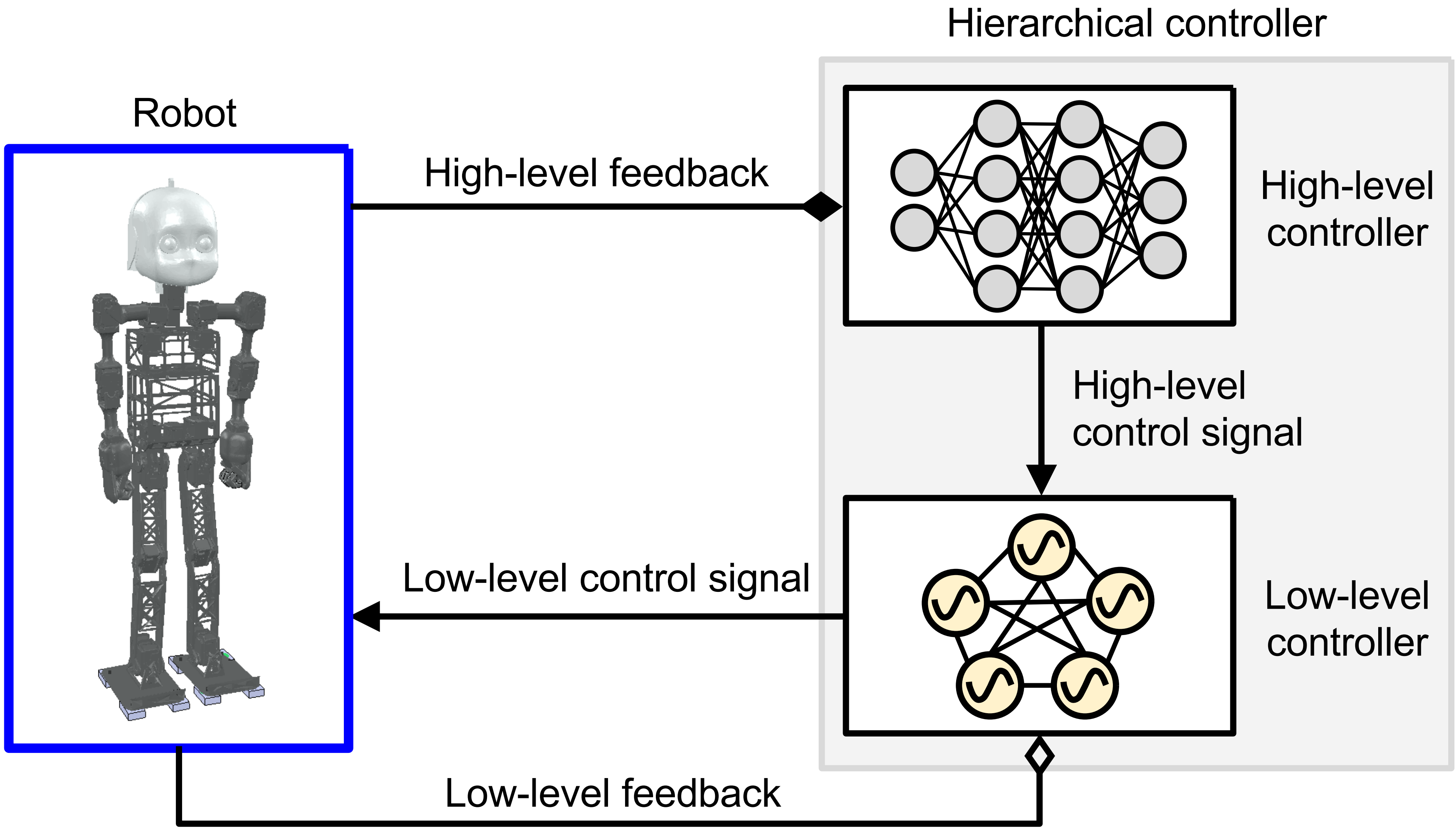
- Can bipedal locomotion be achieved by using a **low-level CPG-based controller** modulated by a **high-level neural network controller**, and is such a control mechanism beneficial for walking?

- Do **feedback mechanisms** for CPGs **improve the gait**? How do different feedback mechanisms **compare** against each other?

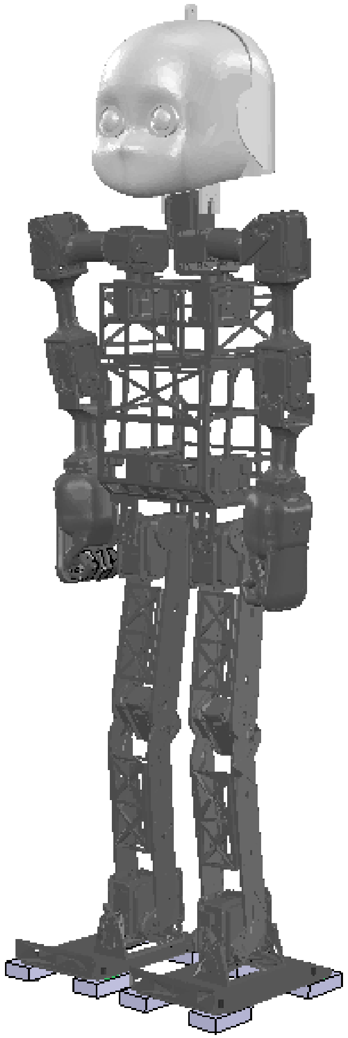
Architecture Overview



Architecture Overview

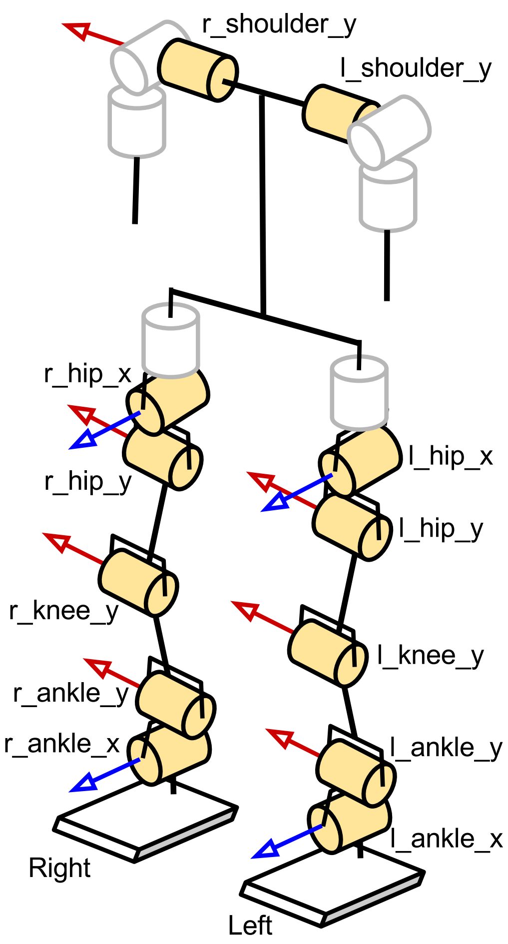


Robot

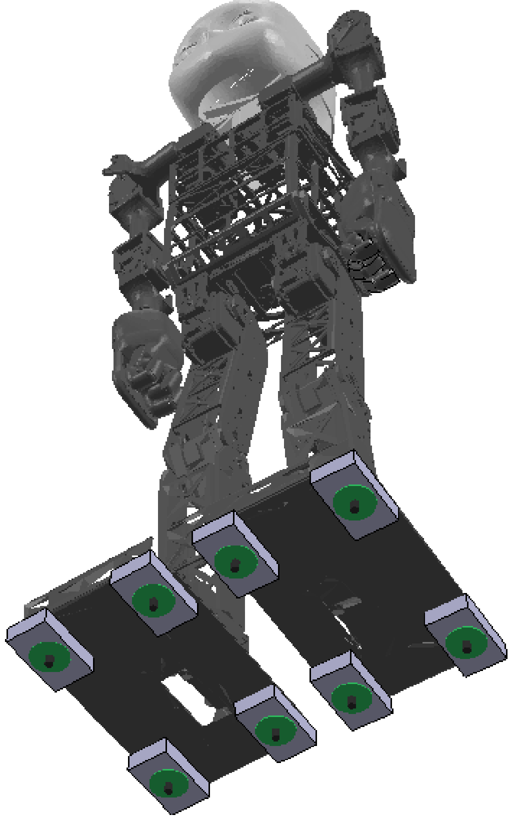


NICO robot used in simulation

[Kerzel, 2017]

