

Interim Report on UESTC4006P(BEng) Final Year Project

Please start by saving this file with the name: GUID_Surname_UESTC4006P_Interim_year

**** Please add appropriate course code

Student Name	Chen Yu
Student Matriculation Number	2288975Y
UESTC Student Number	2016200104015
Degree programme	Electronics and Electrical Engineering (UESTC) BEng
Academic year	4

Placement Company (if appropriate)	--
Working Title of Project	Reinforcement Learning Policy-Search Algorithms for a Walking Robot
Name of First Supervisor	Andre Rosendo
Name of Second Supervisor	Faisal Tariq
Declaration of Originality and Submission Information	<i>I affirm that this submission is all my own work in accordance with the University of Glasgow Regulations and the School of Engineering requirements</i> Signed (Student) : <i>Chen Yu</i>

Your report should be NO more than 8 pages in length and include the below subject headings and incorporated within this document:

Work done so far including thorough literature review (at least 4 pages)

Conclusions from initial work (at least 1 page)

Work to be done (at least 1 page)

Revised Gantt Chart

Deadlines for submission of this report

Please upload this report via the Moodle page by the deadline mentioned in Table 1 of your project handbook.

Comments from your Second Supervisor will be made via Moodle or via email.

Interim Report on UESTC4006P(BEng) Final Year Project

Feedback from Second Supervisors: Second supervisors may provide their feedback by adding comments directly on Moodle taking into account the questionnaire below **or** by filling out the below form and uploading it to Moodle.

Name of Second Supervisor	
---------------------------	--

Was the report satisfactory?

Yes ☐ No ☐

Are you satisfied with the (updated) scope of the project?

Yes ☐ No ☐


Is the revised plan feasible?

Yes ☐ No ☐

Would you like to give any suggestions/recommendations?

Yes ☐ No ☐

Please write your comments in the space provided below:



Signature:

Date:

I. WORK DONE SO FAR INCLUDING THOROUGH LITERATURE REVIEW

According to my study process, I first summarise some learning materials especially online tutorials in the first part. Then with the knowledge in the relative fields been refreshed, I started reading papers and hence a brief literature review is shown in the second part. In addition, I also did some experiment exercises to help better understanding some concepts and exploring methods for my project.

1. Online Learning

1.1. Reinforcement Learning Tutorials

To enhance and refresh some fundamentals of reinforcement learning, I mainly follow the online course *COMPM050 Introduction to Reinforcement Learning*¹ by David Silver from UCL. The lecture series introduces most of the basic concepts in the field of reinforcement learning. I have finished the first 7 lectures, which have covered the topics of *Markov decision processes, planning by dynamic programming, model-free prediction, model-free control, value function approximation, and policy gradient methods*.

1.2. Robotics Tutorials

To comprehend the knowledge on the mechanics' side and review some necessary physics basics, I mainly follow a practical robotics tutorials *MATLAB and Simulink Robotics Arena*² from MathWorks. I have finished the part of walking robots which help me to overview of walking robot system.

2. Paper reading (Literature Reviews)

With the prerequired knowledge gained, I have also read some literature in the field of walking robots (especially biped robots), reinforcement learning and policy search. Since it is an interdisciplinary project that requires knowledge from both algorithm and mechanical aspect, I try to organise some important concepts and work relative to the project in this session.

2.1. Overview

The design of locomotion behaviours is a challenge that increases with the kinematic complexity of the robot. People are often susceptible to the fallacy that the state of the art in robotic control today heavily relies on machine learning. This is often not the case [1]. For instance, the humanoid robot Atlas from Boston Dynamics is one of the most impressive works in robot control, which is able to walk and run on irregular terrain, jump precisely with one or two legs, and even do a backflip, and it is reported that people often assume that Atlas uses reinforcement learning. However, publications from Bost Dynamics do not include explanations of machine learning algorithms for control. On the other hand, the machine learning techniques dose be able to provide solutions to locomotion problems, even with fundamental principles of robot locomotion not yet fully understood [2].

2.2. Reinforcement Learning algorithm

Reinforcement Learning (RL) constitutes a significant aspect of the machine learning field with numerous applications ranging from finance to robotics and a plethora of proposed approaches. Robotics is a very challenging application for RL since it involves interactions between a mechanical system and its environment. These methods could be categorised into value function method and policy search method.

2.2.1. Value Function Method

Value function methods are based on estimating the value (expected return) of being in a given state. The state-value function $V^\pi(s)$ is the expected return when starting in state s and following policy π . The best policy, given quality function $Q^\pi(s, a)$, can be found by choosing a greedily at every state: $\arg\max_a Q^\pi(s, a)$. Under this policy, we can also define $V^\pi(s)$ by maximising $Q^\pi(s, a)$: $V^\pi(s) = \max_a Q^\pi(s, a)$.

