# On the Explicit Role of Initialization on the Convergence and Implicit Bias of Overparametrized Linear Networks

**Hancheng Min** [1 2] **Salma Tarmoun** [1 3] **René Vidal** [1 4] **Enrique Mallada** [1 2]

## Abstract

Neural networks trained via gradient descent with random initialization and without any regularization enjoy good generalization performance in practice despite being highly overparametrized. A promising direction to explain this phenomenon is to study how initialization and overparametrization affect convergence and implicit bias of training algorithms. In this paper, we present a novel analysis of single-hidden-layer linear networks trained under gradient flow, which connects initialization, optimization, and overparametrization. Firstly, we show that the squared loss converges exponentially to its optimum at a rate that depends on the level of imbalance of the initialization. Secondly, we show that proper initialization constrains the dynamics of the network parameters to lie within an invariant set. In turn, minimizing the loss over this set leads to the min-norm solution. Finally, we show that large hidden layer width, together with (properly scaled) random initialization, ensures proximity to such an invariant set during training, allowing us to derive a novel non-asymptotic upper-bound on the distance between the trained network and the min-norm solution.

## 1. Introduction

Neural networks have shown excellent empirical performance in many application domains such as vision (Krizhevsky et al., 2012; Rawat & Wang, 2017), speech (Hinton et al., 2012; Graves et al., 2013) and video games (Silver et al., 2016; Vinyals et al., 2017). Among the many unexplained puzzles behind this success is the fact that gradient descent with random initialization, and without explicit regularization, enjoys good generalization performance despite being highly overparametrized.

One possible explanation of such phenomenon is the implicit bias or regularization that first order gradient algorithms induce under proper initialization assumptions. For example, in classification tasks, gradient descent on separable data can induce a bias towards the max-margin solution (Soudry et al., 2018; Ji & Telgarsky, 2019; Lyu & Li, 2019). Similarly, in regression tasks, it has been shown that (deep) matrix factorization models trained by first order methods yield solutions with low nuclear norm (Gunasekar et al., 2017) and low rank (Arora et al., 2019a). Along the same vein, Saxe et al. (2014); Gidel et al. (2019) have shown that deep linear networks sequentially learn dominant singular values of the the input-output correlation matrix.

Another possible explanation is that, in the Neural Tangent Kernel (NTK) regime, the gradient flow of a randomly initialized infinitely wide neural network can be well approximated by the flow of its linearization at initialization (Jacot et al., 2018; Chizat et al., 2019; Arora et al., 2019c;b). In this regime, training infinitely wide neural networks mimics kernel methods. In particular, the NTK flow is constrained to lie on a manifold, which improves generalization performance as discussed in  (Arora et al., 2019b).

While the aforementioned analysis is quite insightful, it requires assumptions on the model and the initialization that are often disconnected. For example, the implicit bias characterized in (Gunasekar et al., 2017; Arora et al., 2019a) requires vanishing initialization, while the analysis of convergence of gradient algorithms for linear networks requires balanced (Arora et al., 2018b;a) or spectral (Saxe et al., 2014; Gidel et al., 2019) initialization. Similarly, the NTK regime (Jacot et al., 2018; Arora et al., 2019c), requires random initialization and infinitely wide networks, making the non-asymptotic analysis challenging (Arora et al., 2019c).

This paper aims to bridge some of these gaps. We present a novel analysis of the gradient flow dynamics of overparametrized single-hidden layer linear networks, which provides a common set of conditions on initialization that lead to convergence and implicit bias. Specifically, we exploit a certain measure of imbalance at initialization to

[1] Mathematical Institute for Data Science, [2] Department of Electrical and Computer Engineering, [3] Department of Applied Mathematics and Statistics, [4] Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD, U.S.A. Correspondence to: Hancheng Min <hanchmin@jhu.edu>.

*Proceedings of the $38^{th}$ International Conference on Machine Learning*, PMLR 139, 2021. Copyright 2021 by the author(s).

ensure exponential convergence. We further characterize a complementary condition, based on orthogonality, that enforces the learning trajectory to be constrained within an invariant set whose unique global optimum is the min-norm solution. While our analysis does not require infinite width, vanishing, spectral, or random initialization, we show that our exponential convergence and orthogonality conditions are provably approximately satisfied for wide networks with properly scaled random initialization, leading to a bound on the distance to the min-norm solution. Hence, this paper formally connects initialization, exponential convergence of the optimization task, overparametrization and implicit bias.

