We would like to thank all the reviewers for their reviews with insightful comments and suggestions. We would kindly ask the reviewers to consider raising the scores, if we have addressed the scalability concern satisfactorily.

Scalability and ImageNet: We want to first address concerns regarding scalability. *The overall wall-clock training time for ODE-GAN is comparable with SN-GAN*. While the RK4 algorithm requires four gradient calculations and the regularizer two backward passes, it avoids the extra discriminator updates (5 discriminator updates per generator update in SN-GAN). The ODE update of our algorithm has the same memory footprint as gradient descent, as each gradient in the RK4 step is computed sequentially. Therefore, the scalability of our algorithm is comparable to SN-GAN.

Table 1: Comparison of ODE-GAN and the SN-GAN trained on ImageNet (*unconditional* image generation).

	FID	IS
SN-GAN	57.24 ± 0.79	13.83 ± 0.19
ODE-GAN	$\mathbf{49.82 \pm 0.93}$	$\mathbf{15.08 \pm 0.07}$

Table 1 summarises additional experiments on ImageNet at resolution 128 x 128 for unconditional image generation. We followed the setup of [1], using the same ResNet and batch size 64 (learning rate 0.01, regularisation weight 0.001, all other settings are the same as for CIFAR10). The scores were computed from 10,000 samples. The advantage of our algorithm is already clear from these preliminary results; we are working on improving them further for the final version.

Reviewer 1: We agree with your suggestion to include additional context in the manuscript. In particular, we will include an exposition on differential Nash equilibria and rotational fields in the supplementary. Background on the convergence of linear dynamical systems will be added to the proof of Lemma 3.1. **Re. Intuitions about the generator's gradient exploding:** We observed large gradients when the generator samples are poor, where the discriminator may perfectly distinguish the generated samples from data. This potentially sharp decision boundary can drive the magnitude of the generator' gradients to infinity. We will provide additional intuition in the revision.

Reviewer 2: We addressed your request regarding scalability/more complex generation experiments such as ImageNet in the first section of this rebuttal. **Re. Other GAN losses:** The majority of our analysis can be extended to other smooth and differentiable loss functions including the Wasserstein loss. We will discuss this in more details in the revision.

Reviewer 3: We appreciate your in-depth comments which help us further consolidate the links by which we connect GAN training with numerical analysis. The nature of our work can entail a clash of terminologies that are overloaded in these fields, which we want to clarify first. **Re. terminology** We will add a short paragraph to clarify the following: the "instability" we refer to is of training GANs (for example due to mode collapse); "convergence" refers to convergence to either the differential Nash equilibrium or to a fixed point; "gradient descent" in our case means following the gradient of losses for each player in the game; the term "infinitesimal gradient descent" is indeed inherited from [2] which refers to the dynamics when we tend the step size of gradient descent to zero for the two-player game. Further, line 38 should indeed be changed to "We find that the main ingredient we need for stable GAN *training* is to avoid large integration errors" reflecting our hypothesis that *instability in GAN training (at least near a Nash-equilibrium) arises from discretization errors in the underlying ODE*. This hypothesis hinges on the observation that the purely rotational case (namely when $\epsilon = 0$ in Eq. 7) does not apply to GANs near differential Nash equilibrium under mild assumptions (Lemma 3.1 and Remark). Under this assumption we can use higher order numerical integrators as well as first order (such as Euler) with a lower step-size (as also pointed out by the reviewer) to stabilize GAN training. In practice, for large-scale image generation with GANs, decreasing the step-size for Euler integration, slows down training significantly (see Fig. 3 where Euler is stable but convergence is too slow).We will clarify these issues further in the revision. **Re. solver implementation** We required a fast solver implementation in TensorFlow and thus wrote all integrators from scratch. **Re. suggestions** Thanks for your suggestion regarding implicit/symplectic numerical schemes, this is something we want to explore in the future.

Reviewer 4: **Re. Quantitative evaluation on the quality of the generated samples** We have quantitatively evaluated samples generated via the Inception Score and FID (Table 1 and H1), which are the main metrics used in the literature. If there are additional metrics the reviewer would suggest we are happy to include them. **Re. Other GAN architectures** Yes, our method is not restricted to a specific GAN architecture, it can be applied to any GAN setup. In the paper we e.g. used feed-forward nets (Section 5.1), convolutional nets, and ResNets (Section 5.2). We will provide further details on the different architectures to make this more clear.

References

[1] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[2] Satinder P Singh, Michael J Kearns, and Yishay Mansour. Nash convergence of gradient dynamics in general-sum games. In *UAI*, pages 541–548, 2000.