Lessons on the Reality-Gap: Iterations between Virtual and Real Robots

Andre Rosendo
Living Machine Laboratory, ShanghaiTech University
Shanghai,201210, China

Charlie Houseago, Fumiya Iida
Bio-inspired Robotics Laboratory, University of Cambridge
Cambridge, CB21PZ, United Kingdom
E-mail: arosendo@shanghaitech.edu.cn,charliehouseago@gmail.com, fi224@cam.ac.uk
www.shanghaitech.edu.cn

Abstract

Due to approximations between the virtual and real world, the knowledge transfer from simulations to robots is problematic. As physical conditions are prone to unknown and stochastic noise sources, the predictability reduces. We use data experiments from 100 real robots to tune the parameters of a simulation, and later used this tuned simulator to improve the design of the previous robots and find the optimum robot. We compare the simulated and observed behavior of this robot, and discuss our results.

Keywords: Reality gap, evolutionary robotics, virtual robots, design optimization, body-mind co-optimization.

1. Introduction

The major obstacle of researchers creating a bridge between computer simulations of robots and the behavior seen in the real world is termed the 'reality-gap' problem [1]. Although simulations can be helpful to predict behaviors, a few researchers in robotic simulation community have repeatedly warned of the dangers of using over-simplified tools to simulate the behavior of real-world robotic agents [2]. A simulation by its very nature involves a number of levels of abstraction from true physical behavior, and thus fundamentally differ from the real robots they are attempting to emulate.

One common way of overcoming the reality gap for design of controllers is to use the notion of transferability, whereby it is hypothesized that different controllers will correspond better across the reality gap than others. The work of Jakobi [3] stressed the use of minimal simulations, which targeted areas of reality that would insure robust, transferable behavior, but is therefore limiting in terms of broad applicability.

One successful approach to handling the reality-gap problem for control policy evolution is the transferability function, most notably employed in [4]. Under this framework, an additional measure aside from the objective function is employed which assigns a transferability to each potential control policy based on a small number of experiments in reality. They were able to demonstrate success in producing a transferability function using only 10 evaluations from the hardware in reality. Alternatively, rather than attempting to bridge the reality-gap for co-evolution of structure and control policy, the work conducted in [5] evolved 200 robots in the real world using an automated assembly and evaluation process. This consisted of a UR5 robotic arm, and a number of cubic modules, which were attached together using hot-melt adhesive. This experiment demonstrated the success of a real-world evolutionary process which co-evolved both the physical structure and the control policy of a population of robotic agents.

In here we use the training data of experiments performed in the aforementioned experimental setting [6] to tune the parameters governing physical interaction in simulation. In this data-driven approach, we design our simulation environment to create a stronger correlation with the real-world and eventually achieve better correlation between reality and simulation. In addition, we wish to demonstrate that the relationship between simulation and reality is dependent both on the morphology and the control, and thus there is no universal transferability function that is independent of the morphology.

2. Methods

In the first stage of work, a simulation environment is designed to conduct virtual experiments similar to the ones performed at [6], and the output of which can be changed with different floor parameters which describe the response of the environment. Then we perform a Principal Component Analysis (PCA) of the results from reality with those of simulation to evaluate how close the behaviors are. A training process is then applied to choose the next set of parameters, and the process repeated. The output of the process is the choice of parameters which gives the smallest reality-gap, i.e. those for which the results from reality and from simulation are closest together.

2.1. *Simulator and virtual model*

A plentitude of physics simulators exists, and for this work we oriented our choices with the references [7], and we eventually chose Bullet to recreate virtually experiments that we performed a few years ago [6]. Bullet handles rigid collisions with greater accuracy, and we avoid the main limitation of the environment – poor damping behavior - due to the simplistic nature of the agents involved in this experiment.

The process was performed in a similar way to the aforementioned model-free morphology experiment. The genome structure comprises of 10 numeric values per cube (control and morphology), with one additional parameter per robot which defines the total number of cubes. Specifics of these parameters can be found at [6]. The comparison between the baseline real-world experiments and the simulated structure can be seen in Fig.1.

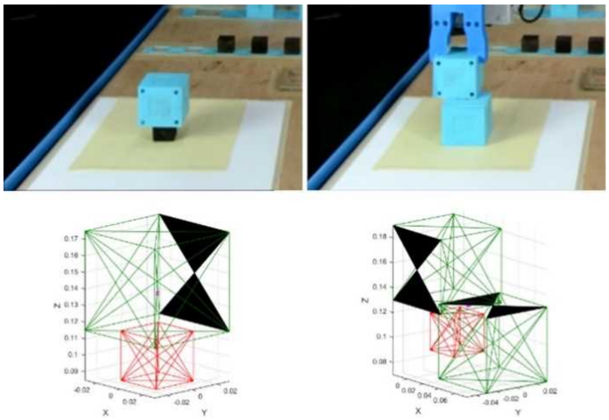

Fig. 1. Comparison of the real-world construction (above) and the simulation created from these constructions (below).