The value function methods provide the means for the derivation of the optimal value function that is used for the reconstruction of the optimal policy. This class of methods is quite popular in the context of model-based RL. For example, in [3] the authors employ a policy iteration approach using the Natural Actor-Critic algorithm [4]. In

¹ The course page: <http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html> and the playlist on YouTube: <https://www.youtube.com/playlist?list=PLqYmG7hTraZDM-OYHWgPebj2MfCFzFObQ>

² The playlist on YouTube: <https://www.youtube.com/playlist?list=PLn8PRpmsu08oLufaYWEvcuez8Rq7q4O7D>

Interim Report on UESTC4006P(BEng) Final Year Project

the critic part, the policy evaluation is performed using a temporal difference approach, namely the LSTD-Q(λ) [5], while the actor part performs the improvement of policy using the natural gradient approach [6].

2.2.2. Policy Search Method

Policy search methods do not need to maintain a value function model, but directly search for an optimal policy π^* . Typically, a parameterised policy π_θ is chosen, whose parameters are updated to maximise the expected return $E[R|\theta]$ using either gradient-based or gradient-free optimisation [7]. In [8], the direct gradient algorithm [9] is used for policy optimization. The policy is parametrized as an artificial neural network whose weights are modified during the policy search. The authors speed-up the learning time by training a simulation-optimal policy and transferring it to the real system which performs further modifications. In [10] the authors use a gradient ascent method for the maximization of the expected reward. The Probabilistic Inference for Learning Control (PILCO) framework is introduced in [11], which uses Gaussian process to estimate the transition model of the system and reduces the model error by taking the distribution of inputs into consideration, where the long-term prediction becomes more accurate. Cutler and How [12] successfully applied PILCO and Bayesian nonparametric prior to an inverted pendulum model (IPM). Englert et al. [13] also used PILCO to deal with an imitation learning problem.

However, PILCO relies on Gaussian processes (GPs), which prohibits its applicability to problems that require a larger number of trials to be solved. Further, PILCO does not consider temporal correlation in model uncertainty between successive state transitions, which results in PILCO underestimating state uncertainty at future time steps [14]. Deep PLICO is proposed in [15] to answer these shortcomings by replacing PILCO's Gaussian process with a Bayesian deep dynamics model while maintaining the framework's probabilistic nature and its data-efficiency benefits.

2.3. Background of Bipedal Walking Control

2.3.1. Stability Criteria in Bipedal Robot Control

Bipedal robot walking can be broadly characterized as static walking, quasi-dynamic walking, and dynamic walking. Different types of walking are generated by different walking stability criteria as follows.

1) Static Stability: The position of the centre of mass (COM) and centre of pressure (COP) are often used as stability criteria for static walking. Static stability is the oldest and the most constrained stability criterion, often used in the early days of bipedal robots. A typical static walking robot is SD-2 built by Salatian et al. [16].

2) Quasi-Dynamic Stability: The most well-known criterion for quasi-dynamic walking is based on the concept of zero moment point (ZMP) introduced by Vukobratović et al. in [17]. ZMP is a point on the ground where the resultant of the ground reaction force acts. ZMP is frequently used as a guideline in designing reference walking trajectories for many bipedal robots.

3) Dynamic Stability: The stability of dynamic walking is a relatively new stability paradigm. The most well-known criterion was introduced by McGeer [18], who proposed the concept of "passive dynamic walking" (PDW) in 1990. The stability of a bipedal robot depends solely on its dynamic balance.

2.3.2. Control Techniques for Bipedal Robots

2.3.2.1. Classical Control

ZMP-based control or CoP-based control utilizes the dynamics model of biped robots to compensate for the deviation of ZMP (CoP) from the desired position to maintain the stability of the robot [19]. Zhao et al. [20] proposed a human-locomotion-inspired walking control on a flat surface, where the multi-contact robotic walking gait was generated by a spring-loaded inverted pendulum model. Huzaifa et al. [21] proposed a stylistic gait generator in a compass-like under-actuated planar biped model, where variable gait styles were generated using model-based trajectory optimization. Shahbazi et al. [22] proposed a Kalman filter-based linear observer to predict the inclination of the ground, and the robot was able to stand on a platform and maintain its balance with different inclinations. Koolen [23] employed a momentum-based control framework and applied to the humanoid robot ATLAS where the robot was able to walk across rough terrain.

2.3.2.2. Machine Learning Control

Although these methods have been proved to be able to allow the robot to balance itself standing or walking on uneven terrains, these model-based control strategies require the full dynamic model of the robot, where it is computationally expensive and require a highly efficient processor to manage all the calculations at real-time. Furthermore, the external disturbances from the environment are normally hard to address. As a result, the

Interim Report on UESTC4006P(BEng) Final Year Project

shortcoming of the model-based classical control algorithm leads the recent research towards the implementation of machine learning algorithms on walking robots.

