Thanks to all the reviewers for their thorough reviews and helpful comments. We appreciate that all the reviewers, even the one most critical of the paper, felt that the fundamental method we introduce here (that of constructing explicitly stable dynamical models, by construction, via the joint dynamics and Lyapunov function), is a novel and exciting methodology.

We also appreciate that the reviewers, especially Reviewer 2, had some concerns with aspects of the papers as well. The main concerns raised were that 1) the method needed further empirical comparisons and 2) the domains studied were rather small/toy domains. To each of these points, we'd like to make the following comments:

Empirical comparisons. Reviewer 2 is correct that in all cases the main comparisons here were against a neural network (of the same approximate architecture) with the explicit stability removed. We should point out that, although this reviewer asks us to compare to an RNN-based approach, because we are applying the dynamics recursively, the comparisons we make in all cases (the "simple" or "naive" models in the experimental results) *are* effectively RNNs, just with a structure to make them as comparable as possible to our stable models. If the objection is that the typical RNN has a squashing non-linearity that prevents divergence to infinity, we note that this doesn't actually prevent divergence to extreme values allowed by the nonlinearity (this is very common; we have a squashing function on the output of our VAE). We're happy to include the traditional (e.g., Tanh or LSTM) RNN for comparison, and will add this to the paper; they do not differ substantially from the simple models. We didn't include Embed2Control-style comparisons, as the latent linear modeling there seems largely orthogonal to our main points here, and because we are considering autonomous rather than controlled systems in the current work.

Toy problems. We also acknowledge the problems used here are largely meant as demonstrations and thus we felt that the problems, while toy tasks, are still sufficient to illustrate the method. Reviewer 2 felt that the video texture generation was not a good candidate, as a random walk in latent space would also create "videos"; but random walks in latent space result in trajectories that look nothing like the actual dynamics of the system, whereas our modeled trajectories do. We can include a discussion and illustration of this in the revision. And while still a "toy" problem, we felt the 320-dimensional state space of these systems did illustrate the scalability of the approach. However, including an additional spring-damper system is a great idea, and we'll add this to the final paper.

While we certainly agree that these elements can be strengthened, we feel that as demonstrations they highlight the value of the approach, and the overall methodological contribution here is substantial enough to warrant acceptance, as several of the reviewers felt. In addition to these main points, we address a few other comments, and will edit the paper to clarify all these points.

Reviewer 1 The "simple" model always refers to a simple feedforward network (with the same structure as $\hat{f}$ but without the stability constraints). For the pendulum this is a $2n$-120-120-$2n$ network where n is the number of links in the pendulum.

We'll fully describe the video texture setup in the text (e.g., the source videos are actual videos of physical fire from YouTube) Naive model (again, just same network structure without stability constraints) means that the predicted latent variable diverged to infinity. Thanks for pointing out the confusion here, we'll clarify all of these.

Reviewer 2 Regarding regularization in general, we can quite easily show that just regularizing the weights of the network is insufficient to achieve stability in practice, unless the system is *extremely* regularized. Additionally, even with regularization this stability is hard to show formally (except locally), and introduces the additional regularization term. However, we'll certainly discuss this point more.

Reviewer 3 We'll include all these details for the experiments (lack of space to desribe them all here). Thanks very much for pointing on these confusing points.

Thanks also for highlighting these two related works. After going through them a bit, they do seem a bit different in focus (and of course quite different in methodology) as both are concerned with controller design rather than modeling autonomous sytems; but we'll absolutely include and discuss them.

Reviewer 4 Thanks for highlighting the connections to e.g., recent work on Neural ODEs and work analyzing the stability of RNNs. While these points are somewhat orthogonal, we believe they may actually highlight some nice additional applications of the method (the aforementioned approaches indeed often rely on stability, and thus could be improved by including systems stabilized via our approach).