ASEC 2024 Conference
The 5th International Electronic Conference on Applied Sciences
04–06 December 2024 | Online
MDPI

Quadruped robot locomotion based on deep learning rules

Pedro Escudero-Villa*1, Danilo Machado-Merino2, Jenny Paredes-Fierro1
1 Facultad de Ingeniería, Universidad Nacional de Chimborazo, Riobamba 06018, Ecuador
2 Facultad de Ingeniería en Sistemas, Electrónica e Industrial. Universidad Técnica de Ambato, Ambato,180207, Ecuador

INTRODUCTION & AIM

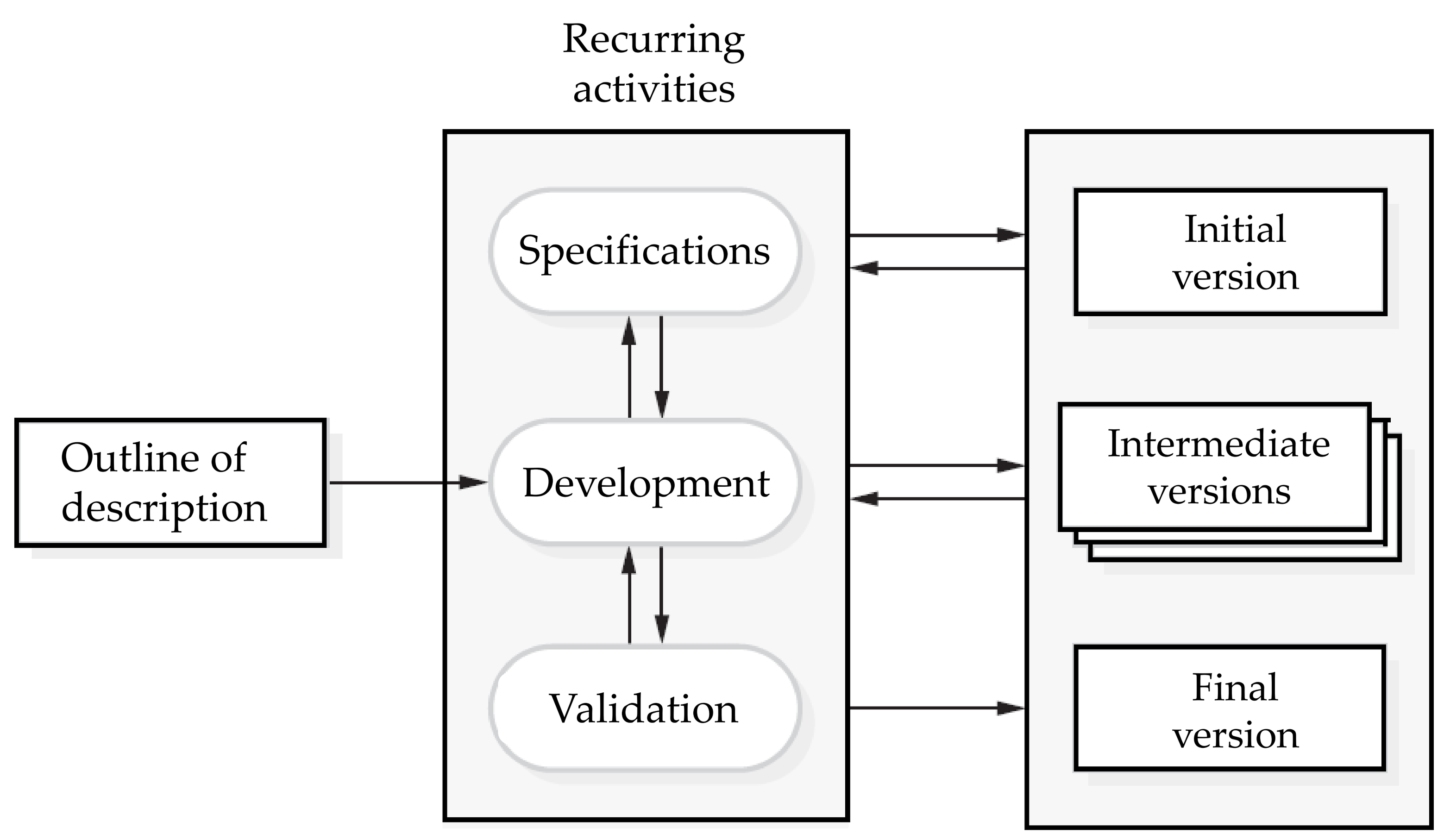
Service robots play a vital role in industrial and social development. For these reasons, this work aims to design and implement a reinforcement learning model based on Proximal Policy Optimization for the locomotion of a 12-degree-of-freedom quadruped robot, ensuring stability in its trajectory. The work implements a reinforcement learning model using TensorFlow and Gym, to test it in the PyBullet simulation environment. By adjusting the model's hyperparameters correctly, the goal is to achieve maximum stability in the robot's walking trajectory, particularly ensuring a smooth response at the center of gravity.