2.4. Learning Algorithms for Bipedal Robot Control

The basic idea of machine learning is to understand how the agent behaves after giving some input and then process the received feedback from the system. For these algorithms to work, only the inputs and consequential output of the system need to be specified. As a result, the dynamic model of the robot is no longer required or only partially required. This significantly reduces computational costs and reduces the complexity of the problem. In the field of supervised learning, Hénaff et al. [24] utilized a neural network to train a robot who was able to maintain balance on a slope with different inclinations. Saputra et al. [25] conducted a biologically inspired recursive neural network for the stabilization system required for supporting locomotion, in which motoric neurons and sensoric neurons were used to represent the muscular and sensor systems of the robot respectively. Although these methods can approximate the system dynamics, the learning outcome still highly relies on the accuracy of their teacher controllers [26].

Compared with supervised learning, reinforcement learning, however, involves interactions between the agent and the environment, where the agent can complete the desired tasks by gathering experience directly from its environment without target controllers. Hwang et al. [27] employed Q-learning method as a pure model-free RL scheme and applied it on a biped robot to maintain its walking stability on a seesaw, where individual learning algorithms were employed to maintain the balance of the robot on uphill, downhill and flat surface environment respectively. Compared with uniformly distributed state space, Wu et al. [28] proposed a Gaussian basis function-based abstraction method to reduce the number of state cells that increase the learning efficiency. The simulation results of the method showed that the robot was able to maintain its posture on a moving platform.

3. Experiment

3.1. Basic Reinforcement Learning Algorithm Coding

I had followed a series of video tutorials by Harrison online³. It is almost the most easy-understanding RL coding tutorials available online and helps me to do some coding exercises of basic reinforcement algorithm, including Q-learning and deep Q-learning.

3.2. Simulation and Training of Walking Robot based on Python

I tried to set up the environment and run a 3D walking robot model by Alan Guitard available online⁴. The training methodology is based on DDPG [29]. The implementation uses the Roboschool environment, an alternative of Mujoco environment. The other dependencies are Tensorflow ($\geq 1.9.0$), PyOpenGL ($\geq 3.1.0$), gym ($\geq 0.10.5$), numpy ($\geq 1.15.0$), and Pillow ($\geq 5.2.0$). The results are as followed (Figure 1).

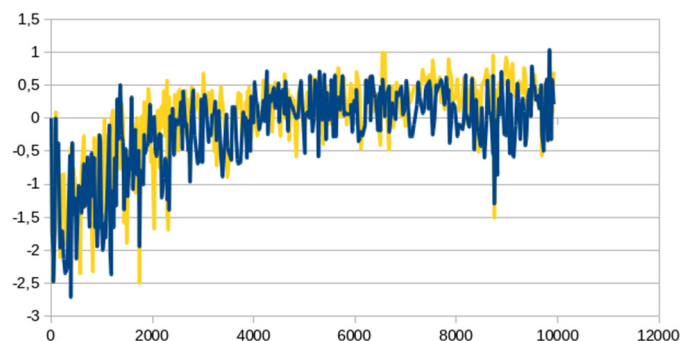


Figure 1. Result of training a walking robot with DDPG on Python (average reward per episode).

The fact is that the body learns how to stand but never learn to talk the first step. It is coherent since falling is giving a much lower reward than not moving in the wrong direction. This also shows the complexity of the task.

³ Link: <https://pythonprogramming.net/q-learning-reinforcement-learning-python-tutorial/>

⁴ GitHub repository: <https://github.com/AIEmerich/capstone-project>

Interim Report on UESTC4006P(BEng) Final Year Project

3.3. Simulation of Walking Robot based on MATLAB

I also tried to do some exercises according to the tutorials *MATLAB and Simulink Robotics Arena* mentioned in session 1.2. I began with a walking robot model based on Simulink, given by the tutorial⁵. I explored the model, ran the simulation and tried to understand each part of the model, as shown in Fig. 2 and 3.

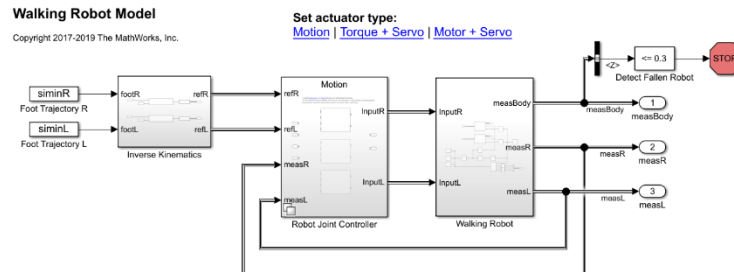


Figure 2. Block diagram of the walking robot.