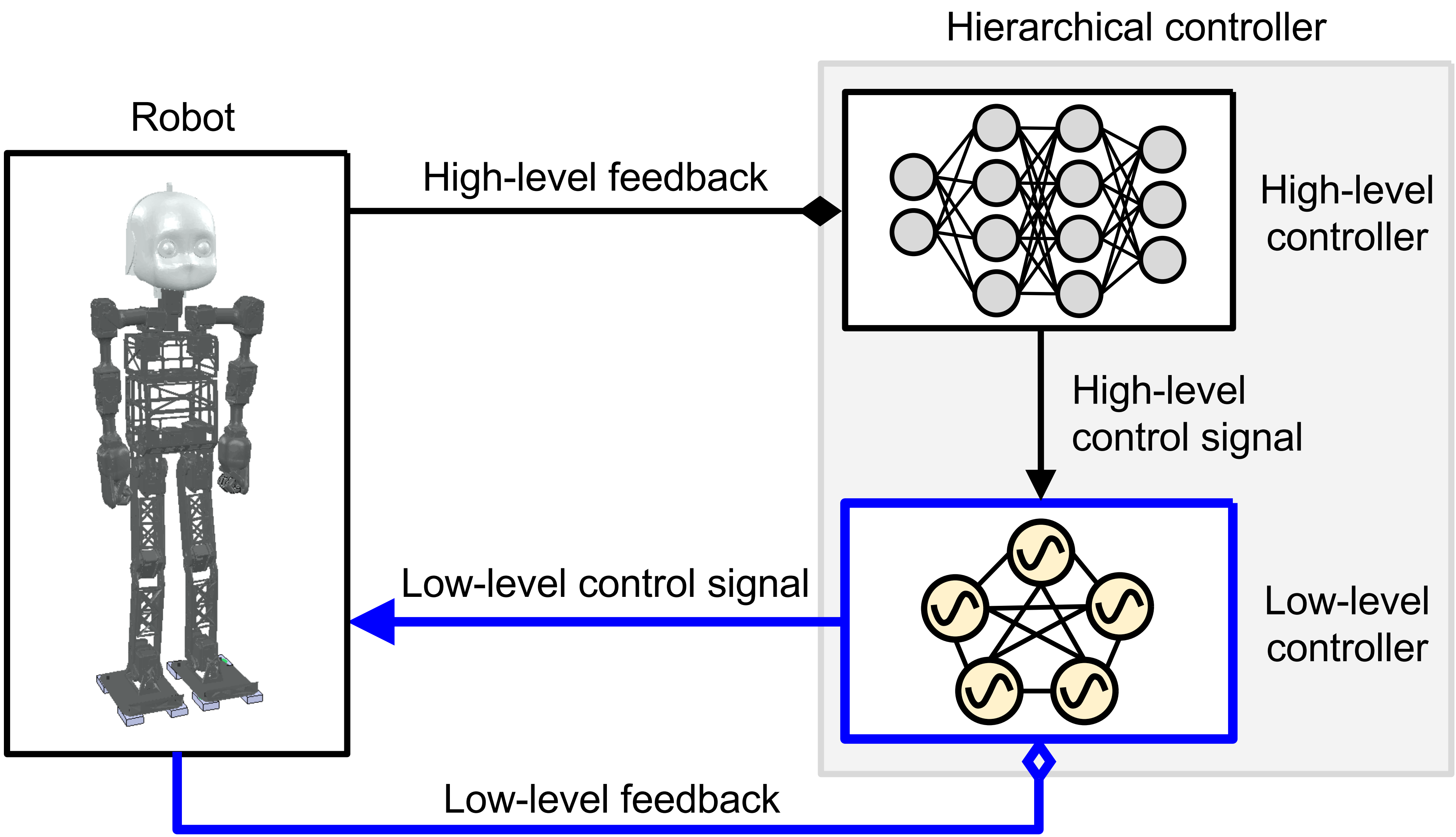


10 leg joints, 2 shoulder joints

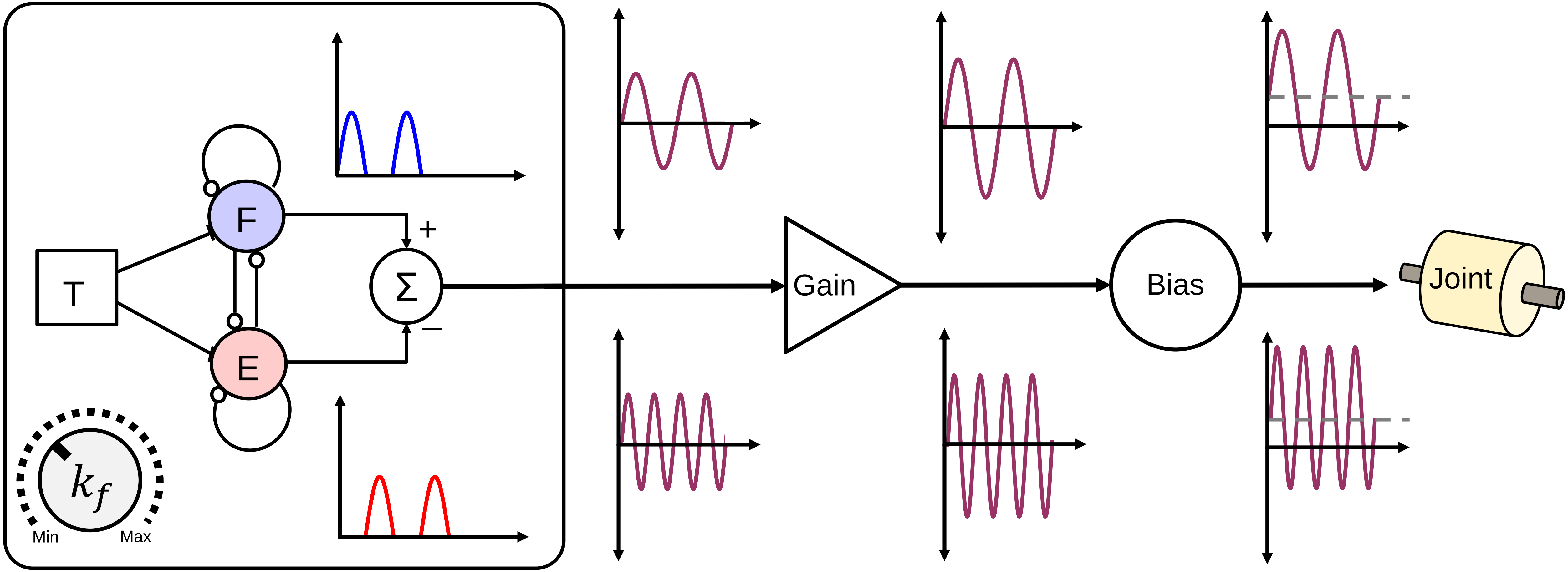


Force sensors added

Architecture Overview



Central Pattern Generator - Matsuoka Oscillator



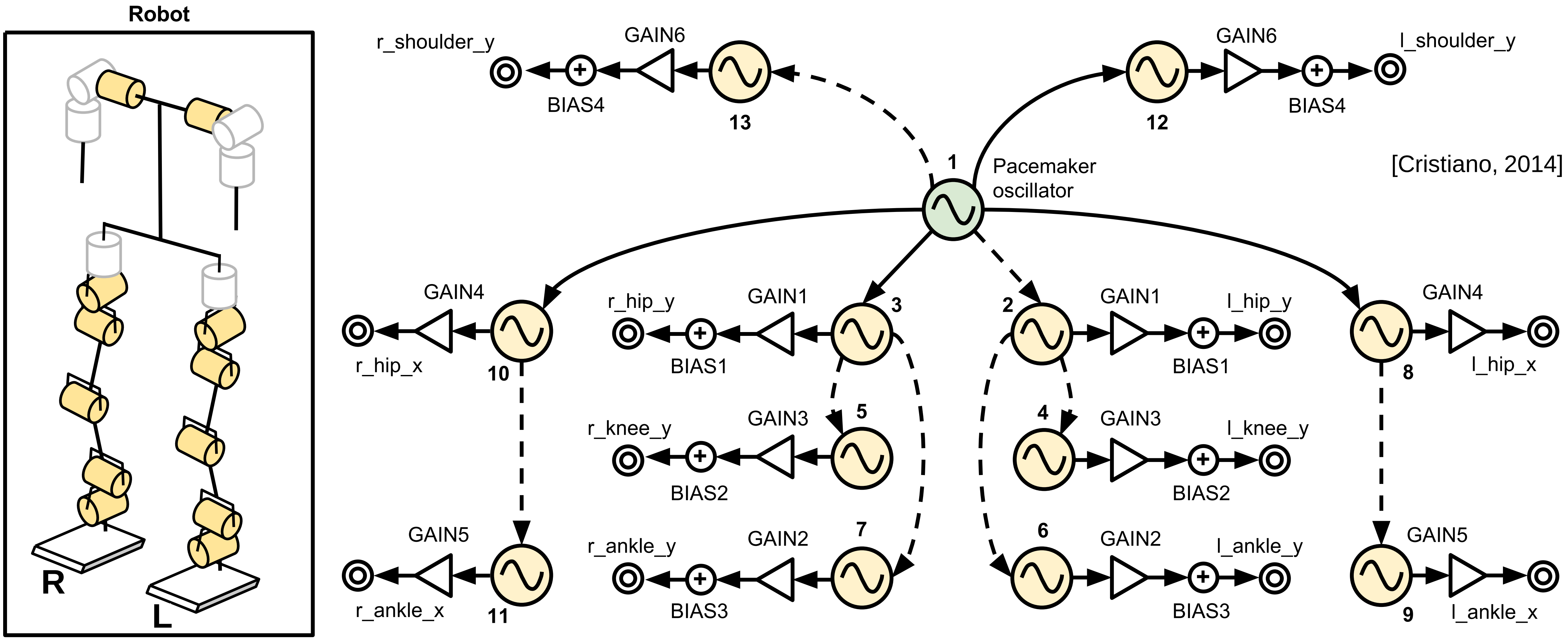
[Matsuoka, 1985, 1987]
[Kamimura, 2005]
[Cristiano, 2014]

Gain → Amplitude

Bias → Mean position

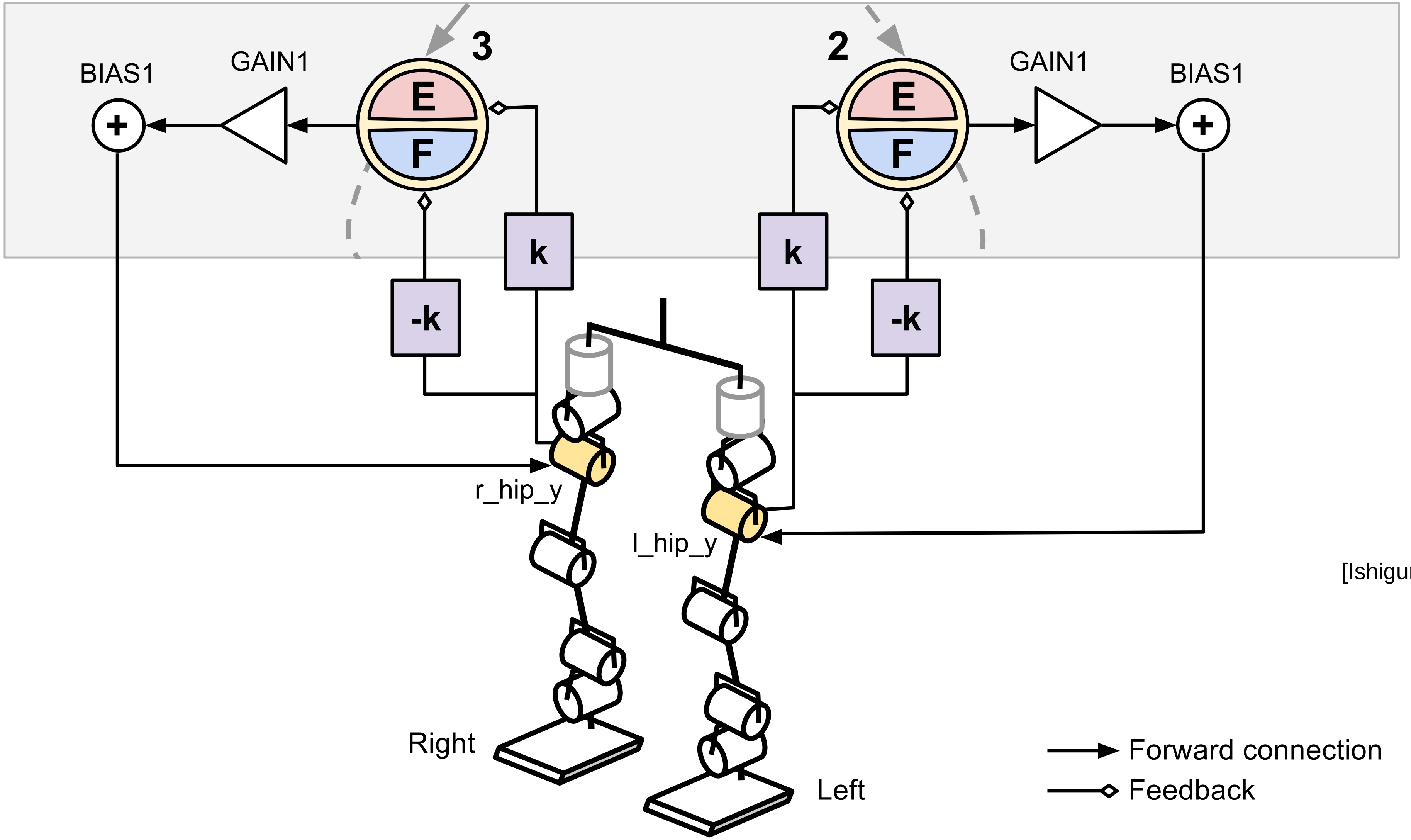
k_f → Frequency

Low-level Controller



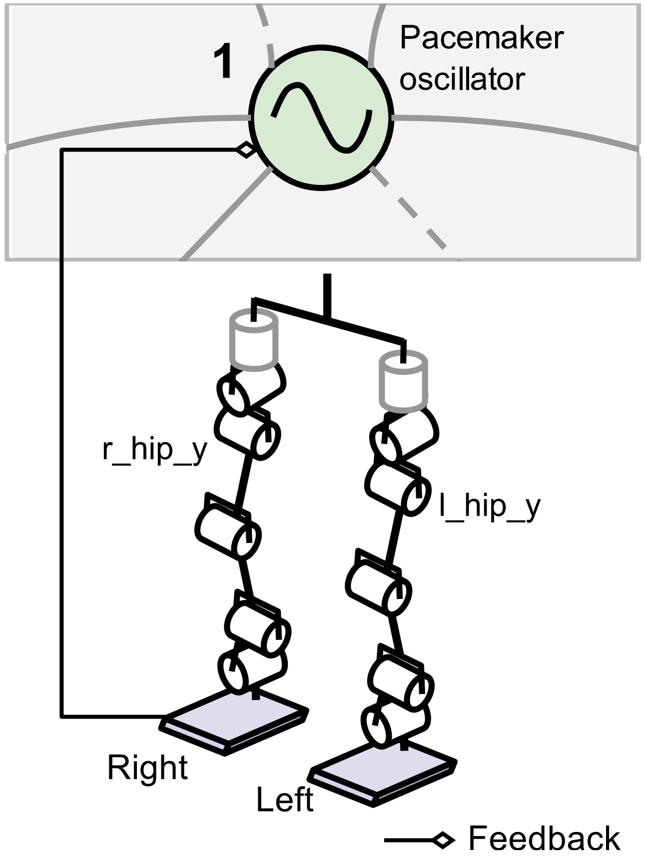
k_f	GAIN1	GAIN2	GAIN3	GAIN4	GAIN5	GAIN6	BIAS1	BIAS2	BIAS3	BIAS4

