Learning and adapting quadruped gaits with the "Intelligent Trial & Error" algorithm

Eloïse Dalin, Pierre Desreumaux, Jean-Baptiste Mouret

Learning and adapting quadruped gaits with the "Intelligent Trial & Error" algorithm

Eloïse Dalin[1], Pierre Desreumaux[1] and Jean-Baptiste Mouret[1]

I. INTRODUCTION

Most reinforcement learning algorithms require thousands of training episodes to find an effective policy, which is often infeasible with a physical legged robot [1]. As a result, locomotion controllers are usually trained in simulation, then transferred to the real robot [2], [3]. Unfortunately, the policies optimized in simulation are likely to exploit the inaccuracies of the simulator and, consequently, to not perform well on the real robot; this phenomenon is called *the reality gap* [4]. The typical approaches to cross this reality gap are to improve the simulator and to (2) encourage the robustness of the policy with techniques like domain randomization [3], [5], [2].

The "Intelligent Trial and Error" (IT&E) follows a different idea [6] (Fig.1): in simulation, it searches in the high-dimensional policy parameter space for thousands of different but high-performing ways of achieving the task (i.e., walking), then it stores them in a "map"; on the real robot, it searches for the most adapted policy in this low-dimensional behavior map, thanks to a Bayesian optimization procedure that uses the simulation results as a prior. The underlying assumption is that among the thousands of different ways of performing the task, some of them will cross the reality gap better. In a series of experiments with a 6-legged robot, the robot was able to discover a gait in less than a dozen of trials in spite of large reality gaps: one missing leg, two missing leg, or one shortened leg [6]. A recent extension of IT&E allows it to automatically select the best prior among many possible (different environments or different physical characteristics) [7].

Here we report our experiments in using the IT&E algorithm to learn and adapt gaits on the Minitaur quadruped robot, which was used by several other teams for similar learning experiments [3], [8].

II. METHODS

The policy is an open-loop position controller in Cartesian space: the Cartesian position of each foot is controlled by smoothed pulse waves, each defined by 3 parameters — amplitude, phase, and duty cycle. Since there are 4 legs and 2 dimensions (vertical and horizontal), there are $3 \times 4 \times 2 = 24$ parameters to learn. Please note that as the parameter space is only used in simulation, much larger parameter space can be used if necessary.

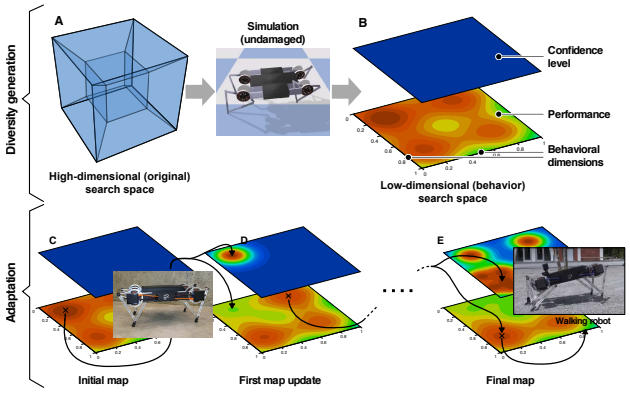

Fig. 1. **Concept of the IT&E algorithm [6]**. The first step occurs in simulation and aims at learning simultaneously thousands of high-performing gaits according to behavior dimensions. When an adaptation is needed on the real robot, a Bayesian optimization procedure searches in this map for the best gait. Picture adapted from [6].

The first step of the IT&E algorithm is to generate a diverse set of high-performing gaits in simulation using the MAP-Elites algorithm [9]. We use the pybullet Minitaur simulation from [3], but we did not use the motor model proposed in [3] (there is therefore a larger reality gap).

For the Minitaur, we chose to distinguish gaits by how they are using the torque during a gait cycle. Each 1-second gait cycle is divided in 4 sections. For each cycle section, the mean torque per leg is computed, normalized in $[0,1]$ (1 is the maximum motor torque in simulation), and stored. Each gait is therefore described in a 16-dimensional behavior space. This behavior space has been chosen in order to be able to have a wide diversity of walking gaits.

This behavior space is divided in 40,000 cells of equal volume using a Centroidal Voronoi Tesselation [10]. The performance function is the covered distance before exiting a 1-meter wide corridor or after 10 seconds. Using the behavior description and the performance function, MAP-Elites simultaneously searches for the best gait in each behavioral cell. We ran MAP-Elites 10 times to get 10 independent maps. At the end of the evolutionary process, each map contains about 16,000 high-performing policies (depending on the map) organized in a 16-dimensional behavior space.

When an adaption is needed (e.g., after a damage or to cross the reality gap), IT&E models the performance of each policy on the real robot in the behavior space using Gaussian Processes [11] whose mean function is the performance

[1]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
firstname.name@inria.fr

This work received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (GA no. 637972, project "ResiBots").