This paper makes the following contributions:

1. In Section 2 we develop our convergence analysis based on the fact that gradient flow on the squared-$l_2$ loss preserves a certain matrix-valued quantity, akin to constants of motion in mechanics or conservation laws of physics, that measures the *imbalance* of the network weights. We show that, some level of imbalance, measured by certain eigenvalues of the imbalance matrix and defined at initialization, is sufficient to guarantee the exponential rate of convergence of the loss. Our analysis complements prior work on convergence analysis discussed in Section 1.1.

2. In Section 3.1 we show the existence of a subset of the parameter space defined by an orthogonality condition, which is invariant under gradient flow. All trajectories within this invariant set lead to a unique minimizer (w.r.t. the end-to-end function), which corresponds to the min-norm solution. As a result, initializing the network within this invariant set always yields the min-norm solution upon convergence.

3. In Section 3.2 we further show that by randomly initializing the network weights using $\mathcal{N}(0, 1/h^{2\alpha})$ (where $h$ is the hidden layer width and $1/4 < \alpha \leq 1/2$), one can approximately satisfy both our sufficient imbalance and orthogonality conditions with high probability. Notably, initializations outside the invariant set require exponential convergence to control their deviation from the set. For linear networks our results also provide a novel non-asymptotic upper-bound on the operator norm distance between the trained network and the min-norm solution.

### 1.1. Other Related Work

**Convergence of Linear Networks**. Convergence in overparametrized linear networks has been studied for both gradient flow (Saxe et al., 2014) and gradient descent (Bartlett et al., 2018; Arora et al., 2018a;b). Saxe et al. (2014) analyze the trajectory of network parameters under spectral initialization, while Bartlett et al. (2018) study the case of identity initialization. Although the fact that the imbalance is conserved under gradient flow has been exploited in Arora et al. (2018a;b), the work studies balanced initialization and exploits the structure conveyed by it to study convergence of the learning dynamics. The analysis of convergence in the imbalanced case was recently studied in Tarmoun et al. (2021) for both spectral and non-spectral initializations. Our analysis improves upon prior work by considering a wider range of imbalance structures, which include ones that arise under random initialization.

**Wide Neural Networks**. There has been a rich line of research that studies the convergence (Du et al., 2019b;a; Du & Hu, 2019; Allen-Zhu et al., 2019b) and generalization (Allen-Zhu et al., 2019a; Arora et al., 2019b;c; Li & Liang, 2018; Cao & Gu, 2019; Buchanan et al., 2020) of wide neural networks with random initialization. The behavior of such networks in their infinite width limit can be characterized by the *Neural Tangent Kernel* (NTK) (Jacot et al., 2018). Heuristically, training wide neural networks can be approximately viewed as kernel regression under gradient flow/descent (Arora et al., 2019c), hence the convergence and generalization can be understood by studying the non-asymptotic results regarding the equivalence of finite width networks to their infinite limit (Du et al., 2019b;a; Allen-Zhu et al., 2019b; Arora et al., 2019b;c; Buchanan et al., 2020). More generally, such non-asymptotic results are related to the "lazy training" (Chizat et al., 2019; Du et al., 2019a; Allen-Zhu et al., 2019b), where the network weights do not deviate too much from its initialization during training. Our results for wide linear networks presented in Section 3.2 do not follow the NTK analysis, but provide an alternative view on the effect of random initialization for linear networks when the hidden layer is sufficiently wide.

### 1.2. Notation

For a matrix $A$, we let $A^T$ denote its transpose, $\mathrm{tr}(A)$ denote its trace, $\lambda_i(A)$ and $\sigma_i(A)$ denote its $i$-th eigenvalue and $i$-th singular value, respectively, in decreasing order (when adequate). We let $[A]_{ij}$, $[A]_{i,:}$, and $[A]_{:,j}$ denote the $(i,j)$-th element, the $i$-th row and the $j$-th column of $A$, respectively. We also let $\|A\|_2$ and $\|A\|_F$ denote the spectral norm and the Frobenius norm of $A$, respectively. For a scalar-valued or matrix-valued function of time, $F(t)$, we let $\dot{F} = \dot{F}(t) = \frac{d}{dt}F(t)$ denote its time derivative. Additionally, we let $I_n$ denote the identity matrix of order $n$ and $\mathcal{N}(\mu, \sigma^2)$ denote the normal distribution with mean $\mu$ and variance $\sigma^2$.