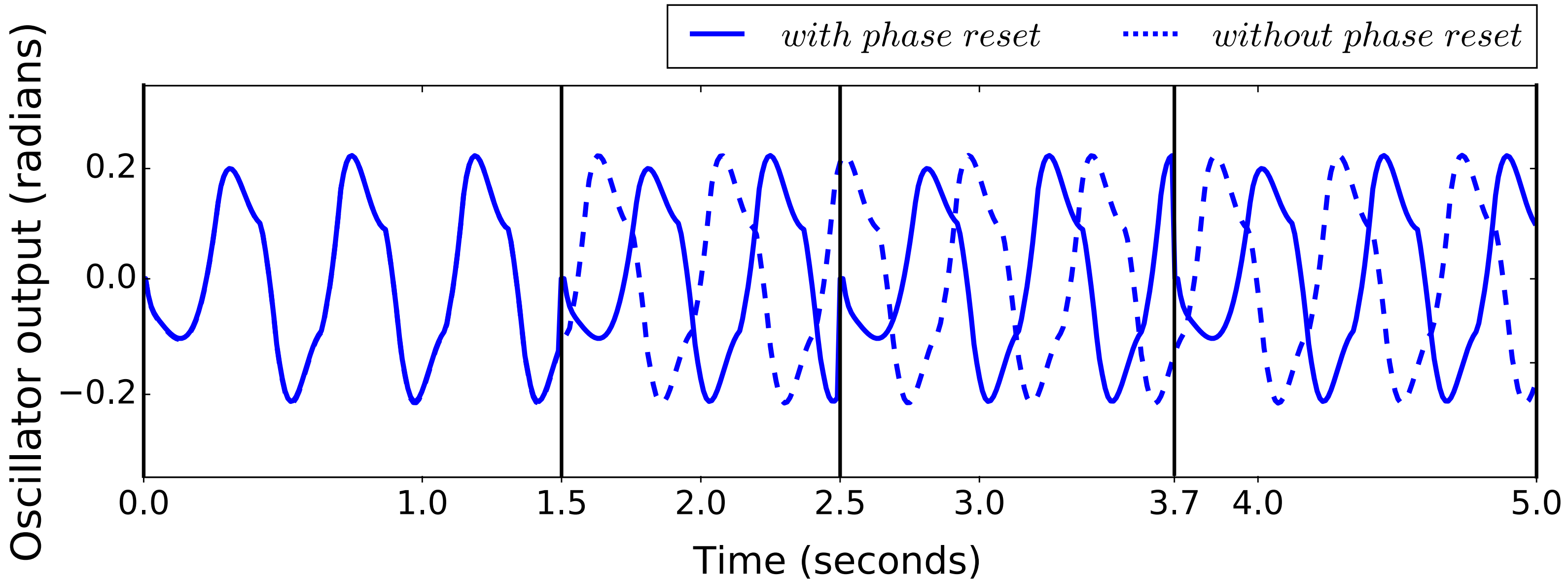
Low-level Feedback: Angle Feedback



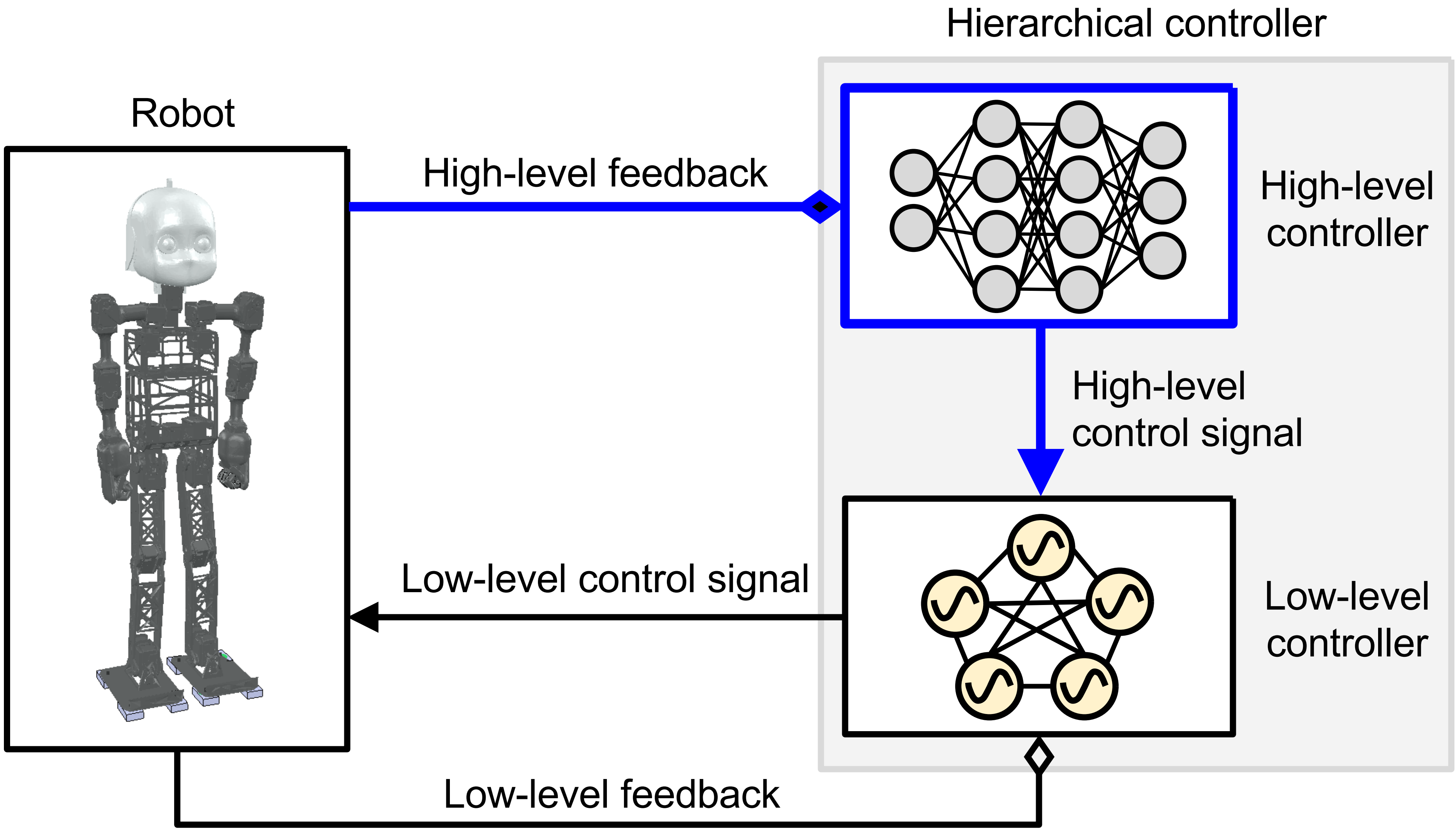
[Ishiguro, 2003]

Low-level Feedback: Phase Reset

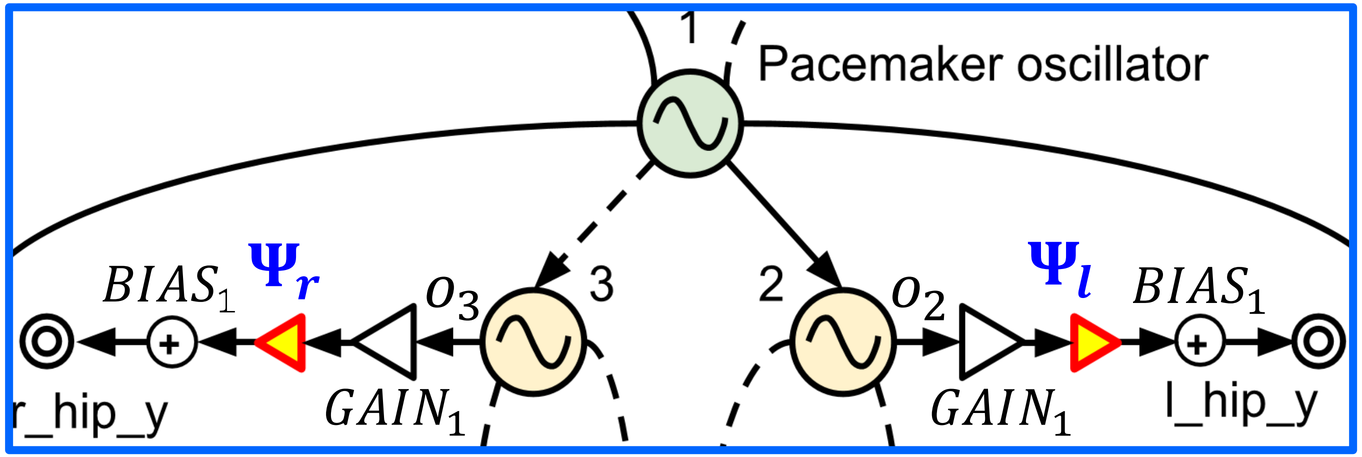


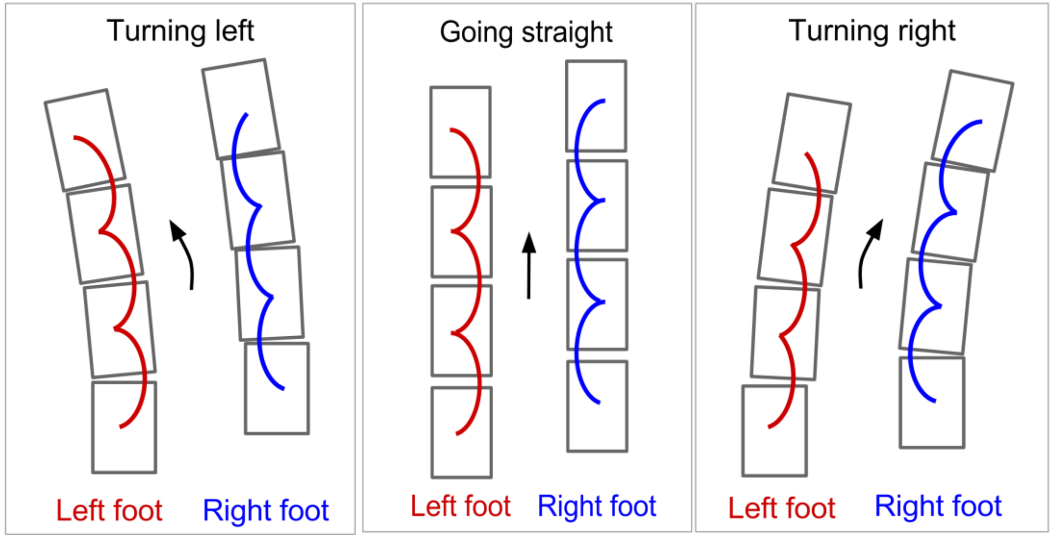


Architecture Overview



High-level Controller: Turning Mechanism



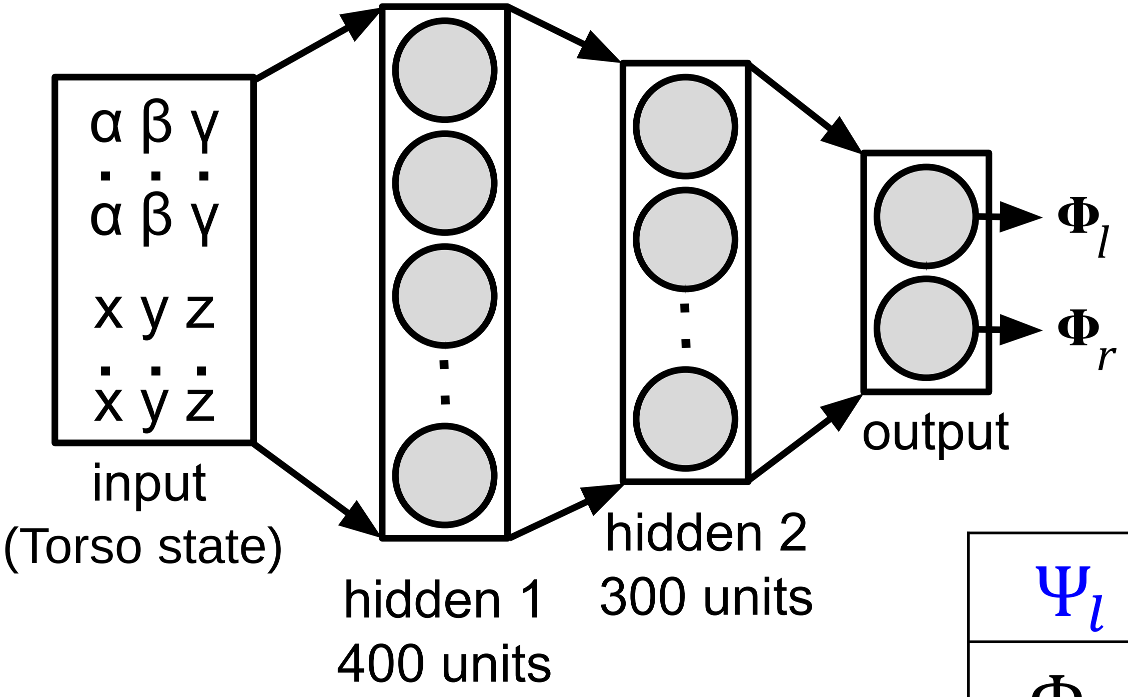


$\Psi_l > \Psi_r$ $\Psi_l = \Psi_r$ $\Psi_l < \Psi_r$

$$l\_hip\_y = o_2 \times \Psi_l \times GAIN_1 + BIAS_1$$

$$r\_hip\_y = o_3 \times \Psi_r \times GAIN_1 + BIAS_1$$

High-level Controller: Neural Network

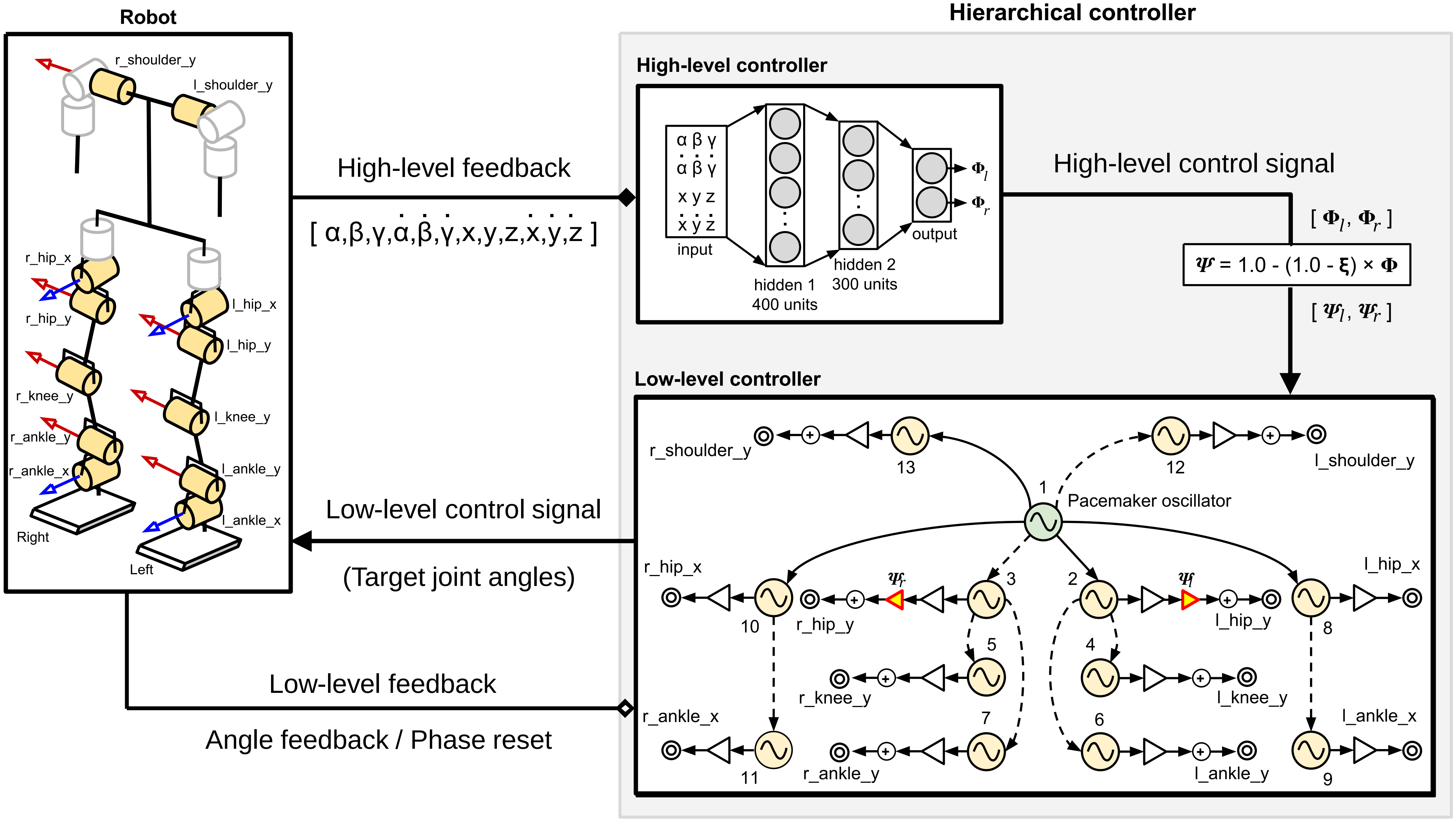


$$\Psi_{l|r} = 1.0 - (1.0 - \xi) \times \Phi_{l|r}$$

Ψ_l Ψ_r

Ψ_l Ψ_r	gain factors
Φ_l Φ_r	Neural network output
ξ	Controls influence of Neural Network 1.0 – No influence 0.0 – High influence

