ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2.
Available from http://www.i6doc.com/en/.

Incorporating Human Priors into Deep Reinforcement Learning for Robotic Control

Manon Flageat, Kai Arulkumaran and Anil A. Bharath

Imperial College London - Dept. of Bioengineering
Exhibition Road, London - UK

Abstract. Deep reinforcement learning (DRL) shows promise for robotic control, as it scales to high-dimensional observations and does not require a model of the robot or environment. However, properties such as control continuity or movement smoothness, which are desirable for application in the real world, will not necessarily emerge from training on reward functions based purely on task success. Inspired by human neuromotor control and movement analysis literature, we define a modular set of costs that promote more efficient, human-like movement policies. Using a simulated 3-DoF manipulator robot, we demonstrate the benefits of these costs by incorporating them into the training of a model-free DRL algorithm and decision-time planning of a trained model-based DRL algorithm. We also quantify these benefits through metrics based on the same literature, which allows for greater interpretability of learned policies—a common concern when learning policies with powerful and complex function approximators.

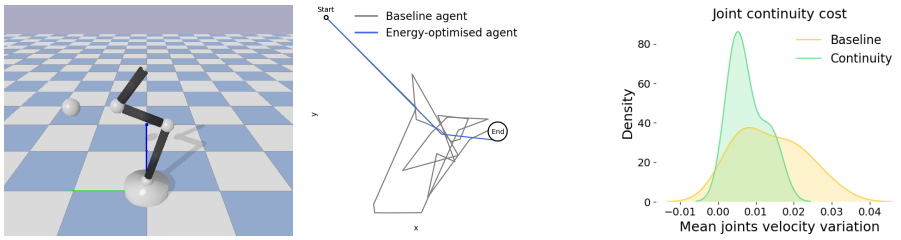
1 Introduction

Classic control strategies, which do not use learning, aim to find the optimal way to control a dynamical system to perform a task. A major limitation is that they require a model of the dynamics of the system. Instead, deep reinforcement learning (DRL) provides a scalable set of techniques allowing us to solve complex control problems without requiring an accurate dynamical model of the system, but simply through interaction with the environment. It therefore provides a way to solve problems with unknown dynamics, and is a popular strategy for finding controllers for robots even with high-dimensional observations [1].

However, DRL requires defining a reward function that can guide the learning towards optimising performance on a task. On top of rewarding task completion, additional costs can be incorporated to either aid learning or to promote finding policies with favourable properties [2]. We believe that a natural prior for costs can come from the human neuromotor control literature [3], as humans are able to accomplish a wide range of tasks, and, particularly after learning, in an efficient manner. In addition, drawing from the human movement analysis literature [4, 5] can give us more insights into the learned policies—a much-needed way of improving interpretability in DRL.

Thus, inspired by these "human priors", we propose a series of modular costs that can be applied in DRL for finding policies that produce more efficient and human-like movements, and demonstrate the results on a simulated manipulator robot (Fig. 1a). We apply these costs during the training of a model-free DRL

ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2. Available from http://www.i6doc.com/en/.



(a) Robot simulation (b) Trajectories (c) Continuity

Fig. 1: Robot, trajectories and analysis. (a) 3-DoF manipulator robot used for target reaching. (b) End-effector trajectories from a model-free agent trained with (blue) and without (grey) energy-saving costs. (c) Analysis of the joint continuity of model-free agent trained with continuity cost against baseline.

algorithm [6], and the decision-time planning of a model-based DRL algorithm [7], and show that the learned policy produces trajectories that are more suitable for application in the real world (Fig. 1b). Finally, we propose a set of performance metrics that allow us to quantify the effects and hence desirability of learned and planned policies (Fig. 1c).

