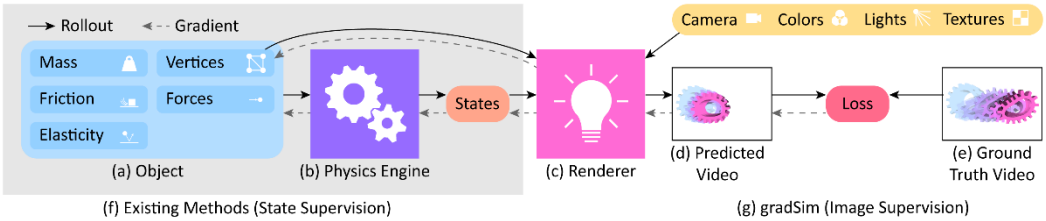
Report on “∇Sim: Differentiable Simulation for System Identification and Visuomotor Control” by Krishna Murthy Jatavallabhula et al.

ADARSH S MENON

The report on “∇Sim: Differentiable Simulation for System Identification and Visuomotor Control” by Krishna Murthy Jatavallabhula et al talk about new technology framework ∇Sim. It is a unified differentiable rendering and Multiphysics framework that allows solving a range of control and parameter estimation tasks (rigid bodies, deformable solids, and cloth) directly from images/video.

The researchers considered the problem of estimating an object’s physical properties such as mass, friction, and elasticity directly from video sequences. Such a system identification problem is fundamentally ill-posed due to the loss of information during image formation. Current solutions require precise 3D labels which are labour-intensive to gather, and infeasible to create for many systems such as deformable solids or cloth.

Accurately predicting the dynamics and physical characteristics of objects from image sequences is a long-standing challenge in computer vision. This end-to-end reasoning task requires a fundamental understanding of both the underlying scene dynamics and the imaging process.



The researchers’ main contributions are:

- GradSim, a differentiable simulator that demonstrates the ability to backprop from video pixels to the underlying physical attributes.
- demonstrated recovering many physical properties exclusively from video observations, including friction, elasticity, deformable material parameters, and visuomotor controls (sans 3D supervision)
- A PyTorch framework facilitating interoperability with existing machine learning modules.

To investigate whether the gradients computed by ∇Sim are meaningful for vision-based tasks, researchers conducted a range of visuomotor control experiments involving the actuation of deformable objects towards a visual target pose (a single image). In all cases, researchers evaluated against diffphysics, which uses a goal specification and a reward, both defined over the 3D state-space.

The first example (control-walker) involves a 2D walker model. The goal is to train a neural network (NN) control policy to actuate the walker to reach a target pose on the right-hand side of an image. Our NN consists of one fully connected layer and a tanh activation. The network input is a set of 8 time-varying sinusoidal signals, and the output is a scalar activation value per-tetrahedron. ∇Sim is able to solve this environment within three iterations of gradient descent, by minimizing a pixelwise MSE between the last

---

Author’s address: Adarsh S Menon.

frame of the rendered video and the goal image. In other second test, they formulated a more challenging 3D control problem (control-fem) where the goal is to actuate a soft-body FEM object (a gear) consisting of 1152 tetrahedral elements to move to a target position shown in the report. They used the same NN architecture as in the 2D walker example, and use the Adam (Kingma & Ba, 2015) optimizer to minimize a pixelwise MSE loss. They also train a privileged baseline (diffphysics) that uses strong supervision and minimizes the MSE between the target position and the precise 3D location of the centre-of-mass (COM) of the FEM model at each time step (i.e., a dense reward). They test both diffphysics and ∇Sim against a naive baseline that generates random activations and plot convergence behaviours. While diffphysics appears to be a strong performer on this task, it is important to note that it uses explicit 3D supervision at each timestep (i.e., 30 FPS). In contrast, ∇Sim uses a single image as an implicit target, and yet manages to achieve the goal state, albeit taking a longer number of iterations.