Architecture Details



Low-level Control – Experimental Setup

Chromosome Structure

Open Loop	k_f	GAIN1	GAIN2	GAIN3	GAIN4	GAIN5	GAIN6	BIAS1	BIAS2	BIAS3	BIAS4	
Angle Feedback	k_f	GAIN1	GAIN2	GAIN3	GAIN4	GAIN5	GAIN6	BIAS1	BIAS2	BIAS3	BIAS4	k
Phase Reset	k_f	GAIN1	GAIN2	GAIN3	GAIN4	GAIN5	GAIN6	BIAS1	BIAS2	BIAS3	BIAS4	

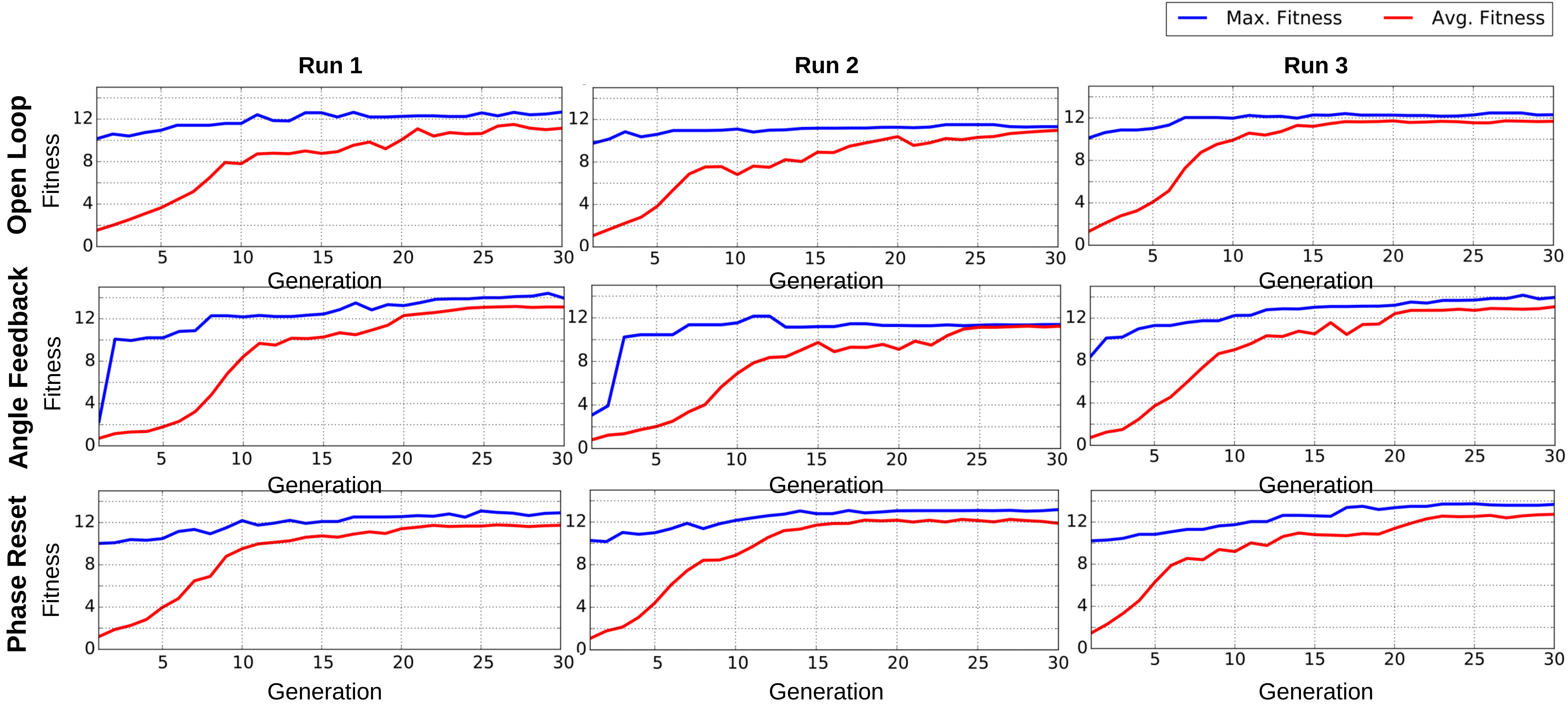
Genetic Algorithm Parameters

Population size	200 (20 seconds each)
Generations	30 (3 runs)
Selection	Tournament selection (size=3)
Crossover	2-point crossover ($\mathbb{P}$=80%)
Mutation	Number from a Gaussian distribution ($\mathbb{P}$=10%)

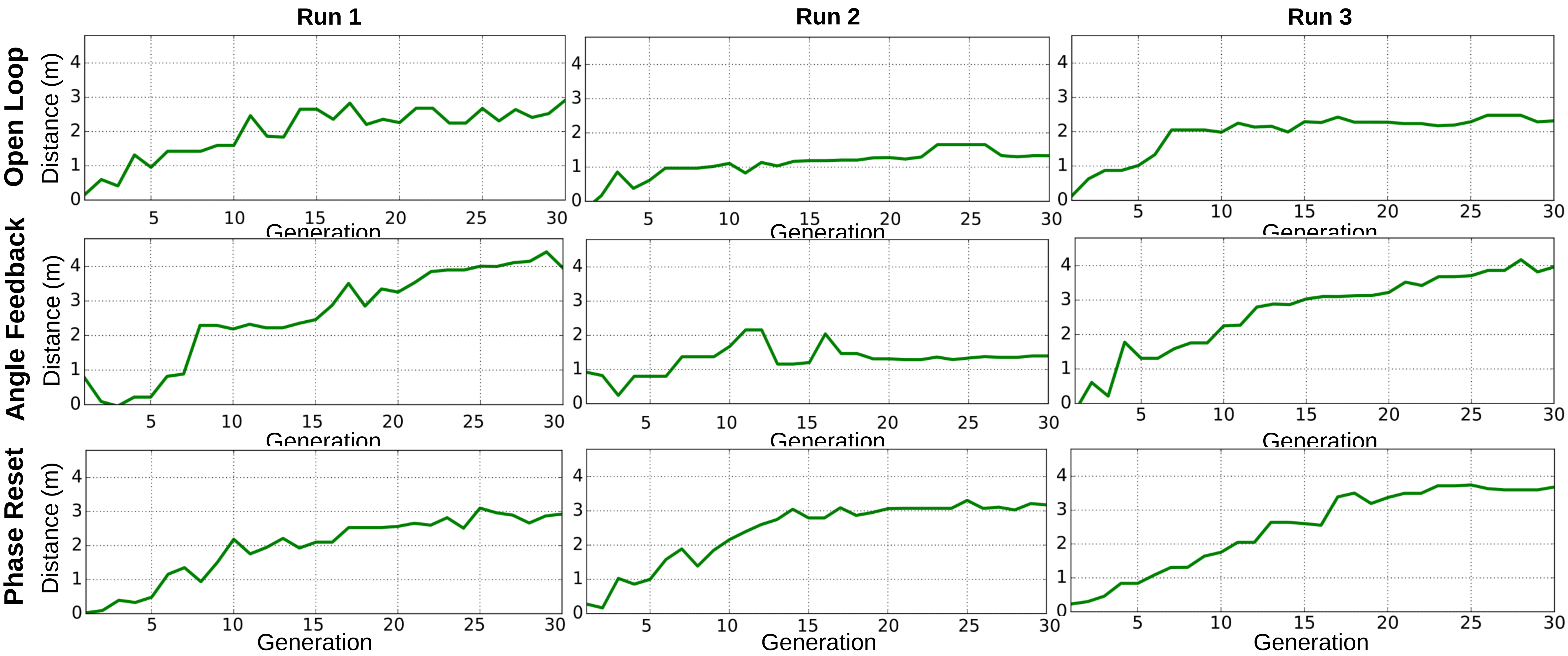
Fitness function

$$f = distance_x + (0.5 \times t_{up})$$

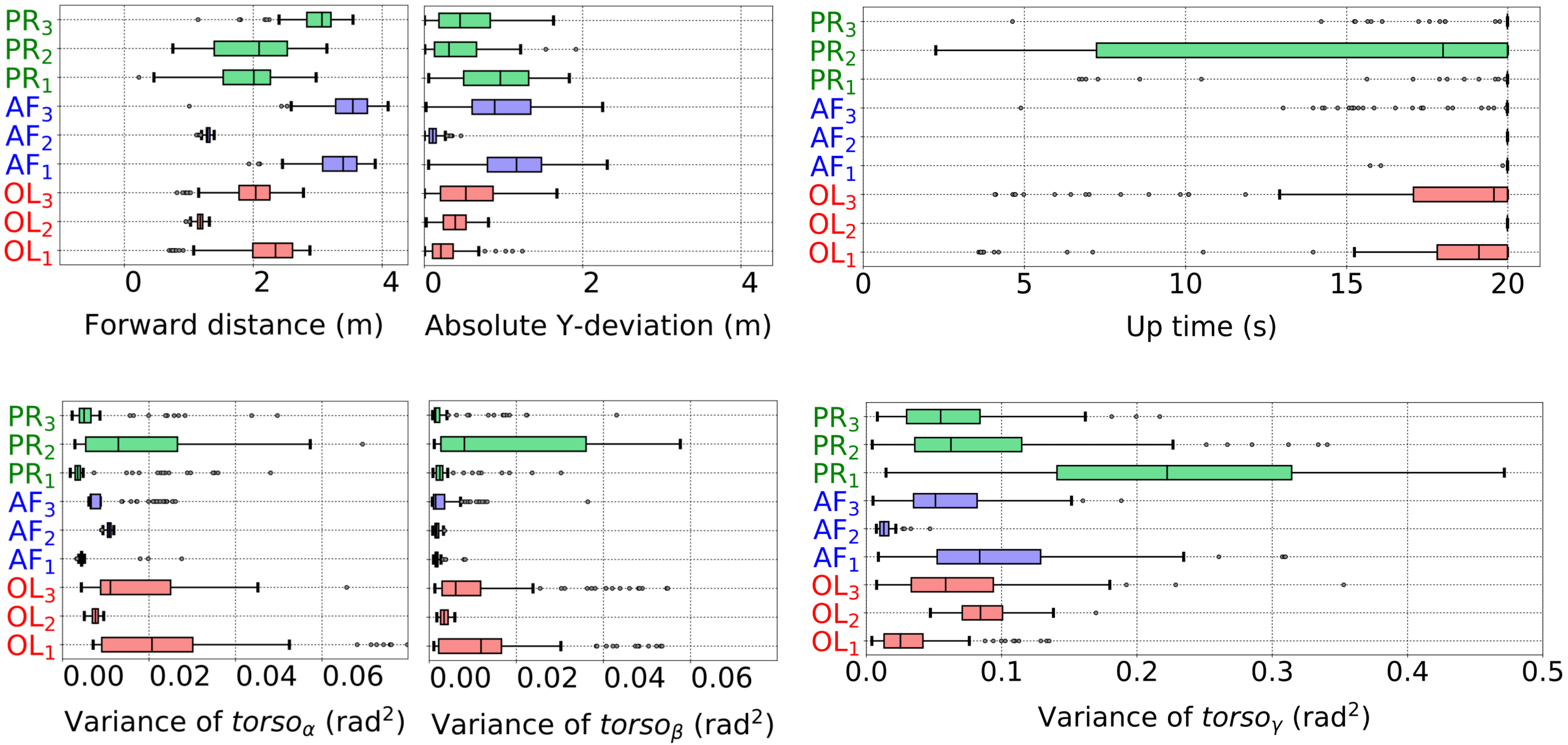
Low-level Control Optimization Results – Fitness