2 Integrating human priors as costs function

2.1 Cost definition

We define multiple modular costs with the aim of influencing specific properties of movement, to promote finding a final policy more relevant for robotic control, while avoiding the trivial but overly conservative approach of constraining control and velocity values. Our final reward function is defined as: $r_t = r_t^{task} - \sum_{cost} w^{cost} r_t^{cost}$, with each w^{cost} as a hyperparameter and each r_t^{cost} scaled to [0, 1] using the loss function: $\mathcal{L}(x, a, b) = \begin{cases} (\frac{x-a}{b-a})^2 & \text{if } x \leq b - a \\ 1 & \text{otherwise.} \end{cases}$, where x is the property of interest and $[a, b]$ is the scaling interval.

Duration: The total duration of the movement can be penalised by scaling the elapsed time from the beginning of the movement to the total available time for one episode, T: $r_t^{duration} = \mathcal{L}(t, 0, T)$.

Energy: Assuming energy used is directly proportional to the intensity of the control put into the actuator, it can be penalised with: $r_t^{energy} = \mathcal{L}(\|\mathbf{a}_t\|, 0, 1)$.

Directness: An interesting property for efficient control is to have the trajectory be as direct as possible from the starting point to the point of interest [8]. In this work, we chose to define directness as the perpendicular distance between the end-effector and the direct path from the initial position to the target. The

ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2. Available from http://www.i6doc.com/en/.

error tolerance, δ_{target}, is the same as the one on the target for the task-specific reward: $r_t^{directness} = \mathcal{L}(\frac{\|(\mathbf{x}_t - \mathbf{x}^{ini}) \times (\mathbf{x}^{target} - \mathbf{x}^{ini})\|}{\|\mathbf{x}^{target} - \mathbf{x}^{ini}\|}, 0, \delta_{target})$.

Smoothness: Derivative continuity, or smoothness, of the end-effector position is a property often described as fundamental in human movement [9]. Here, we limit its variation to a sensible range using a hyperparameter $\delta_{velocity}$: $r_t^{smoothness} = \mathcal{L}(\|\mathbf{v}_t^{end} - \mathbf{v}_{t-1}^{end}\|, 0, \delta_{velocity})$.

Continuity: We can also define smoothness for the angular position of the joints/actuators; it is called "continuity" to distinguish it from the previous cost: $r_t^{continuity} = \mathcal{L}(\sum_{joint}(|\dot{\theta}_t^{joint} - \dot{\theta}_{t-1}^{joint}|), 0, N_{joints}.\delta_{velocity})$.

2.2 Model-free and model-based integration

These costs can trivially be integrated with model-free RL algorithms by adding them to the task reward function during training. The disadvantage is that this requires training a new agent for each cost combination or each value of the cost weight, and the trained policy will then be specific to these hyperparameters.

The costs can similarly be integrated with model-based RL agents by augmenting the task reward function during training. However, one of the main advantages of model-based algorithms is that they can be combined with decision-time planning/search techniques, making them more flexible after training. This allows us to overcome the limitation of model-free integration, as this alternative method does not require training separate models for each combination of costs or weight.

2.3 Performance

To quantify the performance and effects of these costs, we propose multiple metrics, directly inspired by human movement analysis. These measures are independent of the task conditions, and normalised for every target location. In contrast to the per-timestep cost functions, they are applied on full trajectories.

Normalised duration: The total duration is normalised by the direct path from the initial position to the target: $M_{duration} = \frac{T}{\|\mathbf{x}_{target} - \mathbf{x}_{ini}\|}$.

End-effector directness: Directness is quantified using the direct analogue of the directness cost: $M_{directness_1} = mean_t(\frac{\|(\mathbf{x}_t - \mathbf{x}_{ini}) \times (\mathbf{x}_{target} - \mathbf{x}_{ini})\|}{\|\mathbf{x}_{target} - \mathbf{x}_{ini}\|})$, but also by comparing the length of the end-effector trajectory and the length of the direct path [8]: $M_{directness_2} = \sum_{t=0}^{T} \frac{\|\mathbf{x}_t - \mathbf{x}_{t-1}\|}{\|\mathbf{x}_{target} - \mathbf{x}_{ini}\|}$.

