Sim2Real in Robotics and Automation: Applications and Challenges

Sim2Real in Robotics and Automation: Applications and Challenges

Sebastian Höfer[*,1], Kostas Bekris[1,2], Ankur Handa[3], Juan Camilo Gamboa[4], Florian Golemo[5,6], Melissa Mozifian[4], Chris Atkeson[7], Dieter Fox[3,8], Ken Goldberg[9], John Leonard[10], C. Karen Liu[11], Jan Peters[12,13], Shuran Song[14], Peter Welinder[15], Martha White[16]

To perform reliably and consistently over sustained periods of time, large-scale automation critically relies on computer simulation. Simulation allows us and supervisory AI to effectively design, validate and continuously improve complex processes, helps practitioners to gain insight into operation and justify future investments. While numerous successful applications of simulation in industry exist, such as circuit simulation, finite element methods and computer-aided design (CAD), state-of-the-art simulators fall short of accurately modeling physical phenomena, such as friction, impact and deformation.

Recently, the research community has doubled down on efforts to tackle these problems by combining physical simulation with deep learning. These two techniques can be regarded as complementary since deep learning enables training from large quantities of data, which in turn can be produced efficiently in simulation. Furthermore, learning in simulation can be faster than real-time, cheaper, safer, and more informative by providing perfect ground truth labels.

The key challenge in combining learning and simulation is the ability to transfer predictive models and control programs acquired in simulation directly to the real world, a concept termed *Sim2Real* transfer. Recent work in Sim2Real has shown promising results on real-world problems related to autonomous driving, grasping, or in-hand manipulation with policies trained in simulation only. Nevertheless, Sim2Real still faces significant challenges. In particular, it remains an open question to what extent and in which problem domains Sim2Real can compete with or outperform techniques based on real-world experimentation.

This editorial highlights opportunities and challenges related to Sim2Real in automation. It is based on a Sim2Real workshop held in conjunction with the 2020 edition of the "Robotics: Science and System" conference, which is summarized in a more detailed accompanying report [2].

Current State of Sim2Real. On-going research in Sim2Real is approaching the problem from multiple directions, which can be broadly clustered into the following categories: (i) formalizing Sim2Real and developing metrics for comparing Sim2Real solutions, (ii) achieving Sim2Real by leveraging existing simulators, (iii) empirically studying the reality gap of existing simulators, (iv) increasing simulation accuracy in novel ways.

Given this very diverse nature of what constitutes Sim2Real research, it is important to explore the need for, the nature of and the way to approach Sim2Real.

Why should we invest in Sim2Real? On the one hand, simulation is significantly less expensive than real-world experimentation as it is not necessary to build and maintain real systems. These allows for lowering the entry barrier for students and researchers. Furthermore, experimentation in Sim2Real is inherently safe and thus useful for dangerous tasks and early exploration. Simulation can also provide huge amounts of diverse, labeled data using techniques like domain randomization.

On the other hand, some popular examples of Sim2Real transfer, such as OpenAI's Rubik's cube, required over a year of development to successfully transfer controllers [3]. The rise of out-of-the-box simulation environments such as OpenAI gym can also lead to a rise of simulation-only research that does not generalize to the real world, a trend of *Sim2Null* research. At this point, controllers trained in physical simulation environments, such as the OpenAI gym generalize poorly, if at all, to the real world. Furthermore, successful Sim2Real transfer often involves a large amount of manual tuning of black-box parameters so as to generalize to new tasks, in contrast to rigorous modeling and problem understanding [1]. In some domains, real-world data collection can generate more trustworthy and relevant data. It is also not clear that successful simulation is necessary for effective real-world performance, since overly simplified world modeling may render the simulated problem harder than the real one. Furthermore, the safety guaranteed by simulation may reduce the focus from making robots safer, such as through the introduction of soft materials.

What is Sim2Real? It can be argued that Sim2Real does not qualify as a field of its own since many of its methods date back to the 1960s and only differ in access to increased computational resources. For example, Real2Sim can be regarded as system identification and Sim2Real as adaptive control: Both combine analytical models with parametric ones (such as shallow neural networks) to model complex systems and learn controllers. Similarly, model-based reinforcement learning iteratively refines a poor model or a world simulator to learn a controller for the real world.

Sim2Real approaches are having significant impact on a wide variety of research domains beyond automation and control, such as perception and physical modelling. There-

*Corresponding author: mail@sebastianhoefer.de
[1]Amazon Robotics AI [2]Rutgers University [3]NVIDIA Robotics [4]McGill University [5]Mila [6]ElementAI [7]CMU [8]University of Washington [9]UC Berkeley [10]MIT [11]Georgia Tech [12]Technische Universität Darmstadt [13]MPI for Intelligent Systems [14]Columbia University [15]OpenAI [16]University of Alberta

fore, treating Sim2Real as a field of its own provides value by connecting researchers from different disciplines, including computer vision, control theory, physics-based modelling, reinforcement learning and simulation-based inference.

It is understood that highly accurate simulations of reality remain a distant dream and for any level of sophistication of computer simulations there will always exist a reality gap.