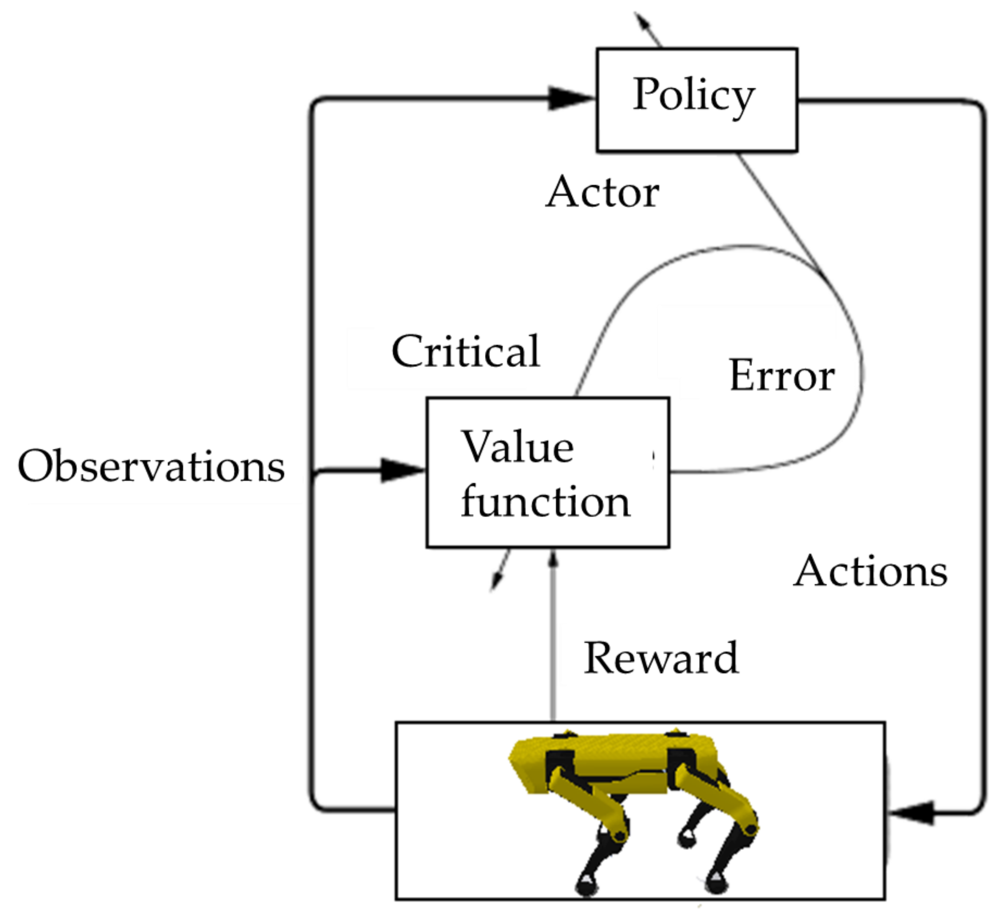
The specific objectives are:

- Analyze the kinematics and define the URDF design of a 12-degree-of-freedom quadruped robot.
- Study and apply the Proximal Policy Optimization algorithm to develop a locomotion model.
- Implement the developed model in a simulation environment for testing and validation.
- Evaluate the model's performance based on the stability of the robot's center of gravity relative to the ground.