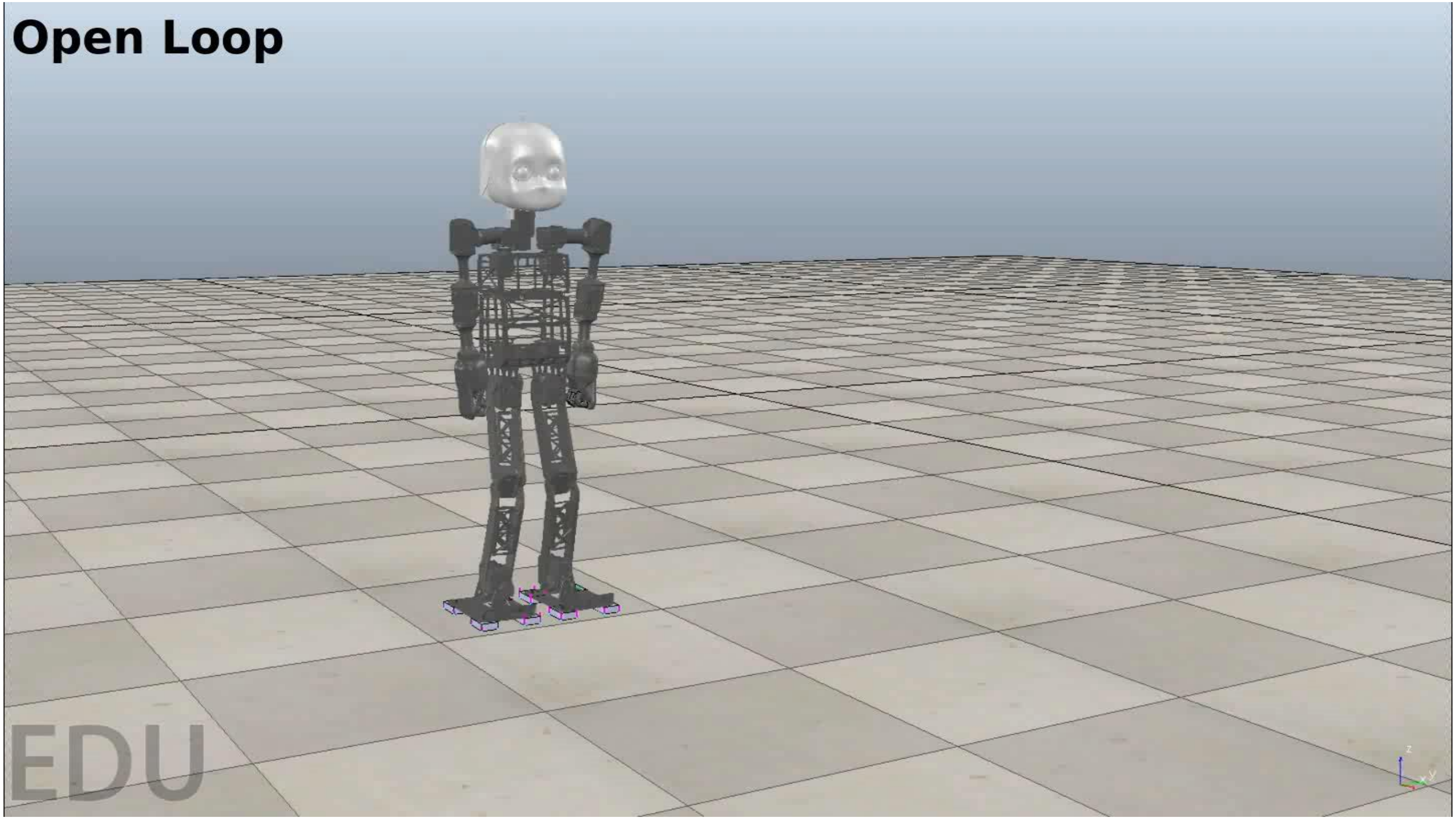
Low-level Control Optimization Results– Distance



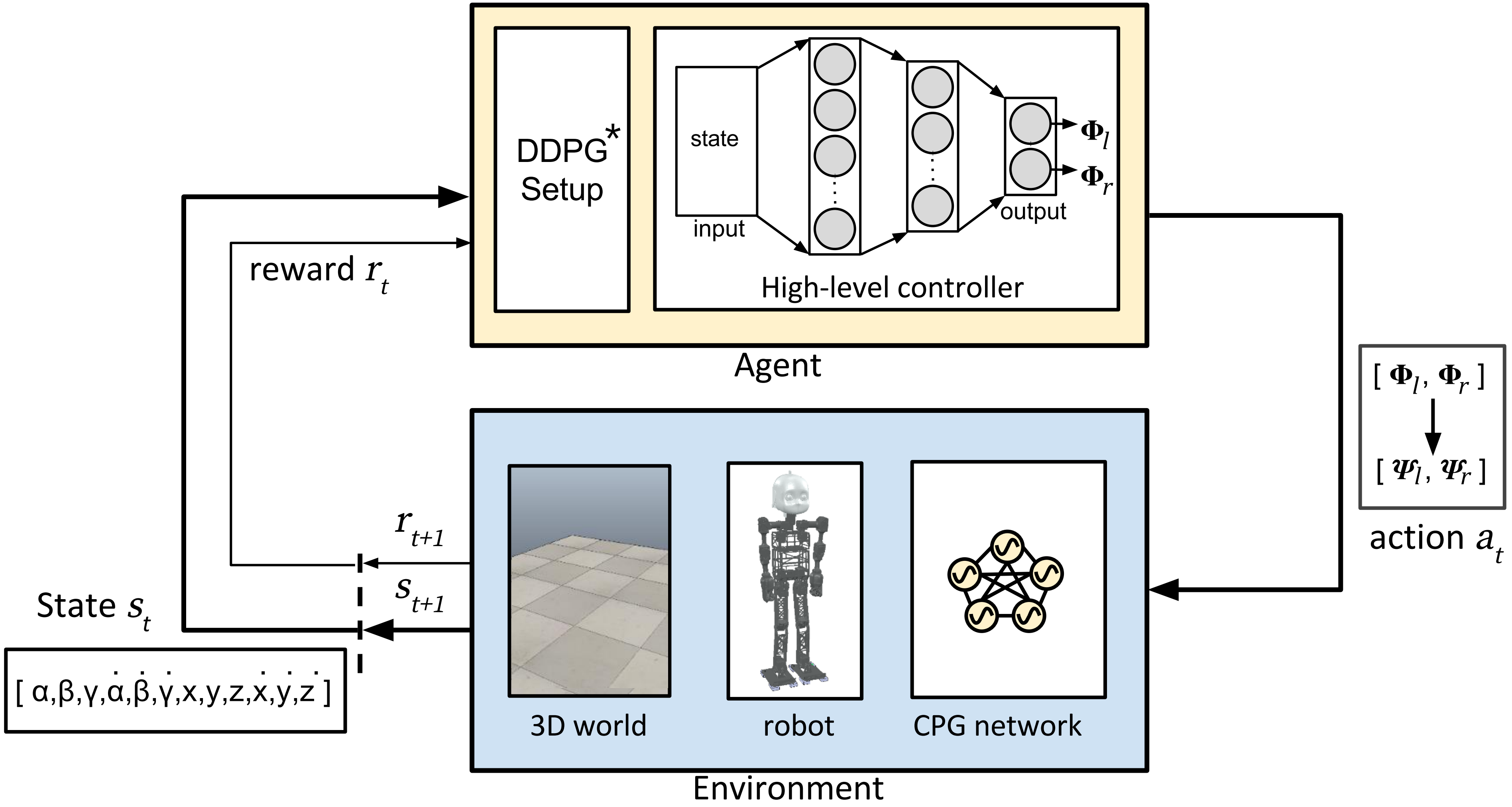
Gait Evaluation Test Results



Low-level Control – Videos

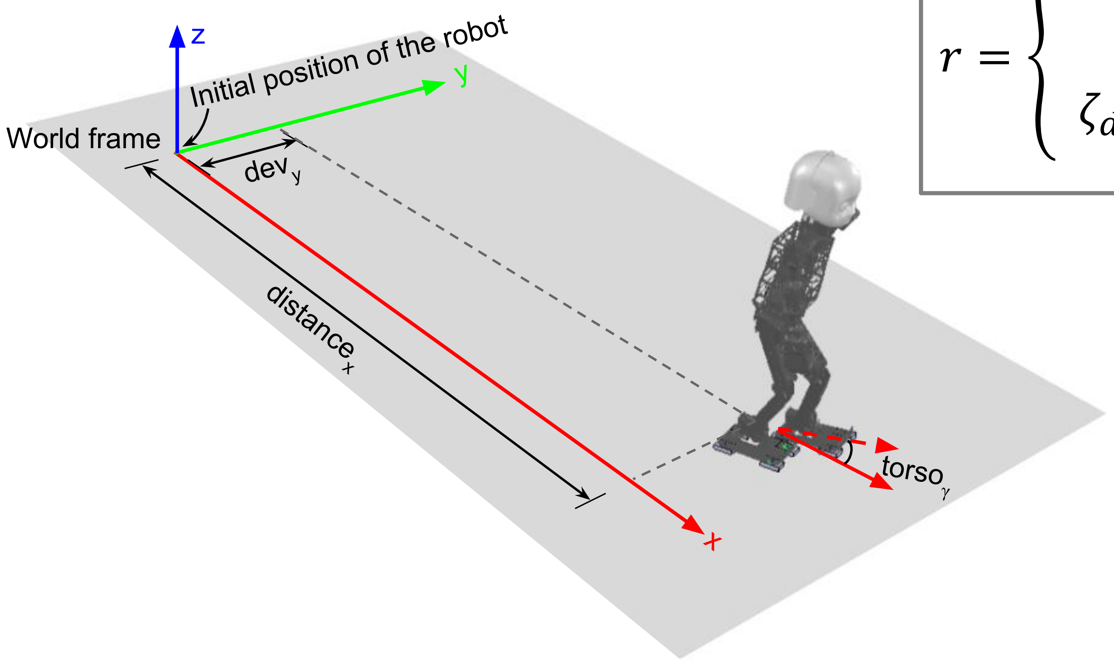


High-level Control - Setup



* Deep Deterministic Policy Gradient Algorithm [Lillicrap, 2015]

High-level Control – Training Setup



Reward Function

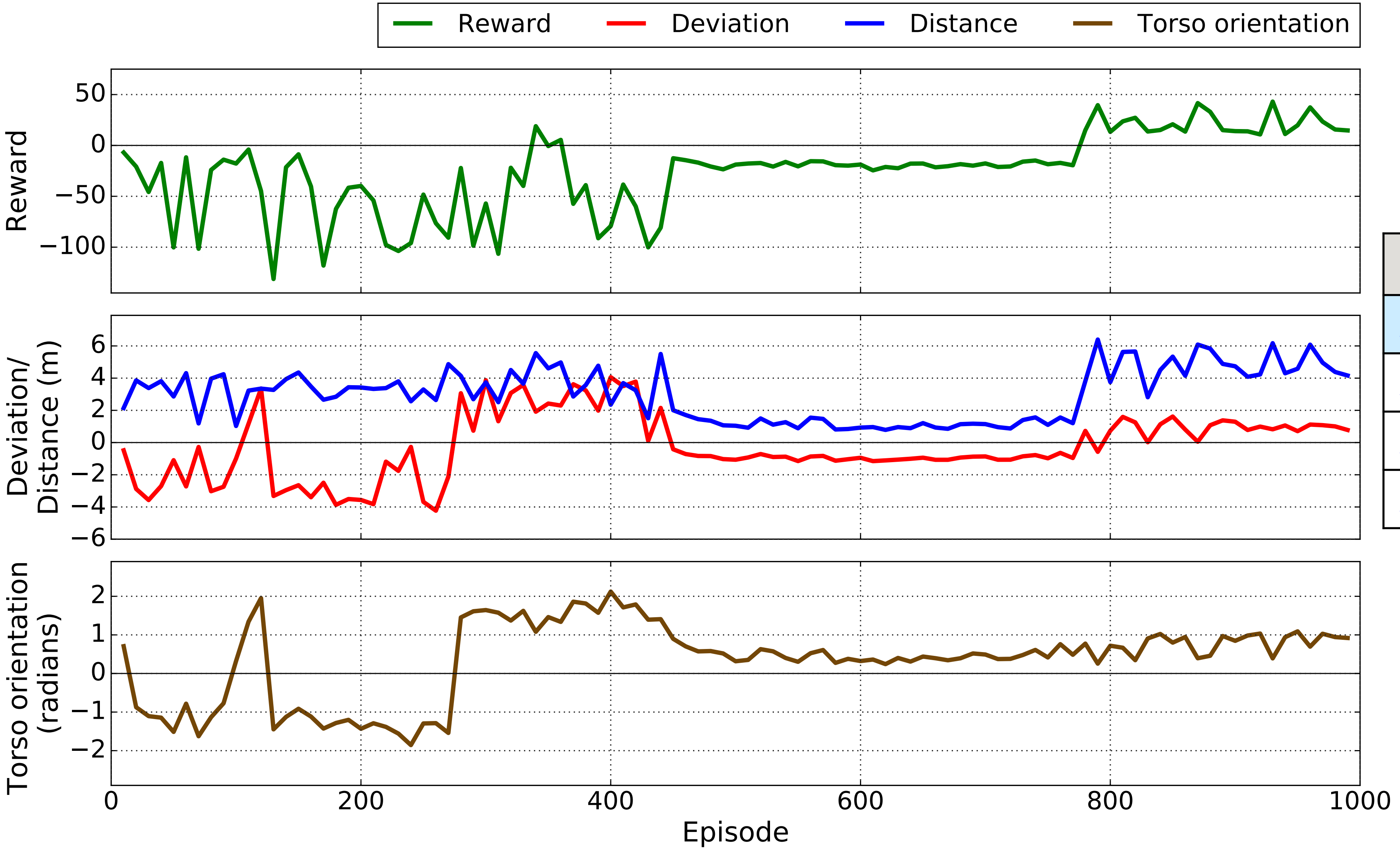
$$r = \begin{cases} -100 & \text{if the robot falls} \\ \zeta_{dev}(-dev_y^{abs}) + \zeta_{dist}(distance_x) + \zeta_\gamma(-torso_\gamma^{abs}) & \text{otherwise} \end{cases}$$