How should Sim2Real methods be used? Many physical phenomena, such as friction, impact or deformation, are not easily modeled with sufficient precision by existing simulators. Two approaches can be pursued to address this reality gap: investment into modeling to increase simulation precision vs. investing into algorithms that learn from imprecise and abstract simulation.

Improving simulation precision can mitigate the problem that imprecise simulators may work for some but not other tasks, limiting generalization and extrapolation of the developed solutions. However, imprecise simulators might be "good enough" for Sim2Real. After all, accurate modelling is expensive and often infeasible and any abstraction – and thus, any simulator – inevitably results in loss of accuracy due to only capturing the underlying process at a given level of fidelity, omitting some details. For instance, robots need to learn to collaborate with humans without simulating the entire human brain, and hence learning physical interaction might not require low-level simulation of physics. In fact, social interaction with humans and estimating human intent are often only remotely driven by physics. Therefore, the imperfection of a simulator when linked with domain randomization, can potentially allow for model resilience and policy generalization. Furthermore, it will require to develop Sim2Real for "qualitative" simulators by leveraging abstraction and high-level reasoning.

Model-based reinforcement learning provides another approach, where a model is learned explicitly to be useful for planning. The accuracy of the model is secondary to the resulting accuracy of the learned policies or value functions.

Sim2Real would greatly profit from simulators that provide explicit uncertainty estimates, i.e., modeling where the simulator is uncertain about its future state predictions. Such estimates, while difficult to obtain, can be useful to identify failure cases early on or leverage them to collect data for improving the simulator.

The Utility of Sim2Sim and Simulation-only Results. It is well understood that simulation results alone are not sufficient. Real-world experimentation is required to verify the validity of automation solutions. Yet, simulation-only results can provide valuable insights, by providing, for example, repeatable benchmarks on standardized tasks. Moreover, they are useful for thoroughly analyzing the properties of algorithms. Therefore, simulation-only work should not be discarded by default. Nevertheless, automation researchers should provide convincing evidence about the real-world applicability of simulation-centered work.

Criteria for Using Sim2Real. We propose the following criteria for practitioners to consider before applying Sim2Real approaches: (i) *Bootstrapping*: Is it possible to scope the problem quickly in simulation before real-world application? (ii) *Long-term data starvation*: Is the effort in real-world data collection considerably higher, more dangerous or even infeasible relative to the effort of developing a simulator? (iii) *Hardware-in-the-loop optimization*: Is the goal to co-design hardware and software, which demands the use of simulation to iterate over alternatives? (iv) *Common computer vision problems*: Is the problem mainly a computer vision task where synthetic data are becoming sufficient for training? If the answer to any of these questions is "yes", current Sim2Real techniques might offer valuable solutions.

Promising Techniques Today. (i) *Domain randomization* improves the generalization capability of models, given that the right set of parameters to randomize is known and not too large. (ii) *Explicitly formulated features or state abstractions* alleviate the domain gap if they capture the task and can be extracted accurately both from simulated and real data. (iii) *Combining analytical models with system identification* can close the reality gap for designing and learning controllers. (iv) *Meta-learning and curriculum learning* allows adaptation to changing environments or erroneous models. (v) *Domain adaption* via generative adversarial networks (GANs) enhances the quality of synthetically generated data.

Open Questions and Unsolved Problems: (i) *Ease of use*: most Sim2Real techniques are not yet Plug-&-Play or fully automated, but require careful human attention for understanding the problem and for tuning the parameters of the chosen approach; (ii) *efficient domain randomization* for high-dimensional problems; (iii) *high-fidelity simulation of complex scenarios*, such as contact-rich manipulation, underwater robotics and human-robot interaction; (iv) *differentiable simulators* to adapt analytic models to learn controllers directly from data with strong priors; (v) *high production-level* success rates in Sim2Real transfer.

A new generation of simulation for automation. Simulation has become a ubiquitous tool in automation with a wide variety of applications ranging from process verification to computer-aided design and visualization. Automation, given its requirement of highly reliable and consistent operations, can profit from the new generation of fast computing, improved physical modeling and advanced machine learning techniques. These advances pave the way for applying simulation to novel problem domains and for building the factories, warehouses and plants of the future.

REFERENCES

[1] Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26), January 2019.

[2] Sebastian Höfer, Kostas Bekris, Ankur Handa, Juan Camilo Gamboa, Florian Golemo, Melissa Mozifian, Chris Atkeson, Dieter Fox, Ken Goldberg, John Leonard, C. Karen Liu, Jan Peters, Shuran Song, Peter Welinder, and Martha White. Perspectives on Sim2Real Transfer for Robotics: A Summary of the R:SS 2020 Workshop. `https://arxiv.org/abs/2012.03806`, 2020.

[3] OpenAI, Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, Jonas Schneider, Nikolas Tezak, Jerry Tworek, Peter Welinder, Lilian Weng, Qiming Yuan, Wojciech Zaremba, and Lei Zhang. Solving rubik's cube with a robot hand, 2019.