End-effector smoothness: Similarly, smoothness is quantified using the metric directly related to the cost: $M_{smoothness_1} = mean_t(\|\mathbf{v}_t^{end} - \mathbf{v}_{t-1}^{end}\|)$, and three other quantifications defined in the human movement analysis literature [10]: the number of peaks in the velocity profile: $M_{smoothness_2} = \#\mathbf{v}_{maxima}$, the normalised jerk variation: $M_{smoothness_3} = \frac{T^2}{\max_t(\mathbf{v}_t)} \int_0^T \|\ddot{\mathbf{v}}\|$, and the spectral arc length: $M_{smoothness_4} = \int_0^{\omega_c} \sqrt{\left(\frac{1}{\omega_c}\right)^2 + \left(\frac{\dot{V(\omega)}}{V(0)}\right)^2} d\omega$, where $V(\omega)$ is the Fourier magnitude spectrum of $v(t)$, and $[0, \omega_c]$ is the frequency band of the movement.

ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2.
Available from http://www.i6doc.com/en/.

Joint smoothness: We define the following metric, which captures the variation in joint velocities: $M_{joint\,smoothness} = \text{mean}_t(|\dot{\theta}_t^{joint} - \dot{\theta}_{t-1}^{joint}|)$.
Control performance: Other important properties are the control values themselves, and their continuity. Thus we define two other metrics:
$M_{control\,range} = \text{mean}_t(\|\mathbf{a}_t\|)$, and $M_{control\,continuity} = \text{mean}_t(\|\mathbf{a}_t - \mathbf{a}_{t-1}\|)$.
Human reference values: These metrics can only be used relatively, to compare different policies. However, we can anchor these values using the performance of a random policy, and, where available, human scores (on reaching tasks) taken from the human movement analysis literature [4, 8, 10].

3 Experiments

We evaluate our methods on simulated robotic tasks in order to demonstrate their advantage for robotic control and the generation of human-like movements. We use soft-actor critic (SAC) [6] as our model-free DRL algorithm and PlaNet [7] as our model-based DRL algorithm. All networks are 2-layer MLPs with ReLU nonlinearities, with a hidden size of 128 for SAC and 200 for PlaNet.

3.1 Experimental setup

We built our own simulated robotics environment for our experiments, to perform reaching tasks with manipulator robots, using PyBullet[1].
Robots: As kinematic and kinetic redundancy is one of the main problems faced when controlling a robotic arm, the DoF of our robot is chosen to maintain this. Thus, we use a 3-DoF manipulator robot with self-collision enabled (Fig. 1), with targets uniformly placed within reaching distance in a 2D plane.
State and action spaces: We design our observations to include all information needed to make the process Markovian for the integration of any of our costs: normalised time t, sin and cos of angular position θ_{joint}, and angular velocity $\dot{\theta}_{joint}$, of each joint, initial position $\mathbf{x}_{ini}$, current position $\mathbf{x}_t$ and velocity $\dot{\mathbf{x}}_t$ of the end-effector, position of the target $\mathbf{x}_{target}$, and actions $\mathbf{a}$. We use either torque or velocity control, applied to the different joints of the robot. The size of the action space is the number of DoF of the robot.
Tasks and reward function: We designed reaching tasks in 2D, with an equipotential distance-based dense reward function. An episode is considered to be successful when the end-effector of the robot reaches the target, within a tolerance of δ_{target}, and will always end with a total reward of 1. Our task reward function is: $r_t^{task} = \frac{\|\mathbf{x}_{t-1} - \mathbf{x}_{target}\| - \max(\|\mathbf{x}_t - \mathbf{x}_{target}\|, \delta_{target})}{\|\mathbf{x}_{target} - \mathbf{x}_{ini}\| - \delta_{target}}$.
Evaluation: All methods are first evaluated on their learning performance, generalisation and convergence speed. However, the effect of the different costs and the comparison between the policies is carried out using the previously defined performance measures.

[1] https://pybullet.org

ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2. Available from http://www.i6doc.com/en/.

