Reviewer 1 We thank the reviewer for their helpful comments. We agree that the increased training time is a notable disadvantage to our method. However, improving test time evaluation at the expense of increased train-time is a reasonable trade-off. In applications where the model is evaluated many more times than it is optimized the increased training cost can be justified. In response to this concern we will connect our motivation for improved test time performance by referencing the existing literature on efficient inference methods, e.g. quantization.

The reviewer's primary concern is this work's relationship with existing literature. We agree that the relationship to the workshop paper Bettencourt et al. 2019 should be made clear.

Our paper includes a detailed comparison to Finlay et al. 2020 that addresses the reviewer's concerns about our contributions. Our regularizer, like theirs, is augmenting the loss with an extra penalty term. Their regularization terms are motivated by optimal transport and reusing computation specifically in FFJORD. We show in the appendix how their regularizer can be generalized to ours, and how ours tracks the expense of the chosen adaptive solver. Our motivation can be extended to optimize properties of other solvers, e.g. stiffness. Due to these distinctions, our method is a contribution to the literature. Empirically, we extensively compare to Finlay et al. 2020 in our experiments on density estimation. We also include experiments on additional domains and phenomena such as regularization effect on overfitting.

The reviewer's comment on time series results reference the similarities between Figures 1 and 4. This suggests to us that we can improve our presentation in those figures, as they are not comparable. Figure 1 is a 1D state vs time plot, whereas Figure 4 is a phase plot, and time is denoted implicitly by the arrows along the curves. Also, the figures describe different tasks with different properties. Figure 1 describes a simple output classified from the input, so the intermediate dynamics are not explicitly constrained by data and could be regularized to straight trajectories. Figure 4 describes a time series task, so the trajectories must model the data along intermediate values and are constrained from becoming straight trajectories. The result is that we can improve evaluation cost without changing the appearance of the trajectories.

As we explain in Section 6.2, we agree that $K = 3$ with a 3rd order solver shows a marginal improvement over other solvers, and there is no clear winner for order 5 (6c) or adaptive (6d). We apologize for the misleading claim on lines 99-100, and will change the wording.

Reviewer 2 We will remove the qualifiers on l. 30 and add more detail on l. 47. We will include the detail on l. 64. Re. l. 77, we fear that we may mislead the reader about the applicability of our method without being clear about its drawbacks. We will remove "better" from Broader Impact. There is a detailed appendix section for Taylor-mode AD.

We agree that the dynamics seem stiff in Figure 4. We note that the caption of this figure summarizes the performance of the model and the improvement in NFE. Your suggestions for improving clarity in Section 5 are very appreciated. We will reorganize the discussion of NFE vs. computation time to avoid repetition. Thank you for highlighting the connection to adversarial robustness. Although we did not investigate it in this work, we think this is an interesting avenue for potential future research, and will cite these works in the main text. The noise in 5b) is from the optimization (noise which is present without our regularization). Any tuning of optimization or other hyperparameters were done on unregularized models and left unchanged when training with regularization. Unfortunately we were not able to analyze the distribution of orders chosen by the adaptive solver due to the engineering required. We thank the reviewer for raising this as we think it's an interesting question. The formula for the y-axis in Figure 8a) is $\log \max_i \{x_i - x_i^{\text{true}}\}$, where x_i^{true} is the ith component of the (fixed) true solution computed using a tight tolerance for the solver, and x_i is the ith component computed with `atol` and `rtol` parameters passed to the solver as given on the horizontal axis.

Reviewer 4 We thank the reviewer for pointing out interesting literature we were not previously aware of. The high-level conceptual connection is a good one, but the methods and motivations are quite distinct: 1) we provide soft constraints in the form of regularization instead of a hard constraint forcing the derivatives to be exactly zero; 2) we only regularize one higher derivative, and not all of them simultaneously, so the lower derivatives need not be small even if the higher ones are; 3) in many cases the ODE trajectories need not smoothly approximate a function along the whole interval, but rather only at the endpoints, e.g. as in Figure 1 (see second last paragraph in comments for Reviewer 1).

Nevertheless it is interesting to motivate our regularizer by connecting it to the notion of certain priors on paths, and we thank the reviewer for making us aware of this literature. We will cite this work in the main text. Potentially related is Figure 8. c) where we investigated the potential effect of statistical regularization from our method, and found there was in fact little effect.

As for the comment about test time, see the comments for Reviewer 1.