2.2. *PCA of simulator with reality*

In addition to comparing agents between reality and simulation based on fitness, we can use a more advanced method of matching agents based on the entire trajectory travelled. A number of different methods of performing 2D trajectory comparison exist, and there is an existing body of research into their applications in video motion tracking. As an example, in [8], the Hausdorff distance is used to express the spatial similarity between two trajectories.

In here, we employ an approach developed in [9], based on representing a trajectory as a cluster of data points using it's PCA coefficients. Given the relatively short length of trajectories in our experiment, and the large number of evaluations, we will work on the entire trajectory as one data cluster. The procedure while applying to a simulated trajectory consists of:

1. Subsample simulated trajectory to 182 data points
2. Translate both trajectories to start at the origin
3. Rotate trajectories such that their vector is aligned
4. Compute principal components and compare

As it can be seen in Fig. 2 and 3, the main trajectory of the real-world experiment depicted in Fig. 2 is compared with nine simulated trajectories from Fig. 3. The simulated behaviors from Fig. 3 have different floor parameters, simulating different friction and restitution coefficients. The number above each of these nine trajectories indicate the PCA discrepancy to the real-world result. The simulations from the middle row of

Fig.3 better approximate the real-world behavior, as not only the first principal component (fitness) is closer, but the shape of the trajectory curve is closer (better match with second and subsequent components.

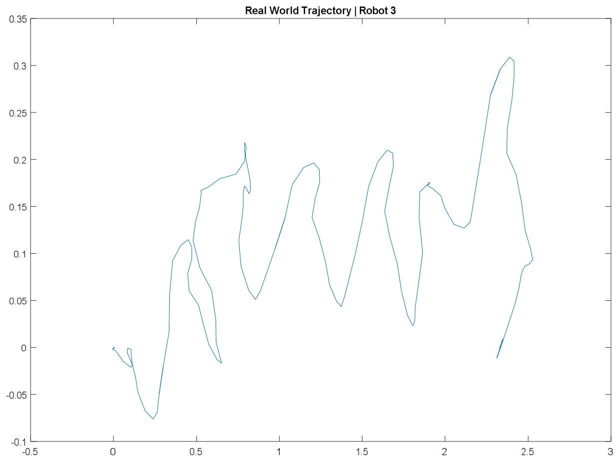

Fig. 2. Trajectory observed at the real-world experiment. The initial position is at (0,0), and distances are in centimeters.

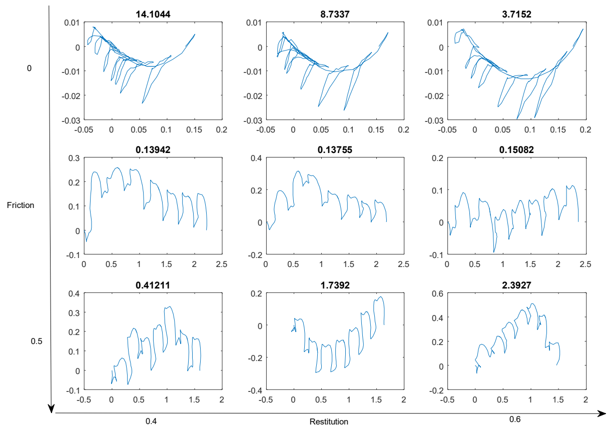

Fig. 3. Simulated trajectories with different friction and restitution coefficients. The number above each of these nine graphs stands for the difference between this particular case and the real-world behavior.

2.3. *Optimizing parameters*

The interaction between the agent and the ground is very complicated, and increasing the friction not only increases the impulse given to the agent by collisions of the rotating face with the terrain, but also increases the drag on the body as the agent moves itself along the ground. Changes in the restitution impact the time that an agents body spends in contact with the terrain, and thus directly impact the response due to friction.

In Fig. 4 we show a heat map of the friction/restitution difference between the simulated fitness and the real world fitness for the robot morphology used to create the trajectory from Fig. 2. This data was collected at 441 evenly spaced points across the specified parameter ranges. The plot displays much steeper gradients in regions of high friction and restitution, as the response to changing either of the parameters becomes more severe. The shape of this plot is dependent on the robot morphology, in this case, the adopted robot moves with three active faces in contact with the ground surface. Depending on the physical values, different servos can be in contact with the ground at different parts of the motion cycle.

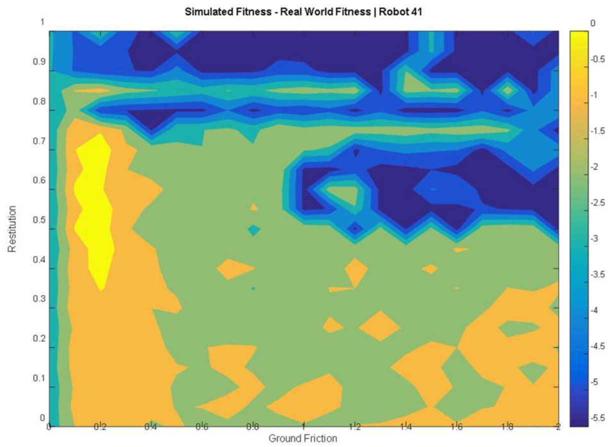

Fig. 4. Heat map of the difference between Simulated and Real behavior for changes in friction and restitution. The yellow area shows a promising region, where those parametric choices translate into a faithful reproduction in reality of the simulated trajectory.