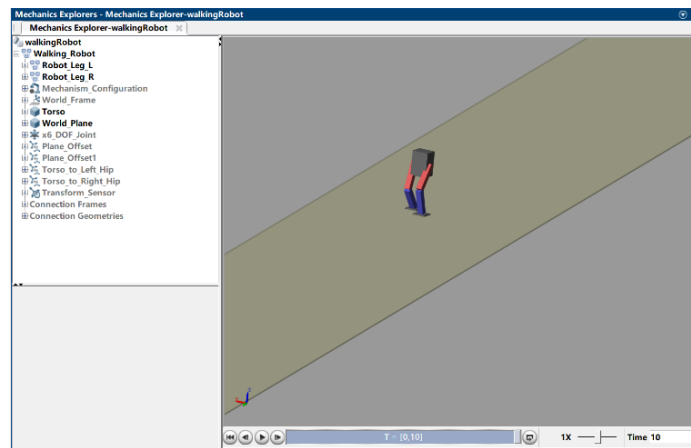


Figure 3. Simulation result of the model.

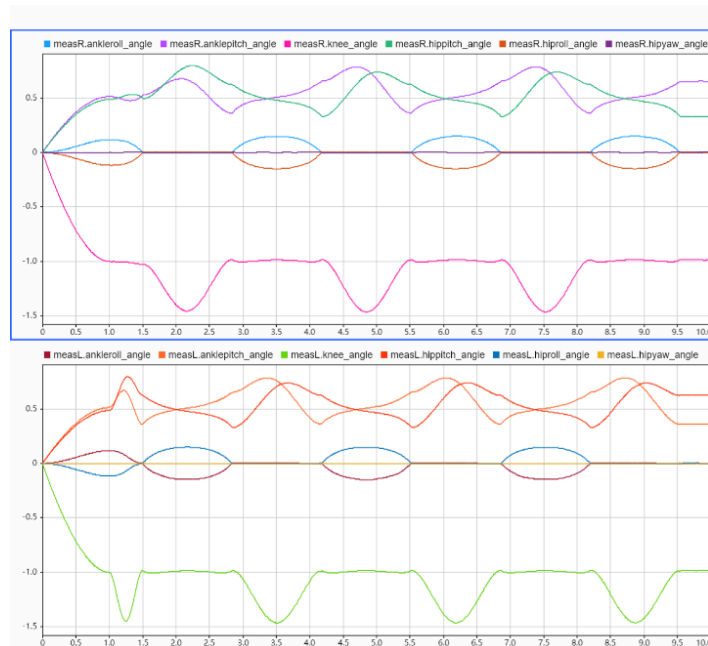


Figure 4. Angles for right leg (upper) and left leg (lower).

Plots of angles of each joint (ankle roll, ankle pitch, knee, hip pitch, hip roll, and hip yaw) for the right and left legs are as shown in Figure 4.

⁵ GitHub repository: <https://github.com/mathworks-robotics/msra-walking-robot>

II. CONCLUSIONS FROM INITIAL WORK

Overall, the main work I have done so far is learning the background knowledge required for my final year project, including the reinforcement learning algorithm, walking robot structure, implementation/coding methods, relative works, and so on. As a conclusion, I summarise the literature in my resent paper library in table 1 and table 2.

Table 1. Summary of my recent paper collection according to robot types, methods used, and experiment environments.

Robot	Publication	Method	Simulation	Physical implement
Biped robots	[30]	VGC-based balancing controller	-	Roborays
	[31]	PILCO	Compass gait walker, JenaWalker II.	-
	[31]	Stylistic gait generation framework	compass-like under-actuated planar biped model	-
	[32]	-	-	Honda Humanoid robot
	[33]	TD3 with RNN	Roboschool Walker2d	
	[34]	ZMP Adjusting	-	HRP-2 full-sized humanoid robot
	[35]	Neuro-Fuzzy Controller	3D model	Simple biped robot
	[35]	Naive reinforcement learning	-	SD-2 robot on a Sloping Surface
	[36]	Extended model ZMP control	✓	-
	[37]	Q-PROP	OpenAI Gym's MuJoCo	-
	[28]	Monte Carlo, State abstraction	Adams	-
	[22]	Kalman Filter etc.	-	2D biped robot Leo
	[10]	Gaussian processes, reinforcement learning	3D biped simulation model	Small humanoid robot
	[38]	Post-BL(imitation learning)	Simulated NAO robot (trained)	Real NAO robot (trained)
	[39]	Allowable ZMP Variation	-	HanSaRam-VII
	[40]	Stochastic policy gradient	-	Actuated version of a passive dynamic walker
	[41]	Q-learning	Simulated NAO robot	-
	[27]	Q-Learning	Robot in Webots simulator. (trained)	Real robot with 18 DOF. (trained)
	[42]	Model-based policy search in simulation and online model learning in the real robot.	Simulated biped robot.	Darwin-OP.
	[43]	Q-learning	Simulated biped robot (trained)	Real biped robot
	[25]	RNN	Simulated biped robot (trained)	Real biped robot
	[44]	Bayesian Optimization	-	ATRIAS robot
	[26]	Hybrid reinforcement learning	Simulated biped robot	-
	[45]	Grid Search, Pure Random Search, Gradient-descent Family, Bayesian Optimization	-	Bio-inspired dynamical bipedal walker Fox
Quadruped Robots	[46]	Deep reinforcement learning	Simulated Minitaurs (trained)	Real Minitaurs
	[47]	Policy gradient reinforcement learning	-	Sony Aibo
	[48]	Bayesian Optimization	-	BayesAnt

