Herbie's Update Write-up

2024 July 26

Introduction

I got the following feedback from last time:

- For the pitch: connecting to downstream task, demonstrate better reconstructions, and couple with V-PRISM
- Create a test suite of 5-10 examples
- Then, explore gradient computation

Thinking about the Pitch

Previously, the pitch was something like:

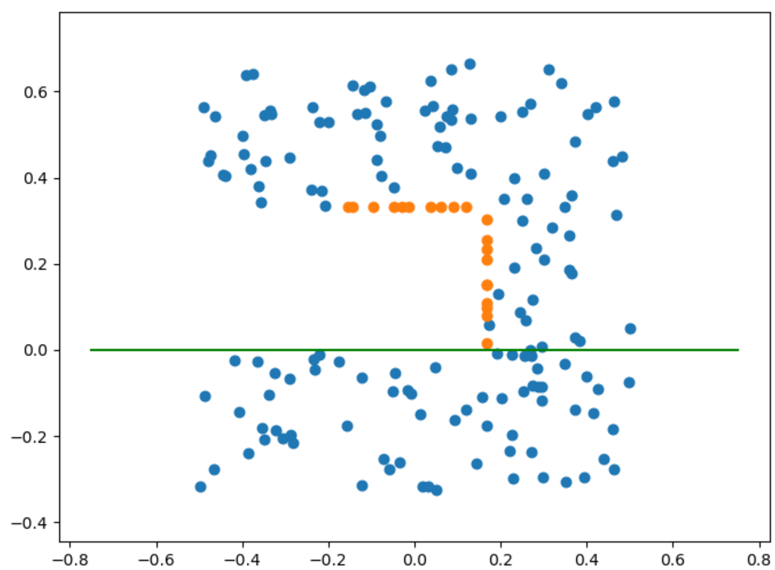
- Detailed, accurate 3D reconstructions allow robots to better reason about 3D geometry, which is useful for many manipulation tasks
- Most SoTA methods are not physically accurate/stable, so there is room for improvement.
  - Methods that do are either non-differentiable [1] or use heuristic loss terms [2]
- We will create a loss term that penalizes physical stability of a scene that can be used in two ways:
  - During training similar to [2]
  - During optimization-based reconstruction like in [3], [4]
- Ideally, we would like our method to improve:
  - Shape completion/reconstruction accuracy
  - Downstream task success

Thinking about what downstream tasks we could show success with, we could do a similar sort of thing to [2]:

- Different robotic tasks (grasping, pushing, and pulling) and did the following:
  - First reconstruct the scene
  - Then plan a trajectory with the 3d reconstruction for the task
  - Finally the trajectory is executed with the original object (all in sim).
- They released this benchmark I think, so it would be possible to reproduce it
  - They didn't release their NN model weights though, so it will need to be retrained for direct comparison.

New Test Suite

Some concerns were brought up about overfitting to the square example:



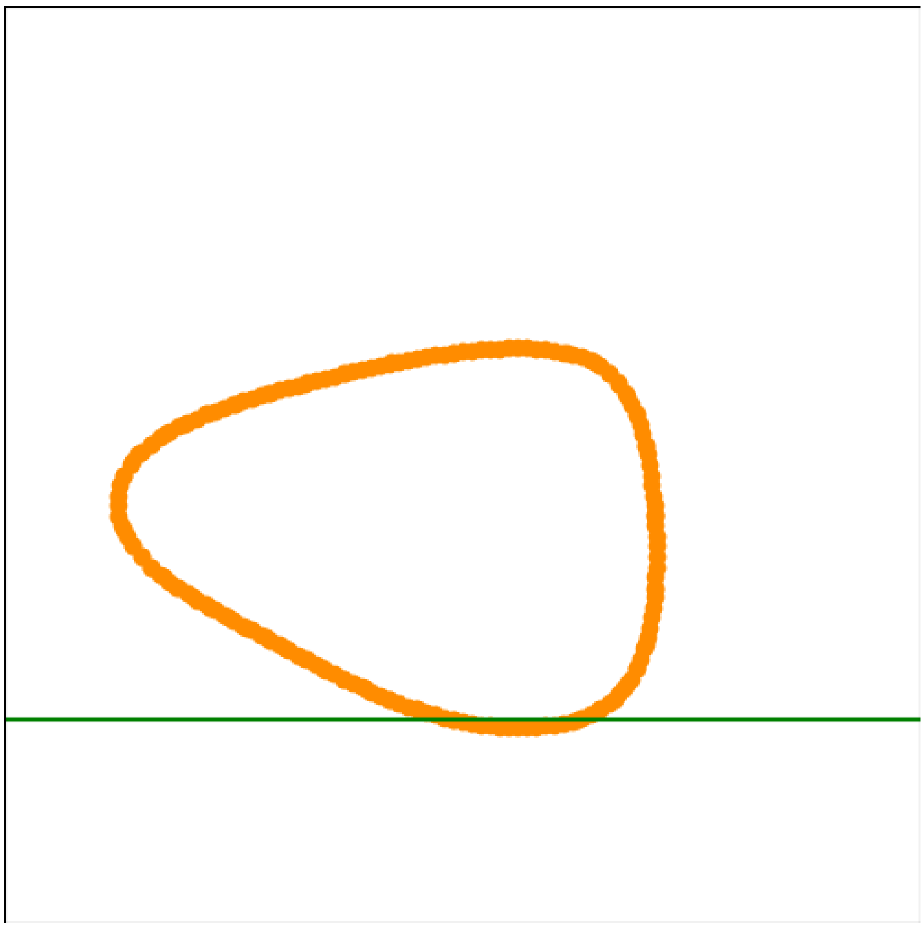
As such, it would be a good idea to come up with a few different examples that could be used to prevent overfitting. I was thinking we could have the following examples

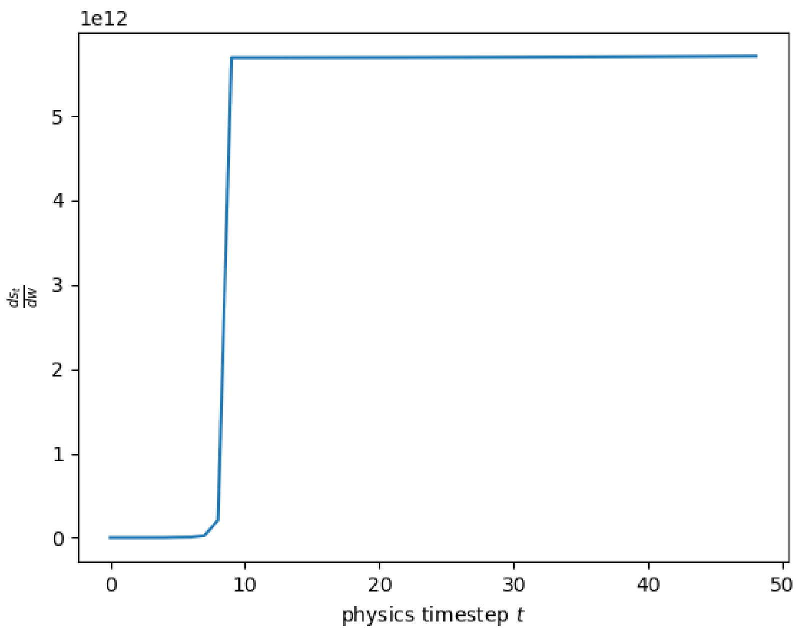
- 2D square (above)
- 2D circle
- 2D tall hourglass
- 3D cube
- 3D mesh of something (like mustard bottle)
- 3D real-world object

I need to add some updates to my code for the 3D stuff to handle rotations, etc. correctly.

Gradient Computation

Last time, I brought up some concerns about gradients. I based my investigation loosely off of [5]. I found that there was evidence of exploding gradients:





Specifically, I was looking at the Jacobian of the dynamics $\partial s_t / \partial s_0$ (The above image is the norm of the Jacobian $\partial s_t / \partial w$). The explosion is likely due to the spectral properties of the Jacobian of the dynamics function being iterated.

