Real2Sim Transfer using Differentiable Physics

Eric Heiden*, David Millard*, and Gaurav S. Sukhatme
Department of Computer Science
University of Southern California
{heiden,dmillard,gaurav}@usc.edu

***Abstract*—Accurate simulations allow modern machine learning techniques to be applied to robotics problems, with sample-collection runtimes orders of magnitudes faster than the real world. Current reinforcement learning approaches require laborious manual calibration of carefully designed models, or, in a model-free context, vast amounts of training data to acquire such accurate models from real-world trials. In this work, we introduce a new layer in the deep learning toolbox that imposes a strong inductive bias to generate physically accurate predictions of rigid-body dynamics and allows for the automatic inference of system parameters given an ad-hoc model description.**

I. Introduction

Reinforcement learning (RL) enables robots to learn robust policies from experience. Since most state-of-the-art RL algorithms suffer from a prohibitively high sample complexity to train in the real world, policies are typically trained in a simulator first before being transferred onto the actual system. Discrepancies between simulation and reality impair the performance of such policies. There are many approaches to solving this transfer learning problem, including improving simulation fidelity and domain randomization techniques [1].

Instead of open-loop Sim2Real transfer, in this work, we investigate how simulators can be leveraged as a model of the real world that can be updated from experience. By maintaining a probabilistic representation of physical parameters, simulators can play a role in designing exploration policies that excite unseen areas of the physical parameter space. In this way, simulators and trained policies can jointly improve in a Real2Sim and Sim2Real feedback process.

This work is based on our recent article [2] that is currently under review. While we introduce a differentiable physics engine in the previous work, this paper differs significantly by focusing on the important robot learning problem of Sim2Real transfer. We present new experiments where we predict the motion of a real-world double pendulum and show how probabilistic estimation can be leveraged to attain a consistent model of the real world that opens avenues to a principled strategy of domain randomization, a commonly used technique in transfer learning.

II. Rigid Body Dynamics

Throughout this paper, we estimate quantities of a kinematic chain of rigid bodies that are connected by joints. Following [3], we implement the Articulated Body Algorithm which computes the forward dynamics, i.e., the joint accelerations at the next time step given the current joint positions and velocities. Subsequently, we advance the system dynamics using semi-implicit Euler integration. Having implemented the entire physics engine in the C++ automatic differentiation framework Stan Math [4] allows us to compute accurate gradients of any quantity in our simulator, achieving fast convergence with gradient-based optimizers that leverage our engine to estimate model parameters, optimize trajectories, or derive control policies. Throughout this paper, we denote a single step of our physics simulator by $f_\theta(\cdot)$ – a function conditioned on model parameters θ which returns the next world state given the current.

* Equal contribution

III. Real2Sim Transfer

To demonstrate the capability of our approach to estimate physical parameters of chaotic systems in the real world, we estimate the kinematic parameters of a compound pendulum. The dynamics of a pendulum are fully determined by the length of each link. Using a VICON motion capture system, we obtain sub-millimeter accurate positional trajectories of a pendulum of two masses attached to a rigid frame.

Estimating the length of each link is trivial given positional information. Instead, we exercise our approach by estimating these lengths from a trajectory $\tau = \{\mathbf{q}_0, \cdots, \mathbf{q}_T\}$, where $\mathbf{q}_t$ are the generalized joint positions at time step t. We model the double pendulum using spherical joints whose 3D rotations are defined by quaternions. Given $\mathbf{q}_0 = \mathbf{q}_0^*$, we minimize

$$\mathcal{L} = \sum_t ||z(f_\theta(\mathbf{q}_{t-1})) - z(\mathbf{q}_t^*)||_2^2, \tag{1}$$

where z computes unit heading vectors for given quaternions, and $\mathbf{q}_t^*$ are the joint coordinates of the real pendulum.

Leveraging the differentiability of f_θ, we employ the gradient-based Adam optimizer to estimate the link lengths. We converge after ca. 80 epochs for trajectories 10 steps long, sampled at 25 Hz (cf. Fig. 1). Our performance is hampered by the fact that we do not currently simulate joint damping, and use a first-order integrator that quickly accumulates error.

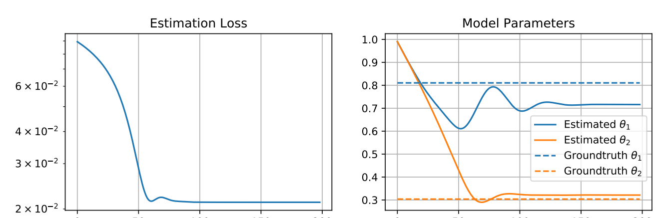

Fig. 1. Estimation of the two link lengths (in meters) of a real double pendulum modeled via spherical joints.

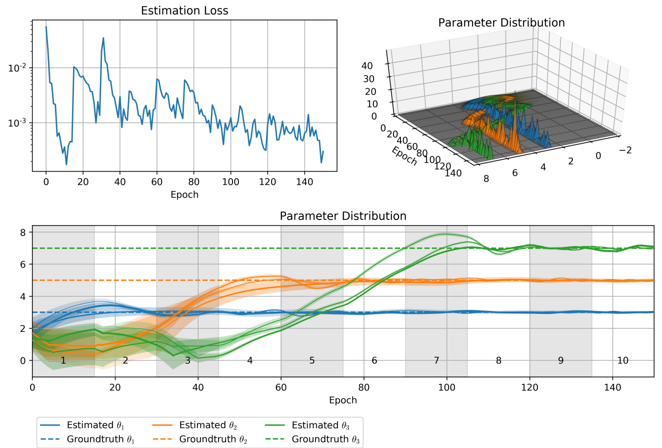

Fig. 2. Probabilistic estimation of the lengths of a three-link compound pendulum using a Gaussian Mixture Model. *Top left:* loss evolution, *top right:* 3D visualization of the evolution of the GMM kernels, *bottom:* GMM kernel evolution (means are solid lines, $\mu_i \pm \sigma_i$ are shaded areas). Numbers on vertical stripes indicate number of real world samples observed in the online estimation process.