Interim Report on UESTC4006P(BEng) Final Year Project

Non-legged	[49]	PILCO	-	Festo Robotino XT
	[8]	Policy Gradient Based Reinforcement Learning	Simulated robot (trained)	Autonomous underwater vehicle
	[50]	PILCO	-	Multi-Task Robot Manipulator
	[15]	Deep PILCO	Cartpole swing-up	-
	[51]	Gaussian Processes and Reinforcement Learning	-	Autonomous Blimp

Table 2. Summary of the survey works in my paper library according to robot types, methods, and publication years.

Publication	Robot	Methods focus	Year
[52]	Bipedal Robot	Machine learning	2012
[53]	All	Reinforcement learning	2013
[7]	All	Policy Search	2013
[54]	All	Model-based reinforcement learning	2017
[1]	All	Behaviour Learning	2019
[55]	humanoid robot	All	2019

III. WORK TO BE DONE

Since I am from an EE background, I have to gain additional knowledge from the mechanical aspect and computer science side to finish my interdisciplinary project. As a result, I have to keep exploring the algorithm world and the robotics world and trying to bridge the gap between them. From a higher-level point of view, a hierarchy diagram of robot locomotion control is shown in figure 8, which is extracted from [55]. It is shown that locomotion problems can be organized hierarchically based on the controlled entities (single or multiple legs, joints of the robot body). E.g., on the lowest level, a PID controller may generate actuator commands to control the joints of a robot leg or the motors of its wheels to reach or maintain a certain position, velocity, or torque.

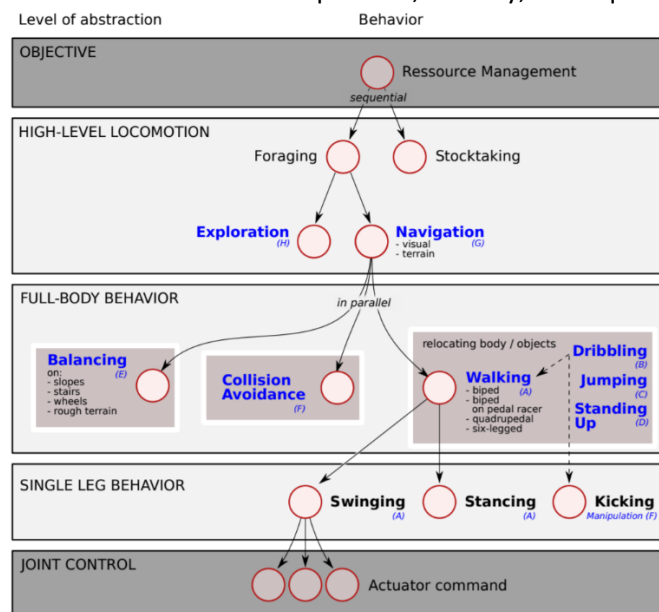


Figure 8. Hierarchy of behaviours with a focus on locomotion.

It could be observed that my project (a walking robot) is to control a robot in a full-body behaviour level. It means that while designing algorithms, I have to also consider the control of the lower levels, which requires a border knowledge scope out of machine learning. In addition, along with learning and theoretical designing, another challenging work is to finish the coding and physical implementation.

Interim Report on UESTC4006P(BEng) Final Year Project

In detail, I should keep learning in both the software and hardware sides through online tutorials, paper reading and other literature reading. I would keep following the course *Introduction to Reinforcement Learning* by David Silver as well as doing some relative exercises. After finishing that, I may take a more advanced course CS 285 *Deep Reinforcement Learning* at UC Berkeley⁶ or *Advanced Deep Learning & Reinforcement Learning* from UCL⁷. I would also keep searching, reading, and organising literature relative to *reinforcement learning*, *walking robot control*, *policy search* and so on. In addition, after visiting the lab in Shanghai, I would begin to explore the physical robot and try to use any methods to control it. After gained the main features of the physical structure, I would focus my paper reading scope, do some relative exercises, and do the implementation based on the robot. A revised Gantt Chart would be shown in the next session.

IV. REVISED GANTT CHART

A revised Gantt chart based on which in the preliminary report is shown in Figure 9.