$\Psi_l \ \ \Psi_r$	gain factors
$\Phi_l \ \ \Phi_r$	Neural network output
ξ	Controls influence of Neural Network 1.0 – No influence 0.0 – High influence

Setup	ζ_{dev}	ζ_{dist}	ζ_γ	ξ
Setup 1	1.0	0.5	1.0	0.1
Setup 2	1.0	0.5	1.0	0.4
Setup 3	1.0	0.3	1.0	0.1
Setup 4	1.0	0.3	1.0	0.4

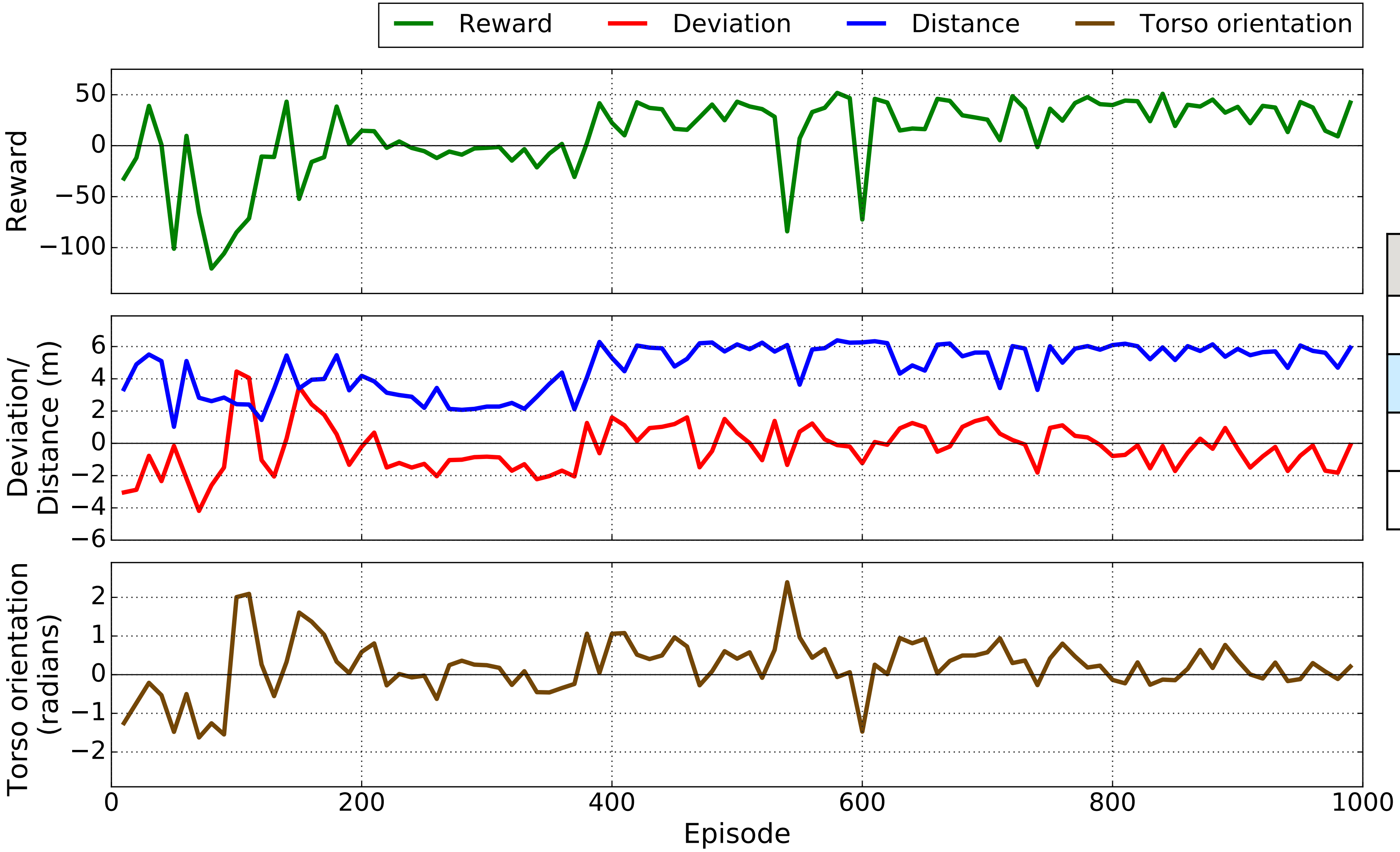
Setup Control	No high-level control

High-level Control – Training Results of Setup 1



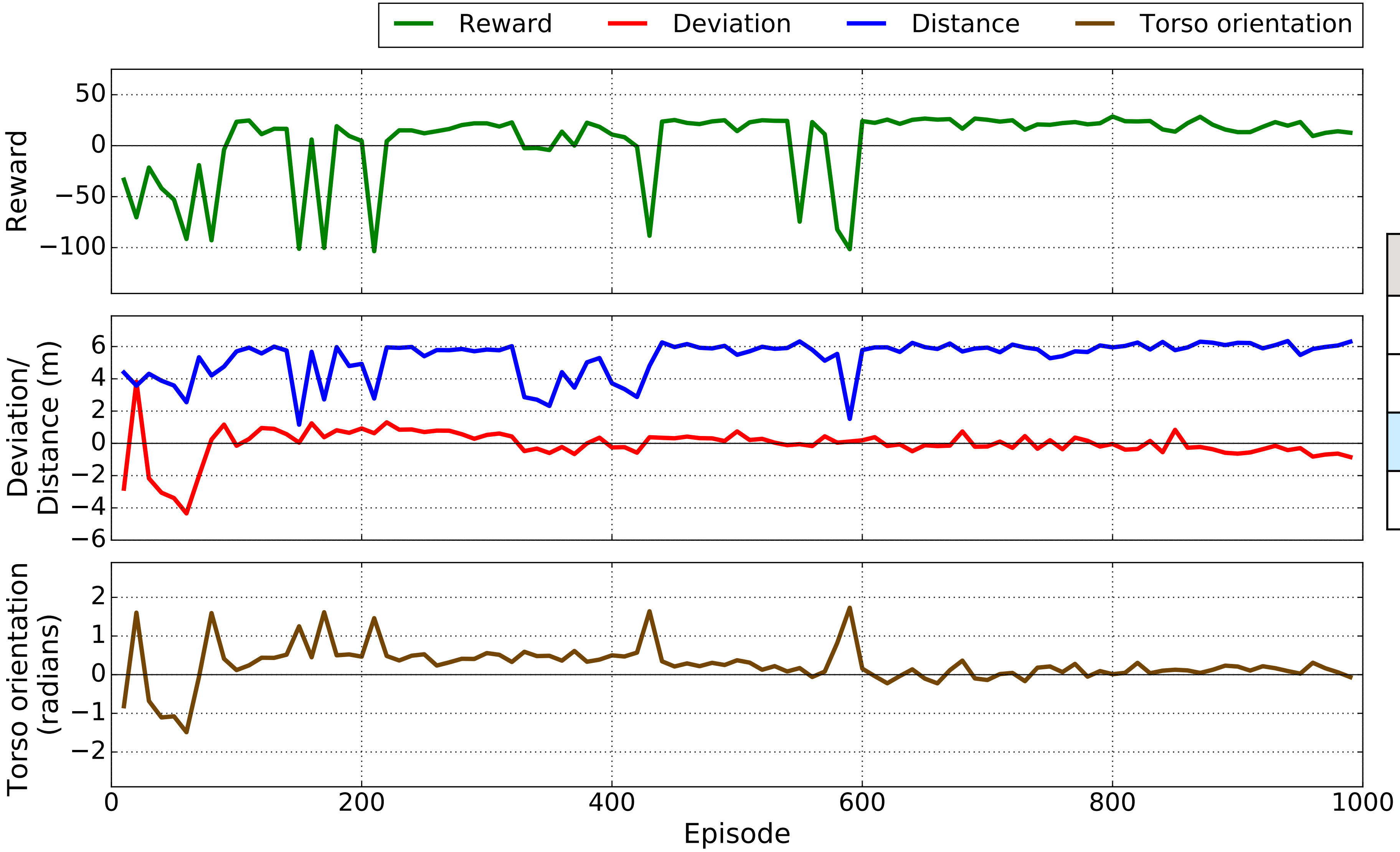
Setup	ζ_{dev}	ζ_{dist}	ζ_{γ}	ξ
Setup 1	1.0	0.5	1.0	0.1
Setup 2	1.0	0.5	1.0	0.4
Setup 3	1.0	0.3	1.0	0.1
Setup 4	1.0	0.3	1.0	0.4

High-level Control – Training Results of Setup 2



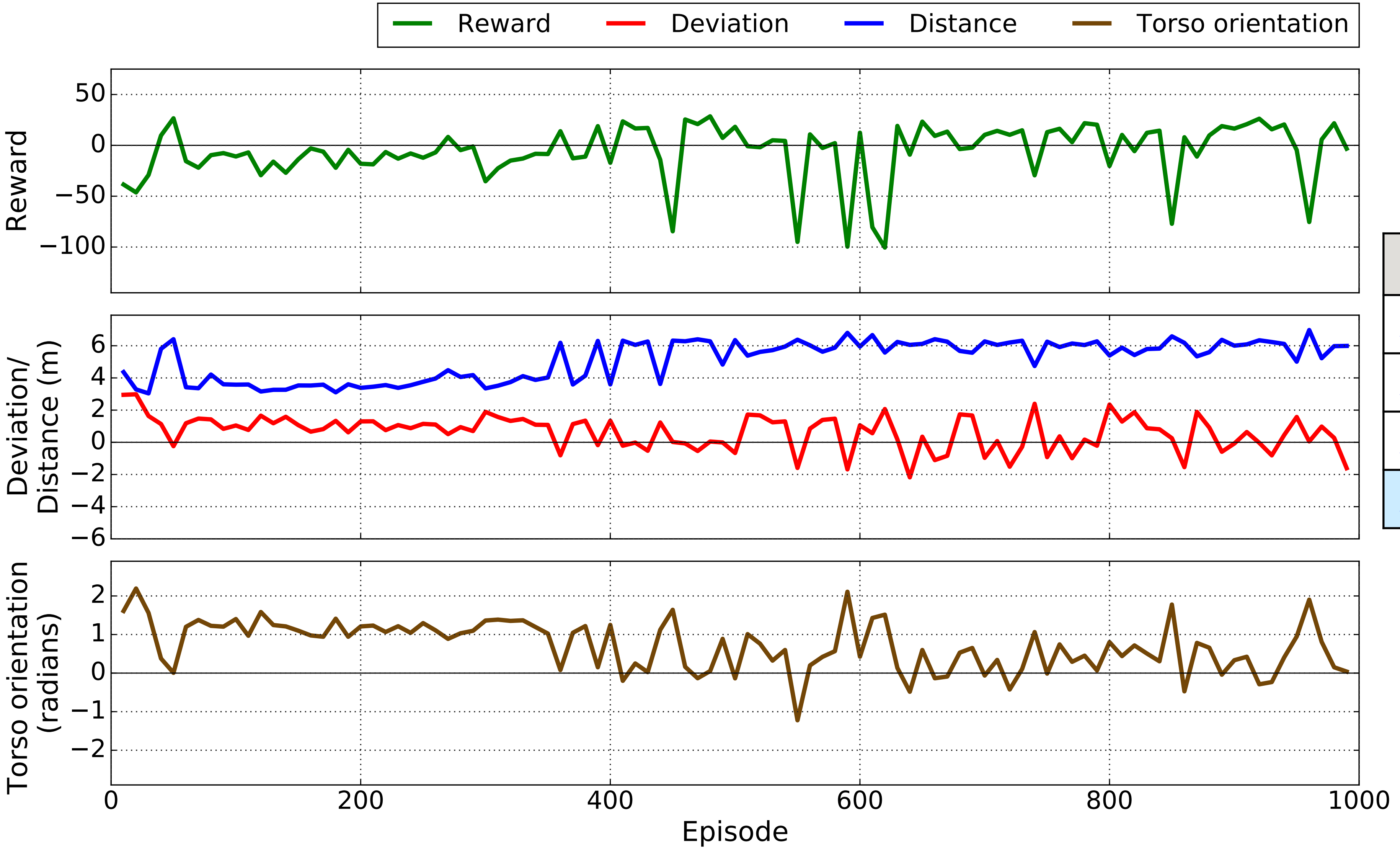
Setup	ζ_{dev}	ζ_{dist}	ζ_{γ}	ξ
Setup 1	1.0	0.5	1.0	0.1
Setup 2	1.0	0.5	1.0	0.4
Setup 3	1.0	0.3	1.0	0.1
Setup 4	1.0	0.3	1.0	0.4