The researchers design an experiment to control a piece of cloth by optimizing the initial velocity such that it reaches a pre-specified target. In each episode, a random cloth is spawned, comprising between 64 and 2048 triangles, and a new start/goal combination is chosen. In this challenging setup, they noticed that state based MPC (diffphysics) is often unable to accurately reach the target. they believed this is due to the underdetermined nature of the problem, since, for objects such as cloth, the COM by itself does not uniquely determine the configuration of the object. Visuomotor control on the other hand, provides a more well-defined problem. An illustration of the task is presented in the report.

Being a white box method, the performance of ∇Sim relies on the choice of dynamics and rendering models employed. An immediate question that arises is “how would the performance of ∇Sim be impacted (if at all) by such modelling choices.” The researchers conduct multiple experiments targeted at investigating modelling errors and summarize them in Table 4 (left). They choose a dataset comprising 90 objects equally representing rigid, deformable, and cloth types. By not modelling specific dynamics and rendering phenomena, researchers created the following 5 variants of our simulator. 1. Unmodeled friction: model all collisions as being frictionless. 2. Unmodeled elasticity: model all collisions as perfectly elastic. 3. Rigid-as-deformable: All rigid objects in the dataset are modelled as deformable objects. 4. Deformable-as-rigid: All deformable objects in the dataset are modelled as rigid objects. 5. Photorealistic render: They employ a photorealistic renderer—as opposed to ∇Sim’s differentiable rasterizers—in generating the target images. In all cases, they evaluated the accuracy with which the mass of the target object is estimated from a target video sequence devoid of modelling discrepancies. In general, it is observed that imperfect dynamics models (i.e. unmodeled friction and elasticity, or modelling a rigid object as deformable or vice-versa) have a more profound impact on parameter identification compared to imperfect renderers.

The researchers also independently investigate the impact of unmodeled rendering effects (assuming perfect dynamics). They independently render ground-truth images and object foreground masks from a photorealistic renderer (Pharr et al., 2016). They use these photorealistic renderings for ground-truth and perform physical parameter estimation from video. They noticed that the performance obtained under this setting is superior compared to ones with dynamics model imperfections.

Although this work does not attempt to bridge the reality gap, researchers were able to show early prototypes to assess phenomena such as shading/texture IZn the report it is shown that accuracy over time for mass estimation from video. The paper evaluates three variants of the renderer - “Only colour”, “Shading”, and “Texture”. The “Only colour” variant renders each mesh element in the same colour regardless of the position and orientation of the light source. The “Shading” variant implements a Phong shading model and can model specular and diffuse reflections. The “Texture” variant also applies a non-uniform texture sampled from ShapeNet (Chang et al., 2015). They notice that shading and texture cues significantly improve convergence speed. This is expected, as vertex colours often have very little appearance cues inside the object boundaries, leading to poor correspondences between the rendered and ground-truth images. Furthermore, textures seem to offer slight improvements in convergence speed over shaded models, as highlighted by the inset (log scale) in the report.

Author’s address: Adarsh S Menon.

The research shows simulation rates for the forward and backward passes of each module. The report forward and backward pass rates separately for the differentiable physics (DP) and the differentiable rendering (DR) modules. The time complexity of ∇Sim is a function of the number of tetrahedrons and/or triangles. The report illustrates the arguably more complex case of deformable object simulation for varying numbers of tetrahedra (ranging from 100 to 10000). Even in the case of 10000 tetrahedra—enough to construct complex mesh models of multiple moving objects—∇Sim enables faster-than-real-time simulation (1500 steps/second).

The report presented ∇Sim, a versatile differentiable simulator that enables system identification from videos by differentiating through physical processes governing dynamics and image formation. The researchers demonstrated the benefits of such a holistic approach by estimating physical attributes for time evolving scenes with complex dynamics and deformations, all from raw video observations. They also demonstrated the applicability of this efficient and accurate estimation scheme on end-to-end visuomotor control tasks. The latter case highlights ∇Sim's efficient integration with PyTorch, facilitating interoperability with existing machine learning modules. Interesting avenues for future work include extending other differentiable simulation to contact-rich motion, articulated bodies, and higher-fidelity physically-based renderers – doing so takes us closer to operating in the real-world.