3. Results and Discussion

With an optimized simulator in our hands we decided to conduct a Grid Search through the possible morphological and control combinations to create the best performing robot. The robot depicted in Fig. 5

showed the best behavior, and it was build and tested in the real world.

Fig. 5. Picture of the best predicted morphology, to which the behavior is finally compared with the simulated behavior.

In Fig. 6 we present the final outcome of this work. The comparison between the simulated trajectory and the real-world trajectory can be seen. Although both trajectories present a strong resemblance in trajectory shape, the simulated behavior was apparently a scaled down version of the real world output. One additional, and extremely important caveat: This resemblance in trajectory shape was only possible in 15% of the trials, as the robot followed different trajectories in other 60% of the cases, and in the last 25% didn't translate at the testing platform (solely rotation).

The previous result brings in question the deterministic approach to the Reality Gap: if the stochastic influence is such that a predicted behavior is rarely occurring, probabilistic approaches, with means and standard deviations of fitness values, might be a better solution for this problem. Although a strong methodology was adopted, in this work we failed to predict the behavior of our proposed robot, and we believe that this work will guide others in search of a similar answer.

4. References

1. N. Jakobi, P. Husbands and Harvey I, "Noise and the Reality Gap: The use of simulation in Evolutionary Robotics," in *Advances in Artificial Life: Proceedings of the 3rd European Conference on Artificial Life*, 1995.
2. R. A. Brooks, "Artificial Life and Real Robots," in *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, 1992.
3. N. Jakobi, "Half-Baked, Ad-hoc and Noisy: Minimal Simulations for Evolutionary Robotics," in *Fourth European Conference on Artificial Life*, 1997.
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, pp. 122-145, 2013.
5. V. Vujovic, A. Rosendo, L. Brodbeck, F. Iida, "Evolutionary developmental robotics: Improving locomotion of physical robots." Artificial Life 23, 2 (2017).
6. L. Brodbeck, S. Hauser and F. Iida, "Morphological Evolution of Physical Robots through Model-Free Phenotype Development," *PLOS One,* 2015.
7. A. Boeing and T. Braunl, "Evaluation of real-time physics simulation systems," *Graphite '07,* pp. 281-288, 2007.
8. I. Junejo, O. Javed and M. Shah, "Multi-Feature Path Modelling for Video Surveillance," *Proceedings 17th international Conference on Pattern Recognition,* 2004.
9. F. I. Bashir, A. A. Khokhar and D. Schonfeld, "Segmented trajectory based indexing and retrieval of video data," *Proceedings International conference on Image Processing,* 2003.

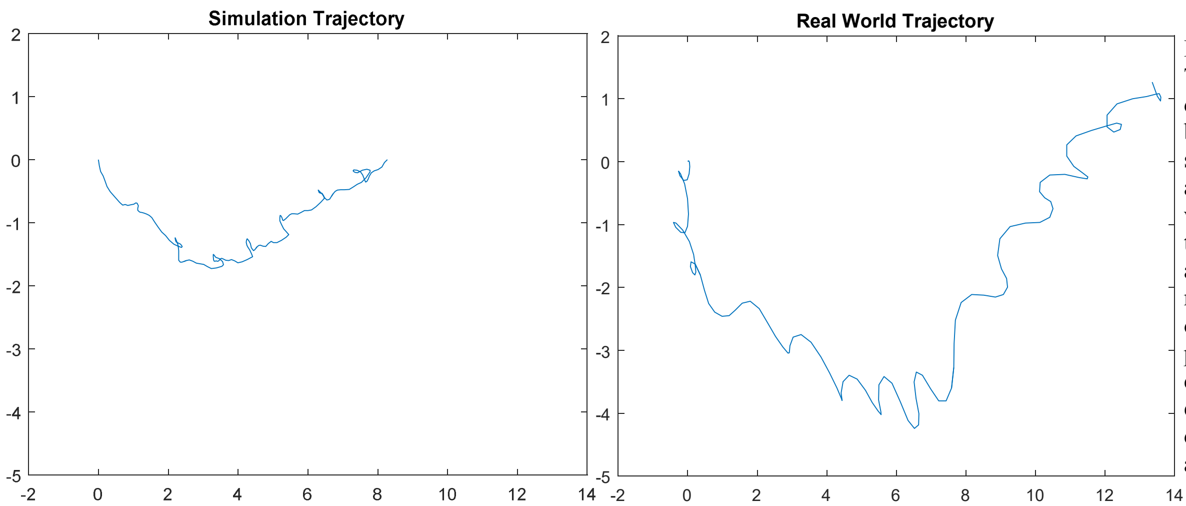

Fig. 6. The difference between simulated and real-world trajectories, and the low repeatability of the results puts in question the efficacy of deterministic approaches.