High-level Control – Training Results of Setup 3



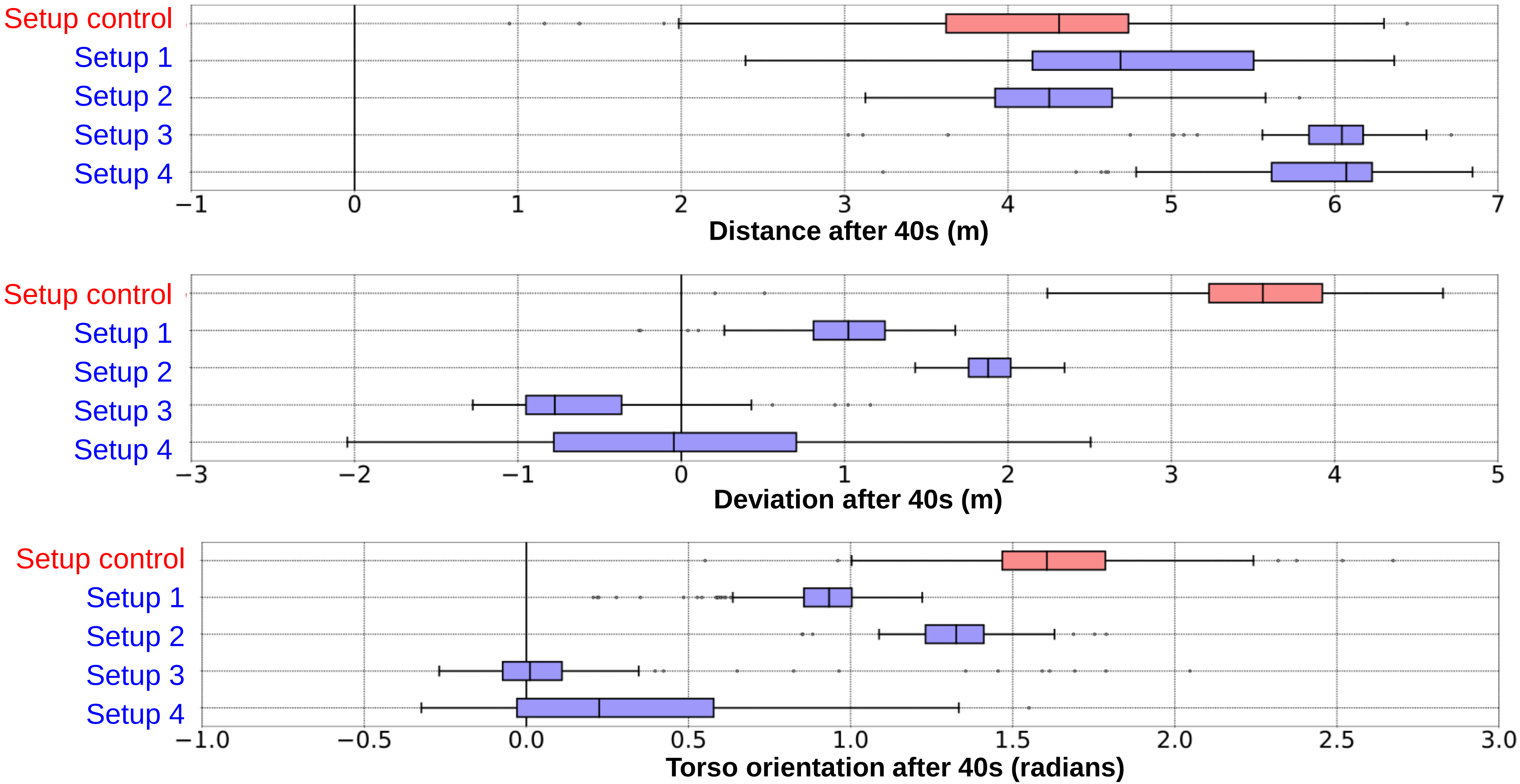
Setup	ζ_{dev}	ζ_{dist}	ζ_{γ}	ξ
Setup 1	1.0	0.5	1.0	0.1
Setup 2	1.0	0.5	1.0	0.4
Setup 3	1.0	0.3	1.0	0.1
Setup 4	1.0	0.3	1.0	0.4

High-level Control – Training Results of Setup 4



Setup	ζ_{dev}	ζ_{dist}	ζ_{γ}	ξ
Setup 1	1.0	0.5	1.0	0.1
Setup 2	1.0	0.5	1.0	0.4
Setup 3	1.0	0.3	1.0	0.1
Setup 4	1.0	0.3	1.0	0.4

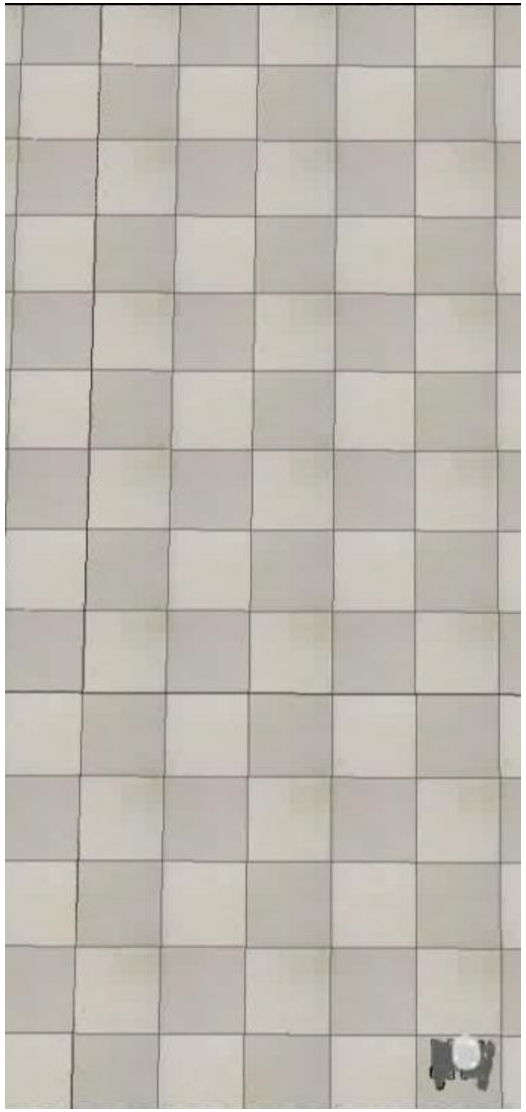
High-level Control – Test Results



High-level Control – Videos

All videos: **2.5x**

Control Setup



Setup 1

Setup 2

Setup 3

Setup 4

Conclusion

Summary

- Bio-inspired hierarchical controller for bipedal locomotion
- Low-level CPG controller optimized by a genetic algorithm
- Feedback mechanisms compared
- High-level neural network controller trained using reinforcement learning

Contribution

- High-level controller can improve the performance of CPG network
- Low-level feedback improves performance
- Hierarchical, modular controller – not tightly coupled to a particular robot

Future work

- Improve turning mechanism
- Combining low-level feedbacks
- Balance control
- Implementation on the physical robot

Thank you for your attention

Literature

[Cristiano, 2014] Julian Cristiano, Domenec Puig, and Miguel Angel Garcia. Locomotion control of a biped robot through a feedback CPG network. In ROBOT2013: First Iberian Robotics Conference, pages 527{540. Springer, 2014.

[Ishiguro, 2003] Akio Ishiguro, Akinobu Fujii, and Peter Eggenberger Hotz. Neuromodulated Control of Bipedal Locomotion Using a Polymorphic CPG Circuit. Adaptive Behavior, 11(1):7, 2003.

[Kamimura, 2005] Akiya Kamimura, Haruhisa Kurokawa, Eiichi Yoshida, Satoshi Murata, Kohji Tomita, and Shigeru Kokaji. Automatic locomotion design and experiments for a modular robotic system. IEEE/ASME Transactions on mechatronics, 10(3):314-325, 2005.

[Kerzel, 2017] Matthias Kerzel, Erik Strahl, Sven Magg, Nicolas Navarro-Guerrero, Stefan Heinrich, and Stefan Wermter. Nico - neuro-inspired companion: A developmental humanoid robot platform for multimodal interaction. In Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), pages 113-120, Aug 2017.

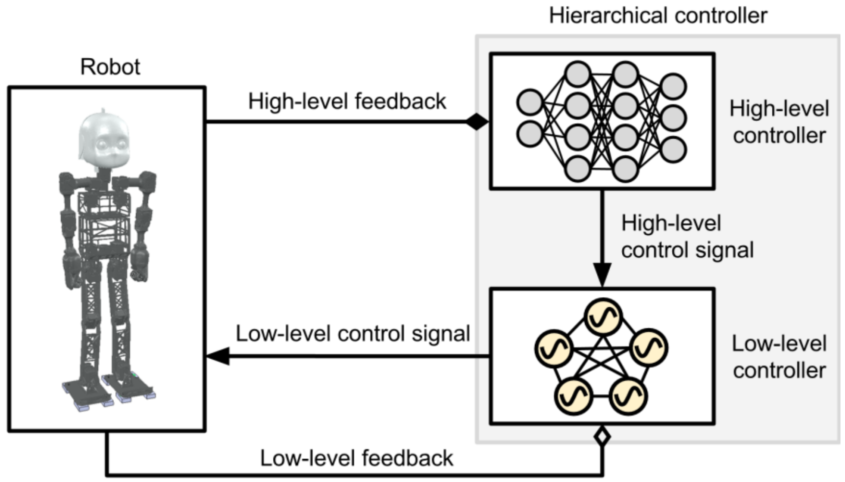
[Lillicrap, 2015] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971, 2015.

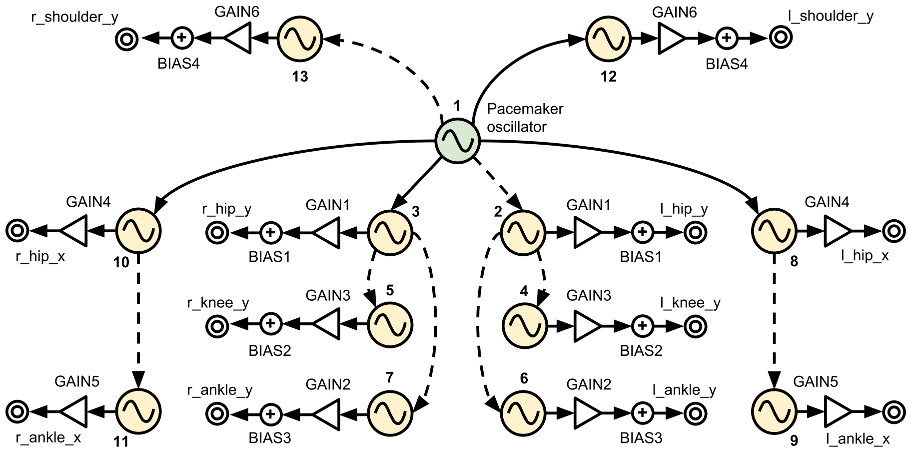
[Matsuoka, 1985] K Matsuoka. Sustained oscillations generated by mutually inhibiting neurons with adaptation. Biological cybernetics, 52(6):367-376, 1985.

[Matsuoka, 1987] Kiyotoshi Matsuoka. Mechanisms of frequency and pattern control in the neural rhythm generators. Biological Cybernetics, 56(5-6):345-353, 1987.

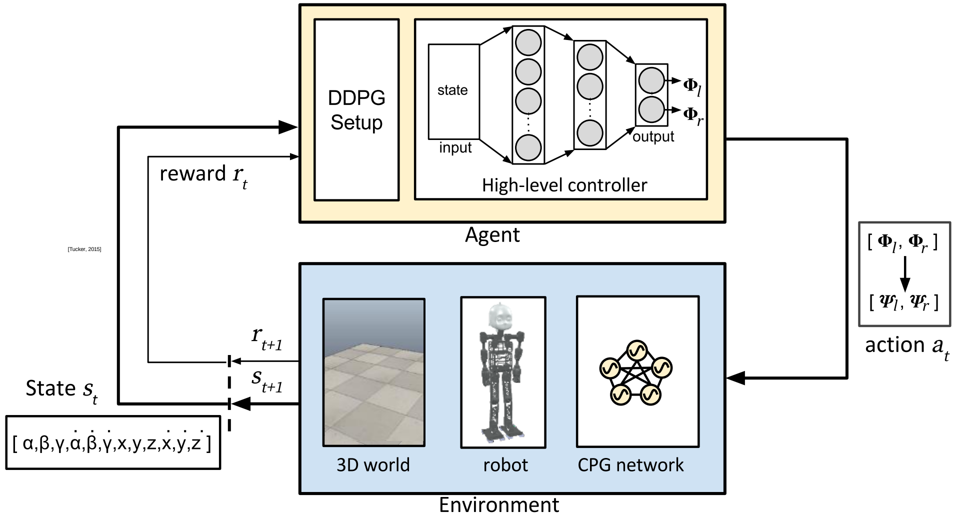
[Shepherd, 1994] Gordon M. Shepherd. Neurobiology. Oxford University Press, 1994.