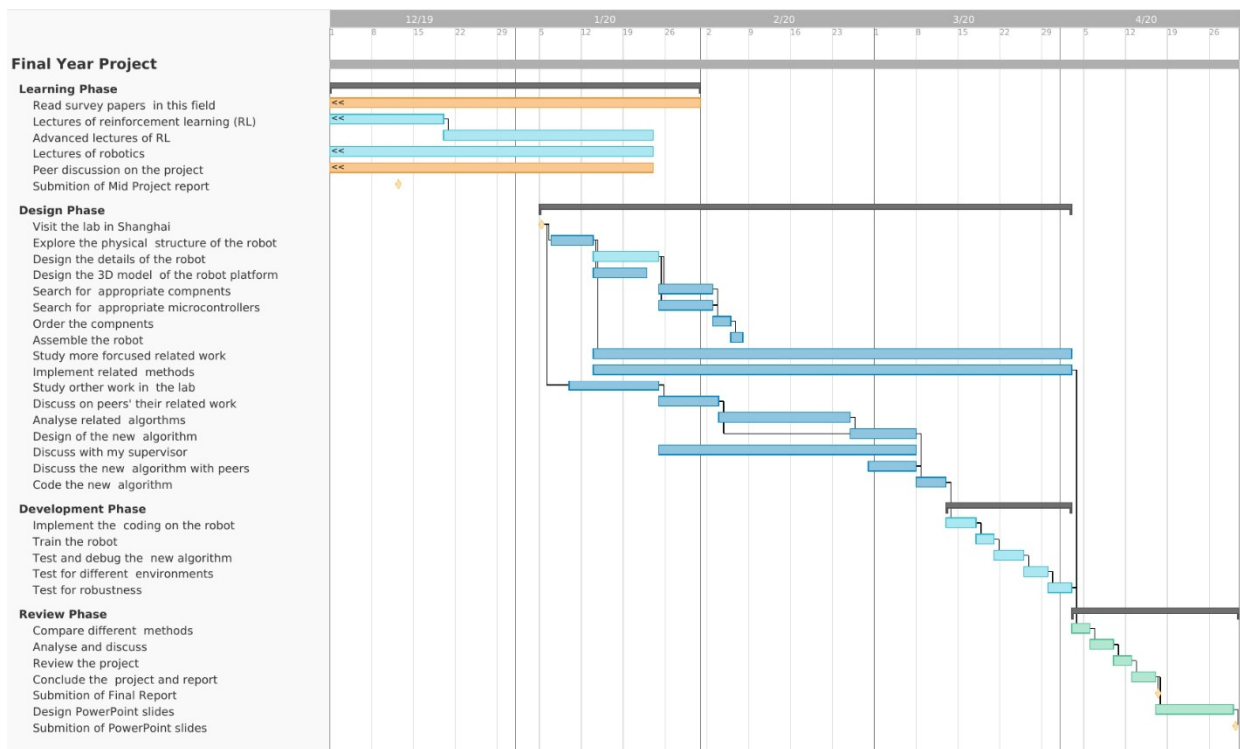


Figure 9. The revised Gantt chart of the project.

⁶ The course page: <http://rail.eecs.berkeley.edu/deeprlcourse/> and the YouTube playlist of lecture videos from Fall2018: https://www.youtube.com/playlist?list=PLkFD6_40KJlxJMR-j5A1mkxK26gh_qg37

⁷ The YouTube playlist: https://www.youtube.com/playlist?list=PLqYmG7hTraZDNJre23vqCGIVpfZ_K2RZs