IV. Probabilistic Estimation

Minimizing the sum-of-squares error in Eq. 1 by directly adjusting the model parameters θ results in the approximation of the mean of the observed data (cf. [5]). While in the previous estimation experiment, the parameters are defined uniquely over a long-enough trajectory, such assumption need not hold anymore for more complex models or very few samples from the real system. Furthermore, some models of real-world systems can have coupled variables. For instance, if one wants to estimate both the length and the center of mass of a pendulum given a trajectory of angles, the two quantities of interest would mirror each other. Since we are interested in a general estimation approach that is able to capture potential couplings between model parameters and yield results under the presence of noisy observations, we investigate a multi-modal stochastic estimation approach to infer physical parameters.

In this experiment we consider a three-link compound pendulum parameterized by the 3D vector of link lengths θ. Instead of having access to the entire joint angle trajectory from the real system, we investigate an online estimation process where the parameters need to be inferred as samples $\mathbf{q}_t^*$ are observed from the actual pendulum. We model the distribution over θ by a Gaussian Mixture Model (GMM) consisting of three multivariate Gaussian kernels conditioned on learned variables for mixing coefficients ϕ_i, mean μ_i and variance σ_i of kernel i (cf. Fig. 3). To compute a sample θ from a GMM, we need to sample from a categorical distribution $\gamma \sim \text{Discrete}(\phi_1, \phi_2, \phi_3)$ that uses the mixing coefficients as probabilities in order to pick the index γ of the Gaussian kernel we eventually sample from. To achieve an end-to-end differentiable stochastic computation graph [6] where we can optimize the GMM parameters w.r.t the final loss, we approximate this operation using Gumbel Softmax [7]:

$$\gamma_k = \frac{\exp\left((\log \phi_k + G_k)/\lambda\right)}{\sum_{i=1}^{n} \exp\left((\log \phi_i + G_i)/\lambda\right)},$$

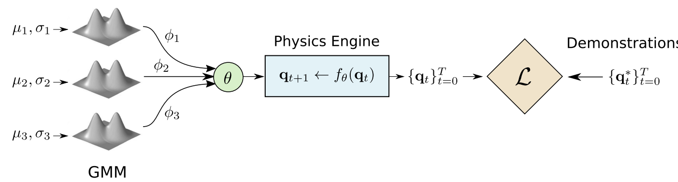

Fig. 3. Stochastic computation graph [6] of our multimodal estimation pipeline based on a three-component Gaussian Mixture Model (GMM) from which the sampled model parameters θ are given to the differentiable physics engine which in turn deterministically computes trajectories $\{\mathbf{q}\}_{t=0}^T$. The overall loss between the trajectories and demonstrations $\{\mathbf{q}^*\}_{t=0}^T$ is minimized, with gradients flowing end-to-end from the GMM parameters μ_i, σ_i, ϕ_i to $\mathcal{L}$.

where $G_i, G_k \sim$ Gumbel and λ (set to $2/3$) is the temperature defining the "spikiness" of the approximation. Next, we sample from each Gaussian kernel and sum up these samples weighted by γ_i to obtain the physical model parameters θ that we feed into the physics engine which computes a trajectory of joint positions. Minimizing the Huber loss between the observed and generated trajectories via the Adam optimizer, the Gaussian kernels converge after about 120 training epochs (Fig. 2) to the true model parameters, while the variance is constantly decreasing.

V. Conclusion

In this work, we have presented a novel approach to estimating physical parameters of complex real-world systems consisting of rigid bodies. Our differentiable physics engine allows for its integration into deep learning models where predictions are based on interpretable, physical parameters that can be optimized using end-to-end-derived gradients. As we have demonstrated, by imposing stochasticity on the model parameters, we are able to capture noisy observations and couplings between these variables. Such approach allows our simulator to act as a model of the real world that can be updated online and provide an informative estimate of its uncertainty. While we have thus far only investigated the Real2Sim aspect of transfer learning, future research is directed towards leveraging the model uncertainty to guide exploration in the real world such that new observations further close the gap between simulation and reality.

References

[1] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *IROS*, 2017.
[2] E. Heiden, D. Millard, H. Zhang, and G. S. Sukhatme, "Interactive differentiable simulation," 2019.
[3] R. Featherstone, *Rigid Body Dynamics Algorithms*. Berlin, Heidelberg: Springer-Verlag, 2007.
[4] B. Carpenter, M. D. Hoffman, M. Brubaker, D. Lee, P. Li, and M. Betancourt, "The Stan Math Library: Reverse-mode automatic differentiation in C++," 2015.
[5] C. M. Bishop, "Mixture density networks," Tech. Rep., 1994.
[6] J. Schulman, N. Heess, T. Weber, and P. Abbeel, "Gradient estimation using stochastic computation graphs," in *NIPS*, 2015.
[7] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," *ICLR*, 2017.