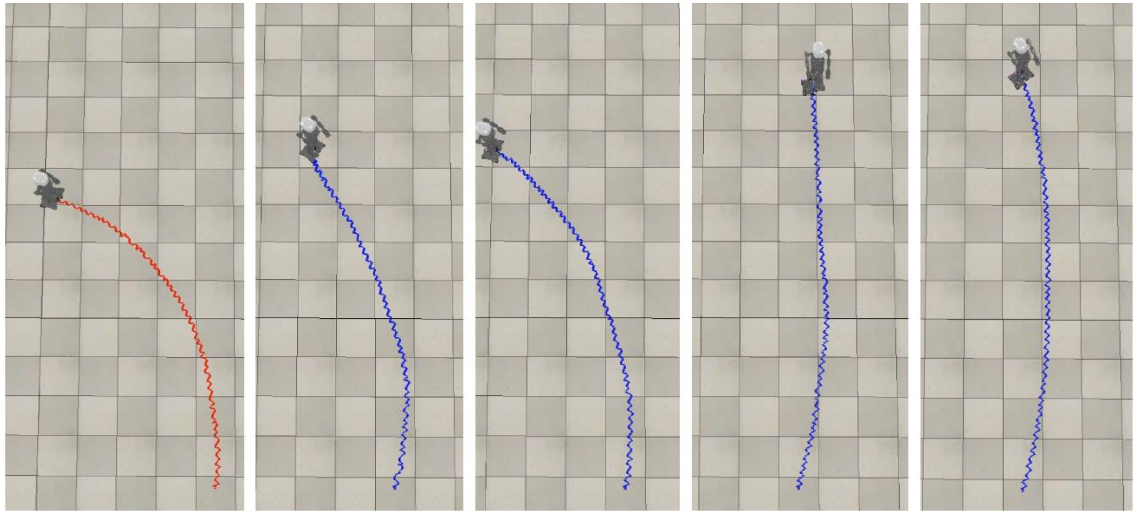
[Tucker, 2015] Tucker, Michael & Olivier, Jeremy & Pagel, Anna & Bleuler, Hannes & Bouri, Mohamed & Lambercy, Olivier & Millan, Jose del R. & Riener, Robert & Vallery, Heike & Gassert, Roger. (2015). Control Strategies for Active Lower Extremity Prosthetics and Orthotics: A Review. Journal of NeuroEngineering and Rehabilitation. 12. 1. 10.1186/1743-0003-12-1


Hierarchical controller
Robot
High-level feedback
High-level controller
High-level control signal
Low-level control signal
Low-level controller
Low-level feedback

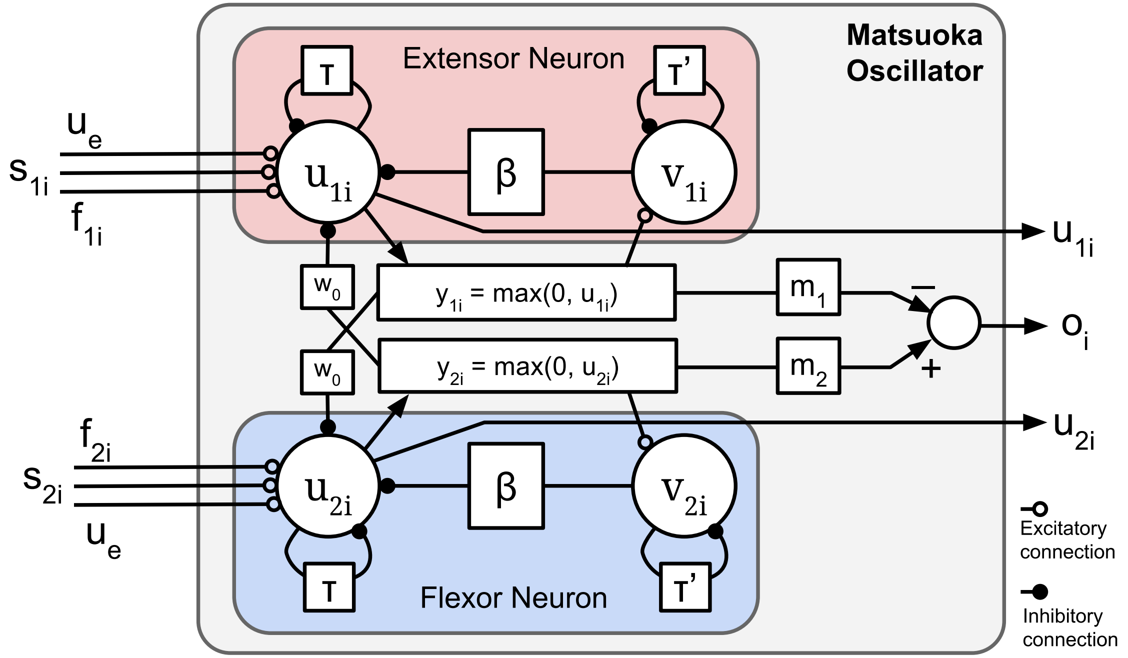

r_shoulder_y
GAIN6
BIAS4
13
GAIN6
l_shoulder_y
BIAS4
12
1
Pacemaker oscillator
GAIN4
r_hip_x
10
r_hip_y
GAIN1
BIAS1
3
2
GAIN1
l_hip_y
BIAS1
GAIN4
l_hip_x
8
r_knee_y
GAIN3
BIAS2
5
4
GAIN3
l_knee_y
BIAS2
GAIN5
r_ankle_x
11
r_ankle_y
GAIN2
BIAS3
7
6
GAIN2
l_ankle_y
BIAS3
GAIN5
l_ankle_x
9

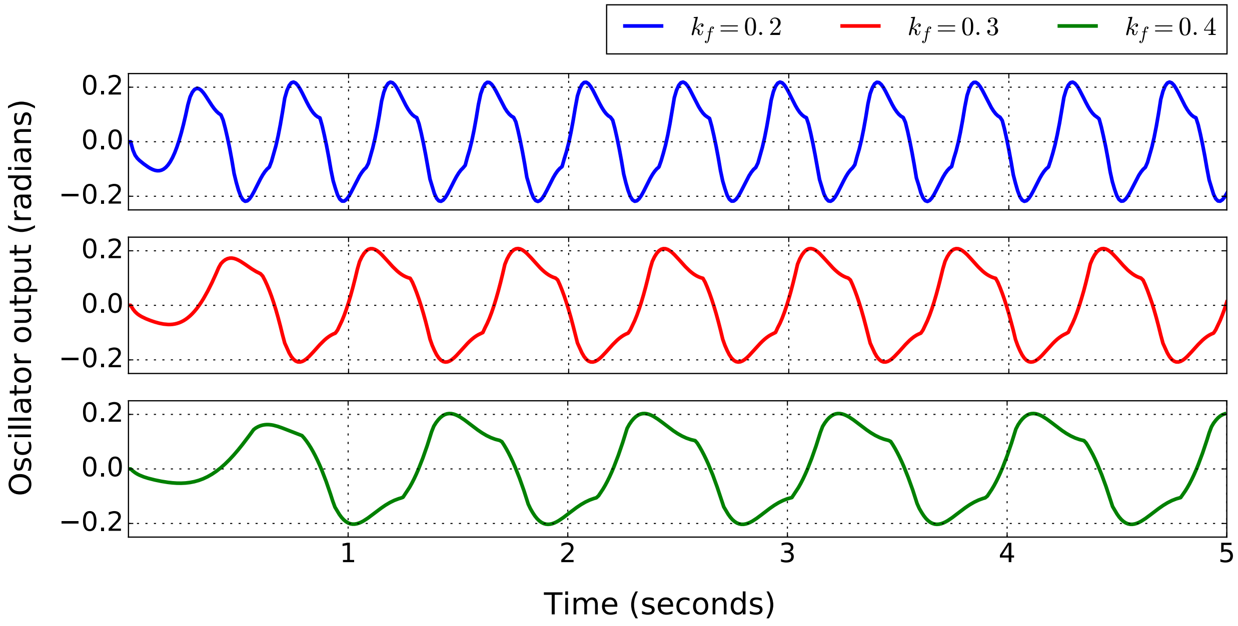
Questions?


DDPG Setup
state
input
output
Φ_l
Φ_r
High-level controller
Agent
reward r_t
$[\Phi_l, \Phi_r]$
$[\Psi_l, \Psi_r]$
action a_t
r_{t+1}
s_{t+1}
State s_t
$[\alpha,\beta,\gamma,\dot{\alpha},\dot{\beta},\dot{\gamma},x,y,z,\dot{x},\dot{y},\dot{z}]$
3D world
robot
CPG network
Environment



Matsuoka Oscillator Details





$$\begin{cases} \tau \dot{u_{1i}} = -u_{1i} - w_0 y_{2i} - \beta v_{1i} + u_e + f_{1i} + s_{1i} \\ \tau' \dot{v_{1i}} = -v_{1i} + y_{1i} \end{cases}$$

$$y_{1i} = max(0, u_{1i}) \text{ and } i = 1, ..., num$$

$$\begin{cases} \tau \dot{u_{2i}} = -u_{2i} - w_0 y_{1i} - \beta v_{2i} + u_e + f_{2i} + s_{2i} \\ \tau' \dot{v_{2i}} = -v_{2i} + y_{2i} \end{cases}$$

$$y_{2i} = max(0, u_{2i}) \text{ and } i = 1, ..., num$$

$$\tau = \tau_0 k_f$$

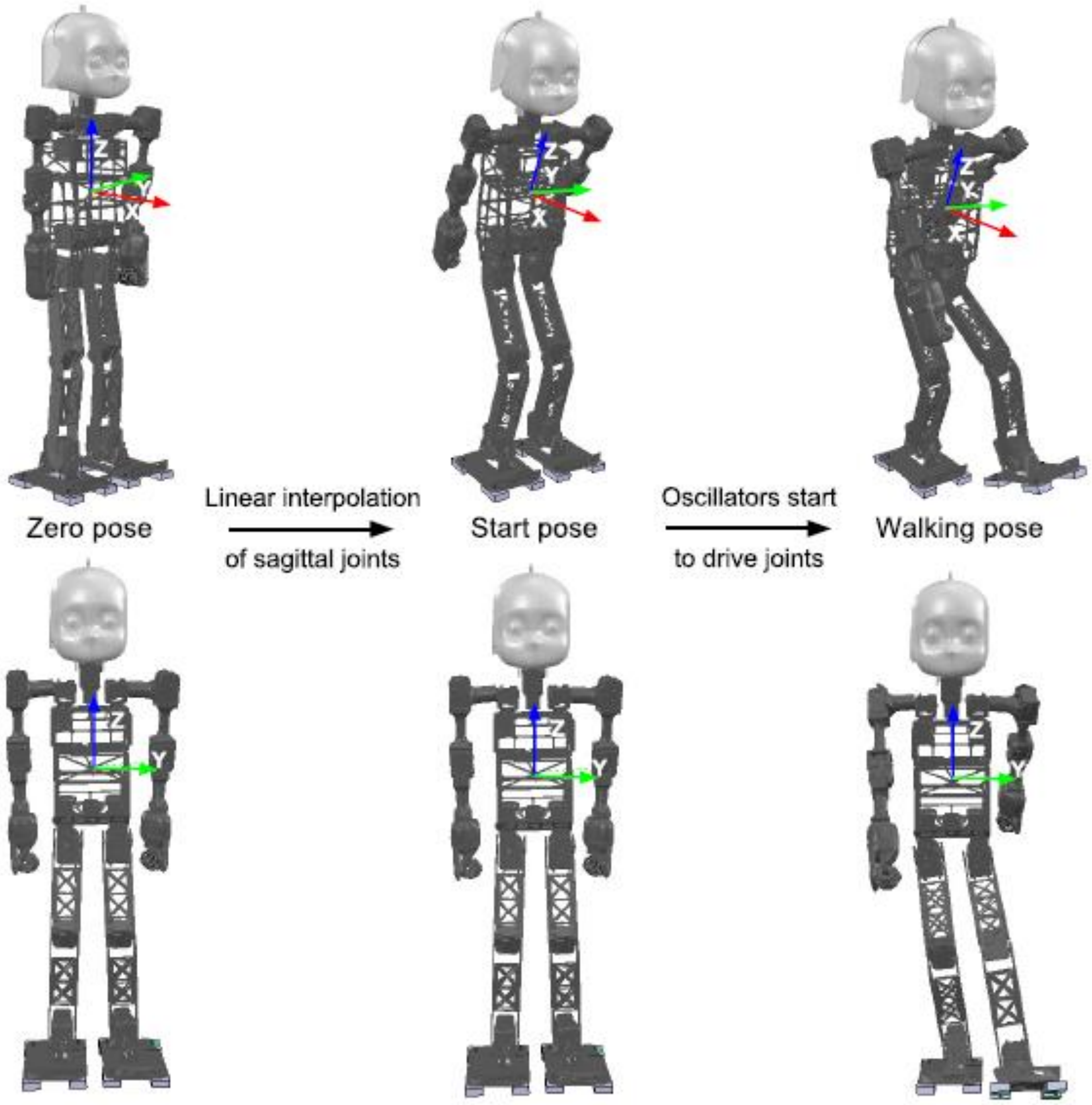
$$\tau' = \tau'_0 k_f$$

$$o_i = -m_1 y_{1i} + m_2 y_{2i}$$

$$s_{1i} = w_{ij} u_{1j}$$

$$s_{2i} = w_{ij} u_{2j}$$

Bias Position



Genetic Algorithm - Chromosomes

	Open loop			Angle feedback			Phase reset		
	run 1	run 2	run 3	run 1	run 2	run 3	run 1	run 2	run 3
k_f	0.4488	0.8144	0.4470	0.2258	0.7594	0.3178	0.3158	0.2840	0.2496
GAIN1	0.4167	0.7825	0.6213	0.5178	0.7708	0.3777	0.6792	0.4846	0.5626
GAIN2	0.1692	0.3372	0.0866	0.0101	0.0813	0.0234	0.0134	0.0469	0.0174
GAIN3	0.1861	0.0489	0.2071	0.0245	0.1381	0.0132	0.1478	0.0236	0.0140
GAIN4	0.6941	0.6607	0.7099	0.4555	0.8223	0.4567	0.4343	0.3094	0.3939
GAIN5	0.1372	0.0636	0.0818	0.2146	0.0223	0.2019	0.2060	0.2936	0.1845
GAIN6	0.5818	0.7723	0.4866	0.8918	0.2309	0.3309	0.3598	0.0883	0.9518
BIAS1	-0.1634	-0.0666	-0.1714	-0.1538	-0.0046	-0.0519	-0.1332	-0.2375	-0.3392
BIAS2	0.0356	0.3010	0.4983	0.1841	0.0895	0.0963	0.4287	0.2509	0.2576
BIAS3	-0.0147	-0.1895	-0.2999	-0.0973	-0.1217	-0.1156	-0.2780	-0.1445	-0.1045
BIAS4	0.0175	0.1774	0.0575	0.0107	0.5806	0.4814	0.5847	0.2630	0.0775
k	NA	NA	NA	0.7654	-0.6130	1.5364	NA	NA	NA

DDPG Algorithm