REFERENCES

- [1] A. Fabisch, C. Petzoldt, M. Otto, and F. Kirchner, "A survey of behavior learning applications in robotics - state of the art and perspectives," *CoRR*, vol. abs/1906.01868, 2019.
- [2] J. Aguilar, T. Zhang, F. Qian, M. Kingsbury, B. McInroe, N. Mazouchova, C. Li, R. Maladen, C. Gong, M. Travers, R. L. Hatton, H. Choset, P. B. Umbanhowar, and D. I. Goldman, "A review on locomotion robophysics: the study of movement at the intersection of robotics, soft matter and dynamical systems," *Reports on Progress in Physics*, vol. 79, p. 110001, sep 2016.
- [3] B. Depraetere, M. Liu, G. Pinte, I. Grondman, and R. Babu 拏 ka, "Comparison of model-free and model-based methods for time optimal hit control of a badminton robot," *Mechatronics*, vol. 24, no. 8, pp. 1021 – 1030, 2014.
- [4] J. Peters and S. Schaal, "Natural actor-critic," *Neurocomputing*, vol. 71, pp. 1180–1190, 03 2008.
- [5] J. Peters, S. Vijayakumar, and S. Schaal, "Reinforcement learning for humanoid robotics," *Proceedings of the third IEEE-RAS international conference on humanoid robots*, pp. 1–20, 01 2003.
- [6] S.-I. Amari, "Natural gradient works efficiently in learning," *Neural Comput.*, vol. 10, pp. 251–276, Feb. 1998.
- [7] M. Deisenroth, G. Neumann, and J. Peters, *A Survey on Policy Search for Robotics*, vol. 2. 08 2013.
- [8] A. El-Fakdi and M. Carreras, "Policy gradient based reinforcement learning for real autonomous underwater cable tracking," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3635–3640, Sep. 2008.
- [9] A. Girard, C. E. Rasmussen, J. Q. n. Candela, and R. Murray-Smith, "Gaussian process priors with uncertain inputs application to multiple-step ahead time series forecasting," in *Advances in Neural Information Processing Systems 15* (S. Becker, S. Thrun, and K. Obermayer, eds.), pp. 545–552, MIT Press, 2003.
- [10] J. Morimoto and C. G. Atkeson, "Nonparametric representation of an approximated poincaré map for learning biped locomotion," *Autonomous Robots*, vol. 27, pp. 131–144, 2009.
- [11] M. P. Deisenroth and C. E. Rasmussen, "Pilco: A model-based and data-efficient approach to policy search," in *International Conference on International Conference on Machine Learning*, 2011.
- [12] M. Cutler and J. P. How, "Efficient reinforcement learning for robots using informative simulated priors," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2605–2612, May 2015.
- [13] P. Englert, A. Paraschos, J. Peters, and M. P. Deisenroth, "Model-based imitation learning by probabilistic trajectory matching," in *2013 IEEE International Conference on Robotics and Automation*, pp. 1922–1927, May 2013.
- [14] M. P. Deisenroth, D. Fox, and C. E. Rasmussen, "Gaussian processes for data-efficient learning in robotics and control," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, pp. 408–423, Feb 2015.
- [15] Y. Gal, R. McAllister, and C. E. Rasmussen, "Improving pilco with bayesian neural network dynamics models," in *Data-Efficient Machine Learning workshop, ICML*, vol. 4, 2016.
- [16] A. W. Salatian, K. Y. Yi, and Y. F. Zheng, "Reinforcement learning for a biped robot to climb sloping surfaces," *Journal of Robotic Systems*, vol. 14, no. 4, pp. 283–296, 1997.
- [17] M. Vukobratovic, A. Frank, and D. Juricic, "On the stability of biped locomotion," *Biomedical Engineering, IEEE Transactions on*, vol. BME-17, pp. 25 – 36, 02 1970.
- [18] T. McGeer, "Passive dynamic walking," *Int. J. Rob. Res.*, vol. 9, pp. 62–82, Mar. 1990.
- [19] M. Vukobratovic and B. Borovac, "Zero-moment point - thirty five years of its life.," *I. J. Humanoid Robotics*, vol. 1, pp. 157–173, 03 2004.

Interim Report on UESTC4006P(BEng) Final Year Project

-
- [20] H. Zhao, A. Hereid, W.-I. Ma, and A. D. Ames, "Corrigendum: Multi-contact bipedal robotic locomotion," *Robotica*, 2017.
- [21] U. Huzaifa, C. Maguire, and A. LaViers, "Toward an expressive bipedal robot: Variable gait synthesis and validation in a planar model," *CoRR*, vol. abs/1808.05594, 2018.
- [22] M. Shahbazi, G. A. D. Lopes, and R. Babu 拏 ka, "Observer-based postural balance control for humanoid robots," in *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 891–896, Dec 2013.
- [23] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Engelsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot atlas," *International Journal of Humanoid Robotics*, vol. 13, pp. 1650007–1, 03 2016.
- [24] P. Henaff, V. Scesa, F. Ouezdou, and O. Bruneau, "Real time implementation of ctrnn and bptt algorithm to learn on-line biped robot balance: Experiments on the standing posture," *Control Engineering Practice*, vol. 19, pp. 89–99, 01 2011.
- [25] A. Saputra, J. Botzheim, I. A. Sulistijono, and N. Kubota, "Biologically inspired control system for 3-d locomotion of a humanoid biped robot," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, pp. 1–14, 01 2015.
- [26] A. Xi, T. W. Mudiyansele, D. Tao, and C. Chen, "Balance control of a biped robot on a rotating platform based on efficient reinforcement learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, pp. 938–951, July 2019.
- [27] J. Lin, K. Hwang, W. Jiang, and Y. Chen, "Gait balance and acceleration of a biped robot based on q-learning," *IEEE Access*, vol. 4, pp. 2439–2449, 2016.
- [28] W. Wu and L. Gao, "Posture self-stabilizer of a biped robot based on training platform and reinforcement learning," *Robotics and Autonomous Systems*, vol. 98, pp. 42 – 55, 2017.
- [29] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2015.
- [30] K. Seo, J. Kim, and K. Roh, "Towards natural bipedal walking: Virtual gravity compensation and capture point control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4019–4026, Oct 2012.
- [31] M. P. Deisenroth, R. Calandra, A. Seyfarth, and J. Peters, "Toward fast policy search for learning legged locomotion," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1787–1792, Oct 2012.
- [32] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of honda humanoid robot," in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 2, pp. 1321–1326 vol.2, May 1998.
- [33] K. Zhang, Z. Hou, C. Silva, H. Yu, and C. Fu, "Teach biped robots to walk via gait principles and reinforcement learning with adversarial critics," 10 2019.
- [34] K. Nishiwaki and S. Kagami, "Strategies for adjusting the zmp reference trajectory for maintaining balance in humanoid walking," in *2010 IEEE International Conference on Robotics and Automation*, pp. 4230–4236, May 2010.
- [35] J. P. Ferreira, M. Crisostomo, and A. P. Coimbra, "Rejection of an external force in the sagittal plane applied on a biped robot using a neuro-fuzzy controller," in *2009 International Conference on Advanced Robotics*, pp. 1–6, June 2009.
- [36] T. Takenaka, T. Matsumoto, T. Yoshiike, T. Hasegawa, S. Shirokura, H. Kaneko, and A. Orita, "Real time motion generation and control for biped robot -4th report: Integrated balance control-," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1601–1608, Oct 2009.