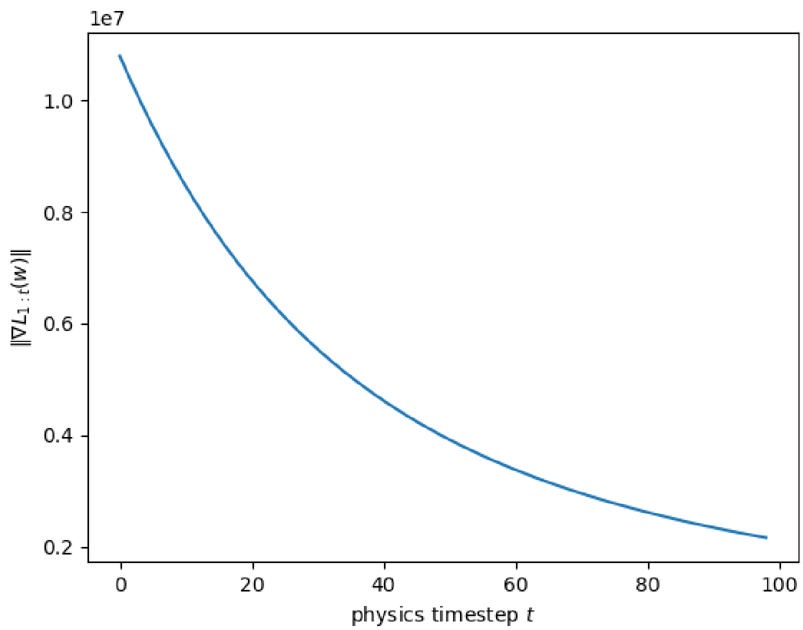
In [5], there were a few suggestions for dealing with this:

- **Smoothing:** "Contact 'softening' has proven particularly fruitful."
- **Black-Box Gradients:** "Sometimes, black box gradient estimates can result in lower variance gradient estimates"

I also had the idea that perhaps an exponentially decaying loss function would help with this problem. Then, the loss would be something like:

$$L_{\text{stable}} = \frac{1}{T} \sum_{t=1}^{T} \frac{\alpha^t}{2} \|s_t - s_0\|^2$$

for $0 < \alpha < 1$. Here is a graph of the norm of the gradient of the above loss for $\alpha = 0.95$ in the above setup (note the scale is 1e7 instead of 1e12):



On Smoothing

In [5] both smoothing of contact and randomization/probabilistic treatment of dynamics were talked about as potential solutions to the high variance of jacobians problem. In [6], an equivalence between the two was drawn. I think this could be a promising approach, but would take a little bit of thought to adapt to the current setup. Currently, I have a spring-damper setup similar to [7], [8], but would like to "smooth" things out. This would probably require replacing the ReLU functions with a Softplus function.

On Black-Box Gradients

Originally, REINFORCE is mentioned as an example of computing Black-Box gradients. Some methods using Complex Step Finite Differencing (CSFD) was mentioned:

$$f'(x) \approx \mathrm{Im}\left(\frac{f(x + hi)}{h}\right)$$

These seem really cool, but there might have to be some thought about using this with high dimensional gradients.

Looking Forward

Looking forward, I want to do the following next week:

- Finish test suite
- Create V-PRISM presentation

References

[1] C. Song and A. Boularias, "Inferring 3d shapes of unknown rigid objects in clutter through inverse physics reasoning," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 201–208, 2018.

[2] W. Agnew, C. Xie, A. Walsman, O. Murad, Y. Wang, P. Domingos, and S. Srinivasa, "Amodal 3d reconstruction for robotic manipulation via stability and connectivity," in *Conference on robot learning*, 2021, pp. 1498–1508.

[3] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 165–174.

[4] H. Wright, W. Zhi, M. Johnson-Roberson, and T. Hermans, "V-prism: Probabilistic mapping of unknown tabletop scenes," *arXiv preprint arXiv:2403.08106*, 2024.

[5] L. Metz, C. D. Freeman, S. S. Schoenholz, and T. Kachman, "Gradients are not all you need," *arXiv preprint arXiv:2111.05803*, 2021.

[6] T. Pang, H. T. Suh, L. Yang, and R. Tedrake, "Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models," *IEEE Transactions on robotics*, 2023.

[7] M. Geilinger, D. Hahn, J. Zehnder, M. Bächer, B. Thomaszewski, and S. Coros, "Add: Analytically differentiable dynamics for multi-body systems with frictional contact," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, pp. 1–15, 2020.

[8] J. Xu, V. Makoviychuk, Y. Narang, F. Ramos, W. Matusik, A. Garg, and M. Macklin, "Accelerated policy learning with parallel differentiable simulation," *arXiv preprint arXiv:2204.07137*, 2022.