## 2. Convergence Analysis for Gradient Flow on Single-Hidden-Layer Linear Networks

We first study the convergence of gradient flow for single-hidden-layer linear networks trained with squared $l_2$-loss. Given $n$ training samples $\{x^{(i)}, y^{(i)}\}_{i=1}^n$, where $x^{(i)} \in \mathbb{R}^D$,

$y^{(i)} \in \mathbb{R}^m$, we aim to solve the linear regression problem

$$\min_{\Theta \in \mathbb{R}^{D\times m}} \mathcal{L} = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \Theta^T x^{(i)})^2 \,. \tag{1}$$

We do so by training a single-hidden-layer linear network $y = f(x; V, U) = VU^T x$, $V = \mathbb{R}^{m\times h}$, $U \in \mathbb{R}^{D\times h}$, where $h$ is the hidden layer width, with gradient flow, i.e., gradient descent with "infinitesimal step size". We consider an *overparametrized* model such that $h \geq \min\{m, D\}$.

We rewrite the loss with respect to our parameters $V, U$ as

$$\mathcal{L}(V,U) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - VU^T x^{(i)})^2 = \frac{1}{2}\|Y - XUV^T\|_F^2 \,, \tag{2}$$

where $Y = [y^{(1)}, \cdots, y^{(n)}]^T$ and $X = [x^{(1)}, \cdots, x^{(n)}]^T$.

Assuming the input data $X$ has full rank, we consider the under-determined case $D > n \geq \operatorname{rank}(X)$ for our regression problem, i.e., there are infinitely many solutions $\Theta^*$ that achieve optimal loss $\mathcal{L}^*$ of (1). We will show that under certain conditions, the trajectory of the loss function $\mathcal{L}(t) = \mathcal{L}(V(t), U(t))$ under gradient flow of (2), i.e.,

$$\dot V(t) = -\frac{\partial \mathcal{L}}{\partial V}(V(t), U(t)) \,, \dot U(t) = -\frac{\partial \mathcal{L}}{\partial U}(V(t), U(t)) \,, \tag{3}$$

converges to $\mathcal{L}^*$ exponentially, and that proper initialization of $U(0), V(0)$ controls the convergence rate via a time-invariant matrix-valued term, the *imbalance* of the network.

### 2.1. Reparametrization of Gradient Flow

Assuming that $D > n \geq r = \operatorname{rank}(X)$, the singular value decomposition (SVD) of $X$ can be written as

$$X = W \begin{bmatrix} \Sigma_x^{1/2} & 0 \end{bmatrix} \begin{bmatrix} \Phi_1^T \\ \Phi_2^T \end{bmatrix} \,, \tag{4}$$

where $W \in \mathbb{R}^{n\times r}$, $\Phi_1 \in \mathbb{R}^{D\times r}$, and $\Phi_2 \in \mathbb{R}^{D\times (D-r)}$. Since $\Phi_1\Phi_1^T + \Phi_2\Phi_2^T = I_D$, we have

$$U = I_D U = (\Phi_1\Phi_1^T + \Phi_2\Phi_2^T)U = \Phi_1\Phi_1^T U + \Phi_2\Phi_2^T U \,,$$

and hence we can reparametrize $U$ as $(U_1, U_2)$ using the bijection $U = \Phi_1 U_1 + \Phi_2 U_2$, with inverse $(U_1, U_2) = (\Phi_1^T U, \Phi_2^T U)$.

We write the gradient flow in (3) explicitly as

$$\begin{aligned} \dot V(t) =& \left(Y - XU(t)V^T(t)\right)^T XU(t) \\ =& E^T(t)\Sigma_x^{1/2}\Phi_1^T U(t) \,, \end{aligned} \tag{5a}$$

$$\begin{aligned} \dot U(t) =& X^T \left(Y - XU(t)V^T(t)\right) V(t) \\ =& \Phi_1 \Sigma_x^{1/2} E(t) V(t) \,, \end{aligned} \tag{5b}$$

where

$$E = E(V, U_1) = W^T Y - \Sigma_x^{1/2} U_1 V^T \,, \tag{6}$$

is defined to be the *error*. Then from (5a)(5b) we obtain the dynamics in the parameter space $(V, U_1, U_2)$ as

$$\begin{aligned} \dot V(t) =& E^T(t)\Sigma_x^{1/2} U_1(t) \,, \\ \dot U_1(t) =& \Sigma_x^{1/2} E(t) V(t) \,,\ \dot U_2(t) = 0 \,. \end{aligned} \tag{7}$$

