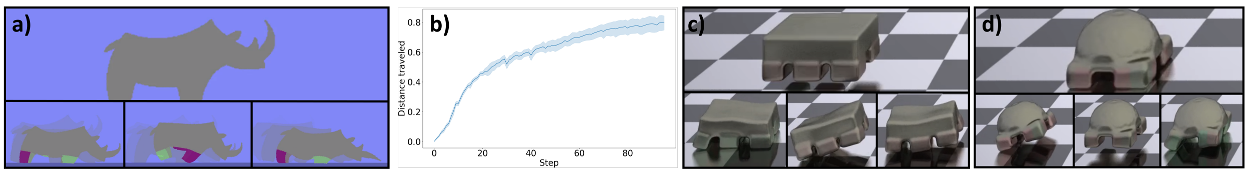
We thank the reviewers for their constructive feedback. While there are many papers on rigid robot control each year at NeurIPS, control of soft robots has seldom been addressed. Our submission is novel, bridging learning and soft robotics, and is the first to tackle end-to-end control and co-optimization of soft robots of arbitrary morphology. Our algorithm takes natural advantage of fully differentiable simulation, which is exploding in popularity and relevance[1,2,3].

We appreciate the reviewers' compliments that our submission is *"an interesting piece of work that can have a good impact in the field of soft robotics..." (R3)*, *"very novel learning work on soft robots"* with *"impressive"* experiments and performance (*R1*), and that the approach is *"promising" (R2, R3)* with *"potential to be relevant for future work..." (R3)*. We believe concerns can be addressed within the review cycle with text improvements and additional experiments.

More Complex, Non-Blocky/3D Designs (*R2, R3*). We can easily add more complex, non-blocky and 3D designs to the results and video. MPM particles are flexible enough to represent most irregular geometry found in soft robotics and CNNs can adequately learn over such inputs. We include a few new results below. We hope these new results complete a convincing gamut of experiments already described as *"impressive," (R1)* and *"diverse"* and *"promising" (R3)*.



a) A new rhino robot loaded from image, serving as a curvy, non-blocky 2D example. **b)** Convergence of the rhino control task over 10 trials. The topheavy, unactuated head makes this a challenging control task. **c)** With 24 actuators and highly nonlinear dynamics, this 3D Hexapod is now our most complex demo. After 100 optimization iters., it runs 1.5 body lengths in $4s$. **d)** This new 3D Quadruped's hemispherical body proves our method works on less blocky 3D shapes as well. After 100 optimization iterations, it runs two body lengths in $4s$.

Clarification on FEM vs. MPM for Learning (*R1*). We are not saying that one cannot learn a latent representation on FEM nodes; in fact, it is possible our approach could extend to FEM by rasterizing node velocities to a grid and directly applying our method. However, such an approach has never been demonstrated. We chose MPM because *1)* prior work[1] demonstrates its success for control, as it naturally handles differentiable contact, and *2)* it acts directly on a velocity grid, providing a representation amenable to CNNs for free.

Why a Latent Space Is Necessary (*R1*). It is indeed impractical to try to learn directly on FEM node or MPM particle coordinates. This approach doesn't scale: we tried feeding 1000 MPM particles into our controller (a relatively small number), and runtime for simulation and backprop ballooned by $10\times$ compared to compact latent features. As a further advantage, velocity grids can easily be captured in the real world *via* optical flow; dense node coordinates cannot.

Simulation as Prior Knowledge (*R1*). We do not consider the *simulator*, be it FEM or MPM, as prior knowledge, but rather the *data* it generates. In previous work the robot is simulated along many random trajectories to build a prior dataset. If the dynamics of the target trajectory are not explored initially, the observer and resulting optimization suffer. This issue is especially salient during design optimization, where system dynamics change. LITL continually generates representative data throughout the optimization phase and re-learns, thus it does not suffer this drawback.

Initial Dataset Generation (*R3*). A small initial dataset is generated from simulating just once with the initial, untrained controller. This is enough to bootstrap our learning.

Benefits of Co-Optimization and Co-Learning (*R1*). The value of co-optimization has been highlighted in prior work for rigid[4,5] and soft robots[1]; it allows robots to solve difficult tasks more easily and improve performance. We can add comparisons of performances with and without co-optimization. The latent space must be co-learned since the experienced dynamics (and thus, optimal observer) change during co-optimization.

Problem Scope (*R1*). *R1* wrote *"of course the paper's focus is on multi-task learning for soft robotics."* We wish to clarify that this paper's focus is *not* on multi-task learning, but single task learning with highly dynamic soft robots. As *R3* states, *"this is a challenging problem."* It is one that has seldom been successfully tackled by any literature; ours is the first end-to-end co-design solution for general morphologies. While multi-task learning would be an interesting extension, it is yet another very hard problem, one rich enough for its own dedicated manuscript.

Training Stability (*R1, R2, R3*). *R2* and *R3* asked how feature oscillation affects *"stability and convergence."* "Backward progress" from feature oscillation is dominated by the following optimization phase and typically undone within 1-2 iterations. We can add tables quantifying this effect. *R1* asked why we couldn't learn *"the representation explicitly over learning the controller directly for the task."* We tried a joint optimization, (see section 4.3, **Alternative vs. Simultaneous Minimization**), but the CNN layers learned too slowly to provide useful signal, leading to bad local minima. Alternating minimization avoids this issue in economical fashion.

[1] Hu, Yuanming, et al. "ChainQueen: A Real-Time Differentiable Physical Simulator for Soft Robotics." 2019 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2019.
[2] de Avila Belbute-Peres, Filipe, et al. "End-to-end differentiable physics for learning and control." Advances in Neural Information Processing Systems. 2018.
[3] Schenck, Connor, and Dieter Fox. "SPNets: Differentiable Fluid Dynamics for Deep Neural Networks." Conference on Robot Learning. 2018.
[4] Ha, Sehoon, et al. "Joint Optimization of Robot Design and Motion Parameters using the Implicit Function Theorem." Robotics: Science and systems. 2017.
[5] Spielberg, Andrew, et al. "Functional co-optimization of articulated robots." 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017.