We are grateful for all the reviewers' valuable suggestions and questions. We start by showing some additional experiments for deep nonlinear ResNets. Consider a nonlinear ResNet $f(x;\theta) := w^T z_L$ with z_L recursively defined as

$$z_0 = V_0 x; \quad z_l = z_{l-1} + U_l \sigma(V_l z_{l-1}), \quad l = 1, \dots, L$$

where $V_0 \in \mathbb{R}^{D\times d}, U_l \in \mathbb{R}^{D\times m}, V_l \in \mathbb{R}^{m\times D}$ and $w \in \mathbb{R}^D$. We test two initializations: (1) standard Xavier initialization; (2) modified zero-asymmetry(mZAS) initialization : $U_l = 0, w = 0$ and $(V_l)_{i,j} \sim \mathcal{N}(0, 1/D)$. The experiments are conducted on Fashion-MNIST, where we select 1000 training samples forming the new training set to speed up the computation. The results are displayed in Figure 1.

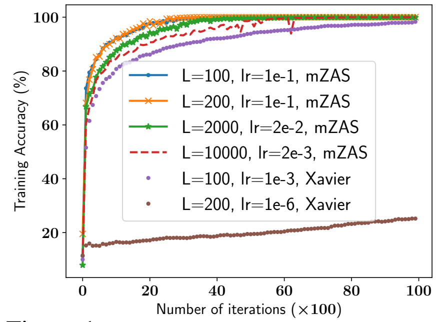

Figure 1: The comparison of training curves between two initializations. The learning rate is manually tuned to achieve the best convergence performance. The curves of GD with Xavier initialization for L=2000,10000 are not shown, since they always blow up.

We can see that mZAS initialization always outperforms the Xavier initialization. Moreover, GD with mZAS initialization is able to successfully optimize a 10000-layer ResNet. It is clearly demonstrated that ZAS-type initialization can be helpful for optimizing deep nonlinear ResNets. However, to make this initialization practical for real scenarios such as ImageNet still requires more efforts, which is beyond the scope of this paper. We will add a section in the paper to discuss how to adapt it for nonlinear residual network and provide some preliminary experiments.

For Reviewer 2: **(1)** (for continuous-time GD: $R(t) \leq \exp(-2\alpha^{2(L-1)}t)R(0)$.) Thanks to the reviewer for pointing out this interesting phenomenon that we did not notice it before. It increases the gradient by $\alpha^{2(L-1)}$ times so the faster rate holds for continuous-time GD. However, discrete-time GD $R(t) < (1-\alpha^{2(L-1)}\eta/2)^t R(0)$ requires $\eta \lesssim 1/\alpha^{2(L-1)}$, thus the number of iterations may not exponentially decrease. **(2)** (general case of rectangular weight matrices) This can be achieved by padding zeros as long as $\min\{d_1, \dots, d_{L-2}\} \geq \min\{d_0, d_{L-1}\}$. Actually, Our analysis works for a general initialization. Let $m = \min\{d_{L-1}, d_0\}$ and $A = U\Sigma V$ where $U \in \mathbb{R}^{d_{L-1}\times m}, V \in \mathbb{R}^{d_0 \times m}, \Sigma \in \mathbb{R}^{m\times m}$ be the singular value decomposition of A. We can initialize

$$W_L = 0; W_{L-1} \simeq U\Sigma^{1/(L-1)}, W_{L-2} \simeq \Sigma^{1/(L-1)}, \dots, W_2 \simeq \Sigma^{1/(L-1)}, W_1 \simeq \Sigma^{1/(L-1)}V, \tag{2}$$

where the symbol "$\simeq$" stands for equality up to zero-valued paddings. This initialization is similar as the Procedure 1 in (Arora et al. ICLR2019), but with the top layer to be zero.

For Reviewer 3: **(1)** (all layers are $d \times d$) Our proof only relies on the *dynamic invariance* and top layer to be zero. So the result also holds for the general case, $\min\{d_1, \dots, d_{L-1}\} \geq \min\{d_0, d_L\}$, in which the matrices are not necessarily square. We will clarify this in the revised version. **(2)** (direct consequence of a known alignment property...) This known property actually has been widely used in the previous works (Bartlett et al. ICML2018, Arora et al. ICLR2019, Shamir COLT2019, Du et al. ICML 2019) for analyzing the optimization of linear networks, but coming up with the right initialization to fully utilize this property is not straightforward. That is why the global convergence of GD for general linear networks has not been established until this submission. Especially, the picture of symmetry break behind the ZAS initialization could be useful for analyzing linear networks in other setting, such as matrix factorization, binary classification, etc.. **(3)** (empirical results on how this specific initialization may help in practice) We have conducted some experiments, and please refer to the beginning of this rebuttal for the results. **(4)** (results on deep linear nets (e.g. Ji and Telgarsky, 2019)) Thanks for pointing out this reference and we will add it to the related work section. This work studies the properties of solutions that the GD converges to, without providing any convergence rate. The ZAS initialization might help to establish the convergence in their setting.

For Reviewer 4: **(1)** (...the development of a new initialization method is useful unless its shown to be competitive on real problems) We have done some experiments for real problems (given in the beginning of this reresponse). and the results suggest that the ZAS-type initialization is useful for nonlinear ResNets in practice. However, we want to stress that the goal of this submission is to provide theoretical understanding of the optimization of deep linear nets. The ZAS initialization is proposed to obtain a global convergence guarantee of GD for optimizing deep linear nets. **(2)** (more convincing in any case to show these curves for multiple runs...) Please see right figure, which shows results of multiple runs. We will add it in the revised version. **(3)** (The paper is rather poorly written) We apologize for the confusion caused by the writting. We will improve it in the revised version.

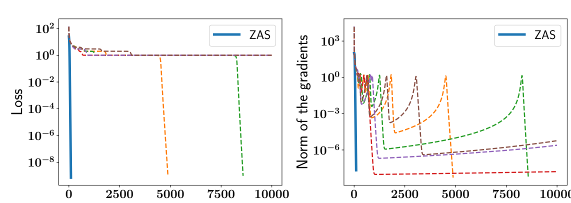

Figure 2: The five dashed lines correspond to the multiple runs of GD with the Xavier initialization. It is shown that GD successfully escape the saddle region for only 2 out 5 times in the given number of iterations.