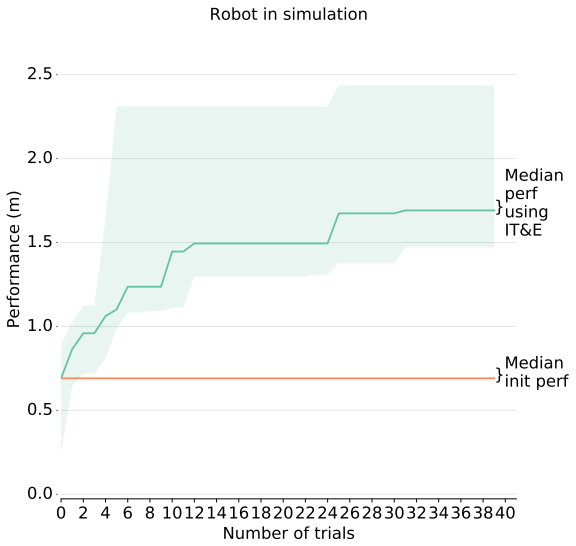

Fig. 2. **Best performance (covered distance) for each trial on the damaged robot in simulation**. The lines represent the median and the colored areas the 25^{th} and 75^{th} percentiles. The green color corresponds to the IT&E output performance. The orange line corresponds to the initial median performance, testing the best behaviors found in simulation. Each behavior is tested on a 10 seconds episode.

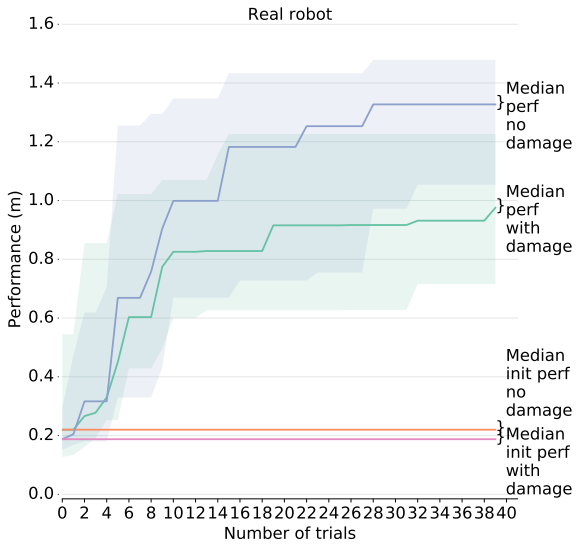

Fig. 3. **Best performance (covered distance) for each trial on the real robot.** The lines represent the median and the colored areas the 25^{th} and 75^{th} percentiles. The blue and green colors correspond to the IT&E output performance, respectively without and with damage. The orange and pink colors correspond to the initial median performances, respectively without and with damage. Initially IT&E tests the best behaviors found in simulation. Each behavior is tested on a 10 seconds episode.

predicted by the simulation. After each test, the GP is updated for each point of the map and the best candidate is selected using the Upper Confidence Bound acquisition function [12].

III. RESULTS

We first studied whether IT&E allows the Minitaur to recover from damage in simulation: we blocked the left front leg in a fully retracted and perpendicular to the ground position. We tested IT&E on 10 independently generated maps. The trials duration is of 10s. The results show that IT&E finds in 5 trials behaviors with a median performance of 1m (Fig. 2), which is comparable to what was found in previous experiments with a 6-legged robot [6].

We then tested IT&E with the same maps but on the real Minitaur. Without damage, IT&E finds effective gaits with a median performance of 0.9m in 10 trials (9 cm/s, Fig. 3). The low performance at the first iteration shows the need to adapt to the reality gap: the best controller found in simulation has a median performance of 0.18m. With the same damage as in simulation, IT&E finds effective gaits with a median performance of 0.9m in in 20 trials (Fig. 3). Video: `https://youtu.be/v90CWJ_HsnM`

IV. CONCLUSION

Overall, the results with the Minitaur are consistent with those previously obtained with our 6-legged robot [6], [7]: 10 to 20 trials are enough to adapt to damage and to cross the reality gap. In future work, we will evaluate the automatic choice of the prior map [7].

The gaits found are not the fastest known for the Minitaur, but (1) the simulation model is not very accurate, and (2) to save the motors, we cut the power as soon as the motors use more than 25A during 1 seconds, which prevents many of the fast but highly dynamic gaits.

REFERENCES

[1] K. Chatzilygeroudis, V. Vassiliades, F. Stulp, S. Calinon, and J.-B. Mouret, "A survey on policy search algorithms for learning robot controllers in a handful of trials," *arXiv:1807.02303*, 2018.

[2] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, 2019.

[3] 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," in *Proceedings of RSS*, 2018.

[4] S. Koos, J.-B. Mouret, and S. Doncieux, "The transferability approach: Crossing the reality gap in evolutionary robotics," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 1, pp. 122–145, 2013.

[5] K. Chatzilygeroudis and J.-B. Mouret, "Using parameterized black-box priors to scale up model-based policy search for robotics," in *Proc. of IEEE ICRA*, 2018.

[6] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret, "Robots that can adapt like animals," *Nature*, vol. 521, no. 7553, p. 503, 2015.

[7] R. Pautrat, K. Chatzilygeroudis, and J.-B. Mouret, "Bayesian optimization with automatic prior selection for data-efficient direct policy search," in *Proc. of IEEE ICRA*. IEEE, 2018, pp. 7571–7578.

[8] T. Haarnoja, A. Zhou, S. Ha, J. Tan, G. Tucker, and S. Levine, "Learning to walk via deep reinforcement learning," *arXiv:1812.11103*, 2018.

[9] J.-B. Mouret and J. Clune, "Illuminating search spaces by mapping elites," *arXiv:1504.04909*, 2015.

[10] V. Vassiliades, K. Chatzilygeroudis, and J.-B. Mouret, "Using centroidal Voronoi tessellations to scale up the multidimensional archive of phenotypic elites algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 4, pp. 623–630, 2018.

[11] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT Press Cambridge, MA, 2006, vol. 2, no. 3.

[12] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2016.