Interim Report on UESTC4006P(BEng) Final Year Project

-
- [37] S. Gu, T. P. Lillicrap, Z. Ghahramani, R. E. Turner, and S. Levine, "Q-prop: Sample-efficient policy gradient with an off-policy critic," *CoRR*, vol. abs/1611.02247, 2016.
- [38] K. Hwang, W. Jiang, Y. Chen, and H. Shi, "Motion segmentation and balancing for a biped robot's imitation learning," *IEEE Transactions on Industrial Informatics*, vol. 13, pp. 1099–1108, June 2017.
- [39] B. Lee, D. Stonier, Y. Kim, J. Yoo, and J. Kim, "Modifiable walking pattern of a humanoid robot by using allowable zmp variation," *IEEE Transactions on Robotics*, vol. 24, pp. 917–925, Aug 2008.
- [40] R. Tedrake and H. S. Seung, "Learning to walk in 20 minutes," 2005.
- [41] C. Gil, H. Calvo, and H. Sossa, "Learning an efficient gait cycle of a biped robot based on reinforcement learning and artificial neural networks," *Applied Sciences*, vol. 9, p. 502, 02 2019.
- [42] I. Mordatch, N. Mishra, C. Eppner, and P. Abbeel, "Combining model-based policy search with online model learning for control of physical humanoids," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 242–248, May 2016.
- [43] K.-S. Hwang, J.-L. Lin, and J.-S. Li, "Biped balance control by reinforcement learning," *Journal of Information Science and Engineering*, vol. 32, pp. 1041–1060, 07 2016.
- [44] A. Rai, R. Antonova, S. Song, W. Martin, H. Geyer, and C. Atkeson, "Bayesian optimization using domain knowledge on the atias biped," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018.
- [45] R. Calandra, A. Seyfarth, J. Peters, and M. P. Deisenroth, "An experimental comparison of bayesian optimization for bipedal locomotion," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1951–1958, May 2014.
- [46] J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke, "Sim-to-real: Learning agile locomotion for quadruped robots," *Robotics: Science and Systems XIV*, Jun 2018.
- [47] N. Kohl and P. Stone, "Policy gradient reinforcement learning for fast quadrupedal locomotion," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 3, pp. 2619–2624 Vol.3, April 2004.
- [48] J. Zhu, S. Li, Z. Wang, and A. Rosendo, "Bayesian optimization of a quadruped robot during 3-dimensional locomotion," in *Biomimetic and Biohybrid Systems* (U. Martinez-Hernandez, V. Vouloutsis, A. Mura, M. Mangan, M. Asada, T. J. Prescott, and P. F. Verschure, eds.), (Cham), pp. 295–306, Springer International Publishing, 2019.
- [49] B. Bischoff, D. Nguyen-Tuong, H. van Hoof, A. McHutchon, C. E. Rasmussen, A. Knoll, J. Peters, and M. P. Deisenroth, "Policy search for learning robot control using sparse data," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3882–3887, May 2014.
- [50] M. P. Deisenroth, P. Englert, J. Peters, and D. Fox, "Multi-task policy search for robotics," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3876–3881, May 2014.
- [51] J. Ko, D. J. Klein, D. Fox, and D. Haehnel, "Gaussian processes and reinforcement learning for identification and control of an autonomous blimp," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 742–747, April 2007.
- [52] S. Wang, W. Chaovaitwongse, and R. Babuska, "Machine learning algorithms in bipedal robot control," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, pp. 728–743, Sep. 2012.
- [53] J. Kober, J. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, pp. 1238–1274, 09 2013.
- [54] A. Polydoros and L. Nalpantidis, "Survey of model-based reinforcement learning: Applications on robotics," *Journal of Intelligent & Robotic Systems*, vol. 86, pp. 153–, 01 2017.
- [55] S. Saeedvand, M. Jafari, H. Aghdasi, and J. Baltes, "A comprehensive survey on humanoid robot development," *The Knowledge Engineering Review*, vol. 34, pp. 1–18, 12 2019.