3.2 Results for Model-free integration

We first integrate the costs in the model-free DRL algorithm during training. We tried cost weights $w^{cost} \in \{0.01, 0.1, 0.5\}$; all agents learned successful policies with the effects of the costs proportional to their weights. The results in Table 1 shows the range and the mean of the scores obtain by each agent for our grid of test targets with $w^{cost} = 0.1$. Given that multiple costs aim to restrain the amplitude of some metrics, the range of values give important information on their effectiveness. All agents trained with additive costs improve over the baseline on all performance measures, moving towards human scores. In addition, some agents outperformed others on their corresponding metrics; for example, the agents trained to minimise the energy used in the actuators outperformed the duration-cost agent on the normalised duration metric, which shows that minimising the total energy used for a movement can be achieved by minimising the time spent per unit of distance.

Table 1: Performance, according to our metrics, of model-free agents trained with each cost individually. Human and random trajectory values are included for reference. [ ] denotes the range of the values and $\sim$ their mean.

Agent	**Duration**	**End-effector directness**	
	Normalised duration	Normalised path len.	Dist. to direct path
Human	0.36	1.2	0.12
Random	$[0.20, 33.7] \sim 3.62$	$[2.08, 133.1] \sim 14.5$	$[0, 2.07] \sim 0.71$
Baseline	$[0.11, 10.1] \sim 1.03$	$[1.18, 14.8] \sim 3.82$	$[0, 0.55] \sim 0.18$
Duration	$[0.11, 2.44] \sim 0.49$	$[1.18, 15.4] \sim 3.53$	$[0, 0.42] \sim 0.16$
Directness	$[0.11, 5.38] \sim 0.50$	$[1.18, 12.5] \sim \mathbf{2.96}$	$[0, 0.49] \sim \mathbf{0.13}$
Energy	$[0.11, 1.76] \sim \mathbf{0.47}$	$[1.18, 11.8] \sim 2.97$	$[0, 0.41] \sim 0.14$
Smoothness	$[0.11, 2.46] \sim 0.58$	$[1.13, 13.6] \sim 3.44$	$[0, 0.77] \sim 0.18$
Continuity	$[0.11, 1.76] \sim 0.50$	$[1.18, 12.6] \sim 3.22$	$[0, 0.45] \sim 0.14$

Agent	**Joint smoothness**	**Control energy**	
	Joint variation	Control mean	Control variation
Human	—	—	—
Random	$[0.97, 559.5] \sim 13.1$	$[0.24, 38.9] \sim 9.03$	$[0.49, 5.05] \sim 1.23$
Baseline	$[0.33, 6.93] \sim 1.59$	$[0.16, 16.5] \sim 1.38$	$[0.33, 3.23] \sim 1.52$
Duration	$[0.20, 4.09] \sim 1.31$	$[0.14, 2.83] \sim 0.74$	$[0.33, 3.23] \sim 1.01$
Directness	$[0.31, 3.31] \sim 1.11$	$[0.10, 2.22] \sim 0.49$	$[0.20, 3.16] \sim 0.77$
Energy	$[0.24, 6.16] \sim 1.16$	$[0.12, 3.23] \sim \mathbf{0.52}$	$[0.24, 3.35] \sim \mathbf{0.73}$
Smoothness	$[0.24, 3.98] \sim 1.14$	$[0.14, 3.04] \sim 0.69$	$[0.29, 3.29] \sim 0.97$
Continuity	$[0.33, 3.32] \sim \mathbf{1.13}$	$[0.14, 2.09] \sim 0.58$	$[0.28, 2.58] \sim 0.84$