Notice that

$$\begin{aligned} \mathcal{L}(V,U) =& \frac{1}{2}\|Y - XUV^T\|_F^2 \\ =& \frac{1}{2}\|(I - WW^T)Y + WE\|_F^2 \\ =& \frac{1}{2}\|WE\|_F^2 + \frac{1}{2}\|(I - WW^T)Y\|_F^2 \\ =& \frac{1}{2}\|E\|_F^2 + \frac{1}{2}\|(I - WW^T)Y\|_F^2 \,, \end{aligned} \tag{8}$$

where the last equality is because $W$ has orthonormal columns. Here the last term in (8) does not dependson $V, U$, and we define it as the residual

$$\mathcal{L}^* = \frac{1}{2}\|(I - WW^T)Y\|_F^2 \,,$$

which is the optimal value of (1). Therefore it suffices to analyze the convergence of the error $E(t)$ under the dynamics of $V(t), U_1(t)$ in (7). As we show later in Section 3.2, the exponential convergence of $E(t)$, or equivalently $\mathcal{L}(t) - \mathcal{L}^*$ is crucial for our analysis of the implicit bias, in the sense that exponential convergence ensures that the parameters do not deviate much away from the invariant set of interest, so that good properties from the initialization are approximately preserved during training.

### 2.2. Imbalance and Convergence of the Error

We define the *imbalance* of the single-hidden-layer linear network under input data $X$ as

$$\textit{Imbalance}:\ \Lambda = U_1^T U_1 - V^T V \in \mathbb{R}^{h\times h} \,. \tag{9}$$

The imbalance term is time-invariant under gradient flow, as stated in the following claim.

**Claim.** *Under the continuous dynamics* (7), $\frac{d}{dt}\Lambda(t) \equiv 0$.

*Proof.* Under (7), we compute the time derivative of $U_1^T(t)U_1(t)$ and $V^T(t)V(t)$ as

$$\begin{aligned} \frac{d}{dt}U_1^T(t)U_1(t) =& \dot U_1^T(t)U_1(t) + U_1^T(t)\dot U_1(t) \\ &= V^T(t)E^T(t)\Sigma_x^{1/2}U_1(t) + U_1^T(t)\Sigma_x^{1/2}E(t)V(t), \\ \frac{d}{dt}V^T(t)V(t) =& V^T(t)\dot V(t) + \dot V^T(t)V(t) \\ &= V^T(t)E^T(t)\Sigma_x^{1/2}U_1(t) + U_1^T(t)\Sigma_x^{1/2}E(t)V(t) \,. \end{aligned}$$

Hence, $\frac{d}{dt}U_1^T(t)U_1(t)$ and $\frac{d}{dt}V^T(t)V(t)$ are identical, and so $\frac{d}{dt}\Lambda(t) = \frac{d}{dt}[U_1^T(t)U_1(t) - V^T(t)V(t)] \equiv 0$. □

The rank of the $h \times h$ imbalance matrix, $\mathrm{rank}(\Lambda) \leq m + r$, characterizes how much the row spaces of $U_1$ and $V$ are misaligned. As stated in the following theorem (We refer readers to Min et al. (2021) for the proof), a mild condition on the imbalance is sufficient for exponential convergence of the error $E(t)$, or equivalently, $\mathcal{L}(t) - \mathcal{L}^*$. The condition measures the amount of imbalance by the scalar $c := [\lambda_r(\Lambda(0)]_+ + [\lambda_m(-\Lambda(0)]_+$, where $[\cdot]_+ := \max\{\cdot, 0\}$. Since $\lambda_r(\Lambda(0))$ (or $\lambda_m(-\Lambda(0))$) is undefined when $h < r$ (or $h < m$), we define it as zero.

**Theorem 1** (Convergence of linear networks with sufficient level of imbalance)**.** *Let $V(t), U_1(t), t > 0$ be the solution of* (7) *starting from some $V(0), U_1(0)$. We have that*

$$(\mathcal{L}(t) - \mathcal{L}^*) \leq \exp\left(-2\lambda_r(\Sigma_x)ct\right)(\mathcal{L}(0) - \mathcal{L}^*)),\ \forall t > 0\,, \tag{10}$$

*where $c := [\lambda_r(\Lambda(0)]_+ + [\lambda_m(-\Lambda(0)]_+$. Moreover, if $h \geq m + r$, then $c = \lambda_r(\Lambda(0)) + \lambda_m(-\Lambda(0))$. Additionally, if $c > 0$, $(V(t), U_1(t)), t > 0$ converges to an equilibrium point $(V(\infty), U_1(\infty))$ such that $E(V(\infty), U_1(\infty)) = 0$.*