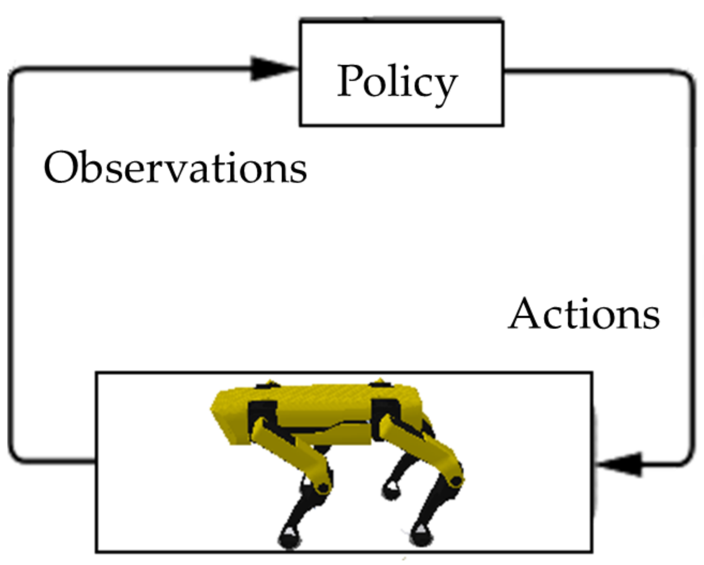
METHOD

Incremental Development Model





Critical - Actor Training Scheme

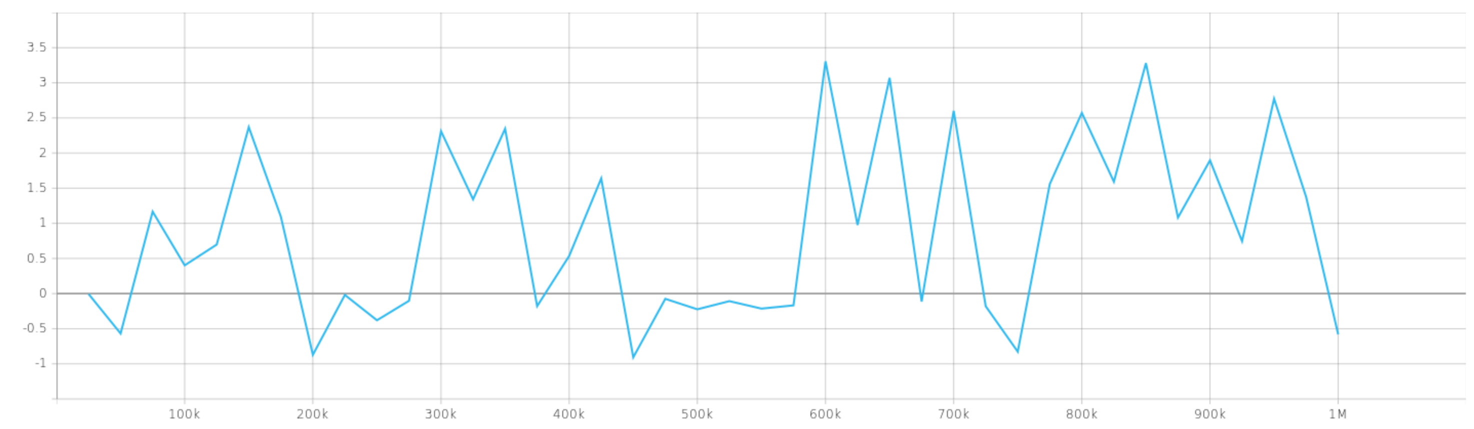


Model production Scheme

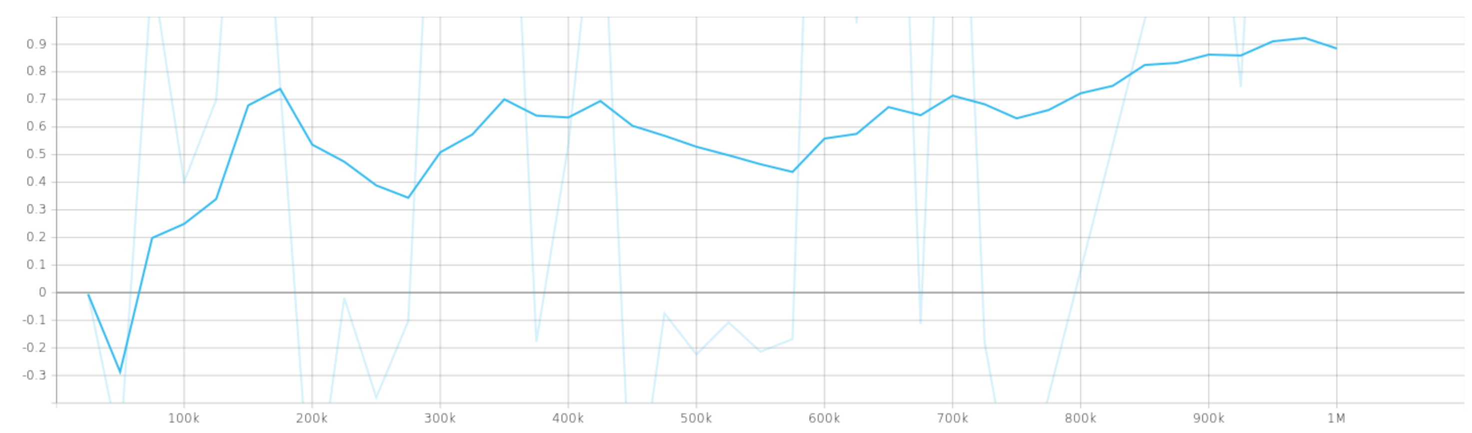
RESULTS & DISCUSSION

Model production

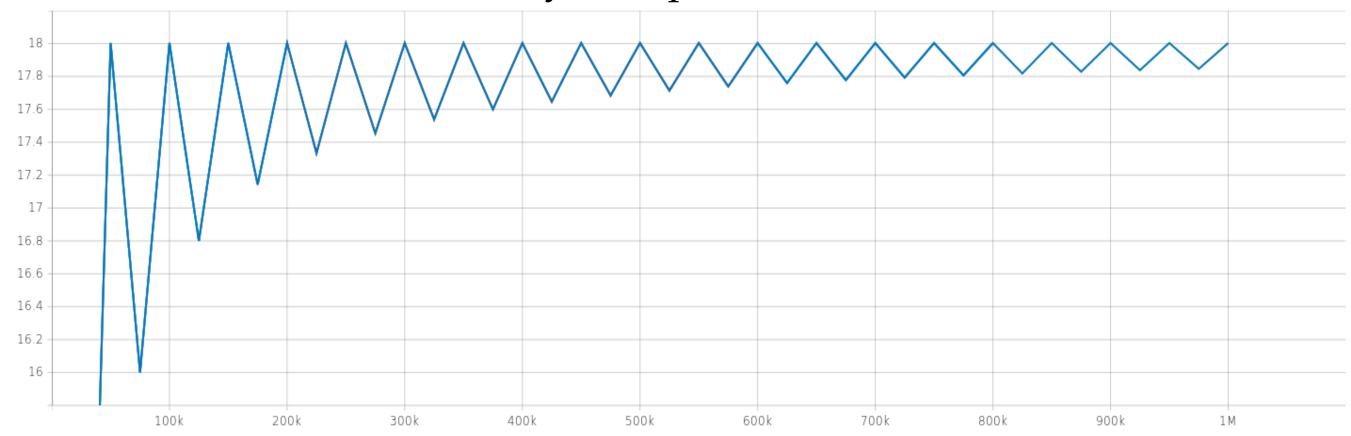
Reward obtained with SGD optimizer



Reward obtained with Adam optimizer



Data obtained from the Gyroscope sensor



Evaluations

Training Rewards		Data Obtained from the Gyroscope Sensor	KL penalty adjustment	
Steps	Average Reward	Average Height Variation (inches)	Loss function	Loss function
200K	1.4	2	-2.1e-4	0.40
400K	1.8	-	-	-
600K	1.6	0.5	-6.9e-4	0.18
800K	1.8	0.4	-5.5e-4	0.15
1M	0.6	0.25	-6.2e-4	0.13

CONCLUSION

- The PPO algorithm, specifically the actor-critic style, was analyzed and successfully implemented, with Adam chosen as the optimizer for better performance compared to the suggested SGD.
- The model demonstrated stable trajectory control for the robot, with continuous improvement in performance as training progressed, confirmed by the reduction in loss and KL penalty.
- The research confirms that the Proximal Policy Optimization algorithm is effective for robot control, generating continuous learning and stability in the robot's locomotion.

REFERENCES

[1] Raibert, M., Blankespoor, K., Nelson, G., & Playter, R. IFAC Proceedings Volumes, 41(2), 10822–10825.
[2] Liu, M., Xu, F., Jia, K., Yang, Q., & Tang, C. 3rd International Conference on Information Science and Control Engineering (ICISCE), 799–803, 2015
[3] Kalakrishnan, M., Buchli, J., Pastor, P., Mistry, M., & Schaal, S. (2010). IEEE International Conference on Robotics and Automation, 2665–2670, 2010.
[4] Di Carlo, J., Wensing, P. M., Katz, B., Bledt, G., & Kim, S. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020.
[5] Dudzik, T., Chignoli, M., Bledt, G., Lim, B., Miller, A., Kim, D., & Kim, S. 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 3664–3671.