Agent	**End-effector smoothness**			
	Velocity variation	Number peaks	Jerk variation	Spectral arc len.
Human	—	1.25	0.018	2.4
Random	$[0.59, 14] \sim 3.19$	$[1, 48] \sim 11.19$	$[0, 3237] \sim 117.7$	$[2, 17.5] \sim 8.43$
Baseline	$[0.21, 4.11] \sim 1.01$	$[0, 47] \sim 3.9$	$[0, 800] \sim 16.1$	$[2, 19.9] \sim 6.59$
Duration	$[0.22, 2.85] \sim 0.80$	$[0, 6] \sim 2.3$	$[0, 1.5] \sim 0.06$	$[2, 16.3] \sim 5.81$
Directness	$[0.21, 2.04] \sim 0.84$	$[0, 8] \sim 2.1$	$[0, 0.9] \sim 0.02$	$[2, 13.9] \sim 4.67$
Energy	$[0.20, 3.02] \sim 0.81$	$[0, 8] \sim 1.9$	$[0, 1.8] \sim 0.04$	$[2, 15.2] \sim 4.54$
Smoothness	$[0.22, 1.47] \sim \mathbf{0.73}$	$[0, 7] \sim \mathbf{1.7}$	$[0, 1.0] \sim 0.05$	$[2, 17.1] \sim \mathbf{4.30}$
Continuity	$[0.22, 1.52] \sim 0.76$	$[0, 6] \sim 1.8$	$[0, 0.5] \sim \mathbf{0.02}$	$[2, 12.4] \sim 4.75$

3.3 Results for Model-based integration

We trained a model-based DRL agent with only the task-related reward, and introduce the cost during decision-time planning. The effect of the costs are less pronounced than with model-free integration. Overall, every cost added in the planner shows improvement over the baseline on the metrics related to the cost it is optimising, but not necessarily on every other cost. The smoothness and duration results indicate that the model-based strategy to solve the task is very different from the model-free one: trajectories tend to be much longer

ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2. Available from http://www.i6doc.com/en/.

and include many small movements around the final position. These results can potentially be explained by the choice of planning horizon for the planner. Indeed, most of the costs are defined on the value over time of a given quantity, which can be hard to optimise with a short planning horizon. This explanation is consistent with the fact that planning with the duration cost results in exactly the same trajectories and same performance as the baseline, which indicates the inability of the planner to take time into account. Unfortunately, increasing the planning horizon incurs additional computational costs.

4 Discussion and Conclusion

This work concentrates on the integration of neuromotor-control-inspired costs in DRL algorithms for robotic control, and additionally proposes movement-analysis-inspired metrics for control-oriented interpretability of learned policies. The results show that the learned policies, especially for model-free agents, can be significantly improved from a real-world application point of view, by integrating our modular costs, with performance approaching human reference scores on some metrics. An interesting outcome is quantifying the interdependency of these costs, as the usage of only one of them can lead to better performance according to all metrics. Future work could concentrate on using models with improved long-term prediction in order to obtain more relevant results with decision-time cost integration, as this is the more flexible approach.

References

[1] Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. A Brief Survey of Deep Reinforcement Learning. *IEEE Signal Processing Magazine*, 34(6):26–38, November 2017.

[2] Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

[3] Emanuel Todorov. Optimality principles in sensorimotor control. *Nature Neuroscience*, 7(9):907–915, September 2004.

[4] P. Morasso. Spatial control of arm movements. *Experimental Brain Research*, 42(2), April 1981.

[5] James Gordon, Maria Felice Ghilardi, Scott E. Cooper, and Claude Ghez. Accuracy of planar reaching movements. *Experimental Brain Research*, 99(1):112–130, May 1994.

[6] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. *arXiv:1801.01290 [cs, stat]*, January 2018.

[7] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning Latent Dynamics for Planning from Pixels. *arXiv:1811.04551 [cs, stat]*, November 2018.

[8] Claes von Hofsten. Structuring of Early Reaching Movements: A Longitudinal Study. *Journal of Motor Behavior*, 23(4):280–292, December 1991.

[9] T Flash and N Hogan. The coordination of arm movements: an experimentally confirmed mathematical model. *The Journal of Neuroscience*, 5(7):1688–1703, July 1985.

[10] S. Balasubramanian, A. Melendez-Calderon, and E. Burdet. A Robust and Sensitive Metric for Quantifying Movement Smoothness. *IEEE Transactions on Biomedical Engineering*, 59(8):2126–2136, August 2012.