Notice that in the "balanced" case ($c = 0$) the condition in (10) is trivial and no convergence guarantee is obtained. The interesting case is when $c > 0$, which implies that the initial weights $U(0), V(0)$ are nonzero and somehow "imbalanced". Notably, imbalance arises for example when 1) $U_1(0), V(0)$ are full rank, and their row spaces are sufficiently misaligned. The extreme case is that $U_1(0)V^T(0) = 0$, where we have $c = \lambda_r(U_1U_1^T) + \lambda_m(VV^T) > 0$. Later we show that this orthonality condition is approximately satisfied for wide networks under random initialization; 2) The row spaces of $U_1(0), V(0)$ are well-aligned, and the singular values of $U(0)$ are sufficiently larger than those of $V(0)$ (or vice versa). In these cases, either $\lambda_r(\Lambda(0))$ or $\lambda_m(-\Lambda(0))$ is non-zero.

As mentioned before, the fact that the imbalance is preserved under gradient flow has been exploited in Arora et al. (2018a;b), where imbalance is assumed to be zero (or small), such that the learning dynamics can be expressed in closed form with respect to the end-to-end matrix. This analysis, requires, however, additional assumptions on the initialization of the end-to-end matrix for exponential convergence. Similarly, though in a more general setting, Du et al. (2018) showed that the imbalance is preserved, and proves convergence under a small imbalance assumption. Exponential rate, however, is not guaranteed. Exploiting imbalance for guaranteeing convergence was first presented in Saxe et al. (2014), under a spectral initialization assumption, where the exact learning trajectory can be computed. The analysis of convergence in the imbalanced case was recently studied in Tarmoun et al. (2021) for both spectral, and non-spectral initializations with additional requirement that the imbalance matrix has all eigenvalues being equal. In contrast, Theorem 1, which complements aforementioned works, shows convergence result without the spectral nor balanced initialization condition, and it has less constraints on the imbalance structure. Lastly, we note that the initialization condition in previous works (Saxe et al., 2014; Arora et al., 2018a; Du et al., 2018), either spectral or balanced initialization, is not satisfied by randomly initialized weights with a non-vanishing scale, while our result do provide convergence guarantees for random initialization.

## 3. Implicit Bias of Gradient Flow on Single-Hidden-Layer Linear Network

In this section, we study a particular type of implicit bias of single-hidden-layer linear networks under gradient flow. Assuming that $D > r = \mathrm{rank}(X)$, the regression problem (1) has infinitely many solutions $\Theta^*$ that achieve optimal loss. Among all these solutions, one that is of particular interest in high-dimensional linear regression is the *minimum norm solution* (min-norm solution)

$$\begin{aligned}\hat{\Theta} &= \arg\min_{\Theta \in \mathbb{R}^{D\times m}}\{\|\Theta\|_F : \|Y - X\Theta\|_F^2 = \min_{\Theta}\|Y - X\Theta\|_F^2\} \\ &= X^T(XX^T)^{\dagger}Y,\end{aligned} \tag{11}$$

which has near-optimal generalization error for suitable data models (Bartlett et al., 2020; Mei & Montanari, 2019). Here, we study conditions under which our trained network is equal or close to the min-norm solution by showing how the initialization explicitly controls the trajectory of the training parameters to be exactly (or approximately) confined within some low-dimensional invariant set. In turn, minimizing the loss over this set leads to the min-norm solution.

### 3.1. Decomposition of Trained Network

Notice that the end-to-end matrix $UV^T \in \mathbb{R}^{D\times m}$ associated with the single-hidden-layer linear network can be decomposed according to the SVD of data matrix $X$, (4), as

$$UV^T = (\Phi_1\Phi_1^T + \Phi_2\Phi_2^T)UV^T = \Phi_1U_1V^T + \Phi_2U_2V^T\,, \tag{12}$$

where $\Phi_1, \Phi_2, U_1, U_2$ are defined in Section 2. The $j$-th column of $UV^T$, $[UV^T]_{:,j}$, is the linear predictor for the $j$-th output $y_j$, and is decomposed into two components within complementary subspaces $\mathrm{span}(\Phi_1)$ and $\mathrm{span}(\Phi_2)$. Moreover $[U_1V^T]_{:,j}$ is the coordinate of $[UV^T]_{:,j}$ w.r.t. the orthonormal basis consisting of the columns of $\Phi_1$, and similarly $[U_2V^T]_{:,j}$ is the coordinate w.r.t. basis $\Phi_2$. Under gradient flow (3), the trajectory $(U(t)V(t)^T, t > 0)$ is fully determined by the trajectory $(U_1(t)V^T(t), U_2(t)V^T(t), t > 0)$, which is governed by the dynamics (7).

**Convergence of Training Parameters**. We have derived useful results regarding $U_1(t)V^T(t)$ for $t > 0$ in Section 2. By Theorem 1, provided sufficient level of imbalance, $U_1(t)V^T(t)$ converges to some $U_1(\infty)V^T(\infty)$ and the stationary point satisfies $W^TY - \Sigma_x^{1/2}U_1(\infty)V^T(\infty) = 0$, which implies $U_1(\infty)V^T(\infty) = \Sigma_x^{-1/2}W^TY$. Then it is easy to check that

$$\begin{aligned}\Phi_1U_1(\infty)V^T(\infty) =&\ \Phi_1\Sigma_x^{-1/2}W^TY\\ =&\ X^T(XX^T)^\dagger Y = \hat{\Theta}\,. \qquad (13)\end{aligned}$$

For $U_2(t)V^T(t)$, notice that $\dot{U}_2(t) = 0$ in dynamics (7), hence $U_2(t) = U_2(0), \forall t > 0$. Overall, under sufficient level of imbalance, $U(t)V^T(t)$ converges to some $U(\infty)V^T(\infty)$ and

$$\begin{aligned}U(\infty)V^T(\infty) =&\ \Phi_1U_1(\infty)V^T(\infty) + \Phi_2U_2(0)V^T(\infty)\\ =&\ \hat{\Theta} + \Phi_2U_2(0)V^T(\infty)\,. \qquad (14)\end{aligned}$$

**Constrained Training via Initialization**. Based on our analysis above, initializing $U_2(0)$ such that $U_2(0)V^T(\infty) = 0$ in the limit, guarantees convergence to the min-norm solution via (14). However, this is not easily achievable, as one needs to know a priori $V(\infty)$. Instead, we can show that by choosing a proper initialization, one can constrain the trajectory of the matrix $U(t)V^T(t)$ to lie identically in the set $\Phi_2^TU_2(t)V^T(t) \equiv 0$ for all $t \geq 0$, thus the min-norm solution is obtained upon convergence, as suggested by the following proposition.

**Proposition 1.** *Let $V(t), U_1(t), U_2(t), t > 0$ be the solution of (7) starting from some $V(0), U_1(0), U_2(0)$. Assuming $V(t), U_1(t), t > 0$ converges to some equilibrium point $V(\infty), U_1(\infty)$ with $E(V(\infty), U_1(\infty)) = 0$. If the initialization satisfies*

$$V(0)U_2^T(0) = 0,\ U_1(0)U_2^T(0) = 0\,, \qquad (15)$$

*then we have*

$$U(\infty)V^T(\infty) = \hat{\Theta}\,.$$

*Proof.* From (7) we have

$$\begin{aligned}&\frac{d}{dt}\begin{bmatrix}V(t)U_2^T(0)\\ U_1(t)U_2^T(0)\end{bmatrix}\\ =&\begin{bmatrix}0 & E^T(t)\Sigma_x^{1/2}\\ \Sigma_x^{1/2}E(t) & 0\end{bmatrix}\begin{bmatrix}V(t)U_2^T(0)\\ U_1(t)U_2^T(0)\end{bmatrix}. \qquad (16)\end{aligned}$$

Since $V(0)U_2^T(0) = 0,\ U_1(0)U_2^T(0) = 0$ is an equilibrium point of (16), we have $V(t)U_2^T(0) = 0, \forall t \geq 0$, hence $V(\infty)U_2^T(0) = 0$. From (14) we conclude that $U(\infty)V^T(\infty) = \hat{\Theta}$. □

In the standard linear regression, where $\Theta(t)$ follows the gradient flow on $\mathcal{L}(\Theta) = \frac{1}{2}\|Y - X\Theta\|_F^2$, it is well-known that if the columns of $\Theta(0)$ are initialized in $\mathrm{span}(\Phi_1)$, namely $\Theta^T(0)\Phi_2 = 0$, then $\Theta(\infty) = \hat{\Theta}$. Proposition 1 is the extension of such results to the overparameterized setting. It is worth-noting that initializing the columns of $U(0)V^T(0)$ in $\mathrm{span}(\Phi_1)$, namely $V(0)U_2^T(0) = 0$ is no longer sufficient for obtaining $\hat{\Theta}$ as the trained network, and additional condition $U_1(0)U_2^T(0) = 0$ is required.

Here the orthogonality constraints (15) define an invariant subset of the parameter space $\{V, U : VU_2^T = 0, U_1U_2^T = 0\}$ under the gradient flow. Proposition 1 shows that given an initialization within the invariant set, the trained network (after convergence) is exactly the min-norm solution, which is the only minimizer in the invariant set.

While in practice we can make the initialization exactly as above, such choice is data-dependent and requires the SVD of the data matrix $X$. Moreover, we note that while the zero initialization works for the standard linear regression case, such initialization $V(0) = 0, U(0) = 0$ is bad in the overparametrized case because it is an equilibrium point of the gradient flow, even though it satisfies the orthogonal condition $V(0)U_2^T(0) = 0$ and $U_1(0)U_2^T(0) = 0$.

In the next section, we show that under (properly scaled) random initialization and sufficiently large hidden layer width $h$, both conditions for convergence and implicit bias on initialization are probably approximately satisfied, i.e., with high probability the level of imbalance is sufficient for exponential convergence, and the parameters are initialized close to the invariant set, allowing us to obtain a non-asymptotic bound between the trained network and the min-norm solution.

### 3.2. Wide Single-Hidden-Layer Linear Network

In this section, we show how the previously mentioned conditions for convergence and implicit bias, i.e., high imbalance and orthogonality, are approximately satisfied with high probability under the following initialization $(1/4 \leq \alpha \leq 1/2)$

$$\begin{aligned}&[U(0)]_{ij} \sim \mathcal{N}\left(0, \frac{1}{h^{2\alpha}}\right),\ 1 \leq i \leq D, 1 \leq j \leq h\,,\\ &[V(0)]_{ij} \sim \mathcal{N}\left(0, \frac{1}{h^{2\alpha}}\right),\ 1 \leq i \leq m, 1 \leq j \leq h\,,\end{aligned}$$

where all the entries are independent.

Both our parametrization and initialization are, at first sight, different from the one used in previous works (Jacot et al., 2018; Du & Hu, 2019; Arora et al., 2019c) on NTK analysis for wide neural networks. We note that with time-rescaling, however, we can relate our initialization to the one in Arora et al. (2019c). Please see Appendix B for a comparison.

Recall in the last section, one can obtain exactly min-norm solution via proper initialization of the single-hidden-layer network. In particular, it requires 1) convergence of the error $E(t)$ to zero; and 2) the orthogonality conditions $V(0)U_2^T(0) = 0$ and $U_1(0)U_2^T(0) = 0$. Under random initialization and sufficiently large hidden layer width $h$, these two conditions are approximately satisfied. Using basic random matrix theory, one can show the following lemma. (We refer readers to Min et al. (2021) for the proof)

**Lemma 1.** *Let $\frac{1}{4} < \alpha \leq \frac{1}{2}$. Given data matrix $X$. $\forall \delta \in (0,1)$, $\forall h > h_0 = poly\left(m, D, \frac{1}{\delta}\right)$, with probability at least $1-\delta$ over random initializations with $[U(0)]_{ij}, [V(0)]_{ij} \sim \mathcal{N}(0, h^{-2\alpha})$, the following conditions hold:*

1. *(Sufficient level of imbalance)*

$$\lambda_r(\Lambda(0)) + \lambda_m(-\Lambda(0)) > h^{1-2\alpha}\,, \qquad (17)$$

2. *(Approximate orthogonality)*

$$\left\|\begin{bmatrix} V(0)U_2^T(0) \\ U_1(0)U_2^T(0)\end{bmatrix}\right\|_F \leq 2\sqrt{m+r}\frac{\sqrt{m+D}+\frac{1}{2}\log\frac{2}{\delta}}{h^{2\alpha-\frac{1}{2}}}\,, \qquad (18)$$

$$\left\|U_1(0)V^T(0)\right\|_F \leq 2\sqrt{m}\frac{\sqrt{m+D}+\frac{1}{2}\log\frac{2}{\delta}}{h^{2\alpha-\frac{1}{2}}}\,. \qquad (19)$$

From (18), we know that the parameters are initialized close to the invariant set of our interest, as measured by $\|VU_2^T\|_F + \|U_1U_2^T\|_F$. The dynamics (16) quantify at time $t$ how fast this measure can maximally increase given that its current value is non-zero. It is clear that the smaller norm the current error $E(t)$ has, the lower is the rate at which this measure could increase. This suggests that as long as the error converges sufficiently fast, $\|VU_2^T\|_F + \|U_1U_2^T\|_F$ will not increase too much from its initial value. For our purpose, as the width $h$ increases, we need at least a constant rate of exponential convergence of the error (given by (17)), and an initial error $E(0)$ that is bounded by some constant (derived from (19)). With these conditions satisfied with high probability, we have the following Theorem regarding the implicit bias of wide linear networks. (We refer readers to Min et al. (2021) for the proof)

**Theorem 2** (Implicit bias of wide single-hidden-layer linear network for regression)**.** *Let $\frac{1}{4} < \alpha \leq \frac{1}{2}$. Let $(V(t), U(t), t > 0)$ be a trajectory of the continuous dynamics (7). Then, $\exists C > 0$, such that $\forall \delta \in (0,1), \forall h > h_0^{1/(4\alpha-1)}$ with $h_0 = poly\left(m, D, \frac{1}{\delta}, \frac{\lambda_1(\Sigma_x)}{\lambda_r^3(\Sigma_x)}\right)$, with probability $1-\delta$ over random initializations with $[U(0)]_{ij}, [V(0)]_{ij} \sim \mathcal{N}(0, h^{-2\alpha})$, we have*

$$\|U(\infty)V^T(\infty) - \hat{\Theta}\|_2 \leq 2C^{1/h^{1-2\alpha}}\sqrt{m+r}\frac{\sqrt{m+D}+\frac{1}{2}\log\frac{2}{\delta}}{h^{2\alpha-\frac{1}{2}}}\,. \qquad (20)$$

*Here $C = \exp\left(1 + \frac{\lambda_1^{1/2}(\Sigma_x)}{\lambda_r(\Sigma_x)}\|Y\|_F\right)$, which depends on the data $X, Y$.*

Previous works (Arora et al., 2019c) show non-asymptotic results on bounding the difference of predictions between the trained network and the kernel predictor of the NTK over a finite number of testing point (non-global result) using more general network structure and activation functions. We work on a simpler model, we are able to study it without going through non-asymptotic NTK analysis, which is considerably more complicated than ours. We believe this theorem is a clear illustration of how overparametrization, in particular, in the hidden layer width, together with random initialization affects the convergence and implicit bias.

Notably, although our initialization is related to the NTK analysis (Jacot et al., 2018; Arora et al., 2019c) and the kernel regime (Chizat et al., 2019), we significantly simplify the non-asymptotic analysis with the exact charaterization of an invariant set tied to the regularized solution. Specifically, our analysis does not rely on approximating the training flow to one in the infinite width limit, or one from the linearized network at initialization. Instead, we have the exact characterization of the properties required to reach min-norm solution and show how such properties are approximately preserved during training.

## 4. Experimental Simulations

In this section, we provide numerical verification for Theorem 1 and 2, the full description of the experiments is presented in Appendix A.

**Convergence via Imbalanced Initialization**: We train the linear network using gradient descent with a fixed small step size on the averaged loss $\mathcal{L}(U,V) = \|Y - XUV\|_F^2/n$. We use the initialization $U(0) = \sigma_U U_0, V(0) = \sigma_V V_0$ for some randomly sampled $U_0, V_0$ with i.i.d. standard normal entries, and scalars $\sigma_U$, $\sigma_V$. Under this setting, we can change the relative scales of $\sigma_U, \sigma_V$ but keep their product fixed, so that we obtain initializations with different level of imbalance $c$ while keeping the initial end-to-end matrix $U(0)V^T(0)$ fixed. To eliminate the effect of ill-conditioned $\Sigma_x$ on the convergence, we have $\Sigma_x = I_r$ in this experiment.

For comparison, we also consider the balanced initialization that corresponds to the same end-to-end matrix. For a given $\Theta(0) = U(0)V^T(0)$, we choose an arbitrary $Q \in \mathbb{R}^{h\times m}$ with $Q^TQ = I_m$, then a balanced initialization is given by

$$U_{\text{balanced}}(0) = \Theta(0)\left[\Theta^T(0)\Phi_1\Phi_1^T\Theta(0)\right]^{-1/4}Q^T,$$
$$V_{\text{balanced}}(0) = \left[\Theta^T(0)\Phi_1\Phi_1^T\Theta(0)\right]^{1/4}Q\,.$$

Such initialization ensures the imbalanced is the zero matrix while keeping the end-to-end matrix as $\Theta(0)$. We note here

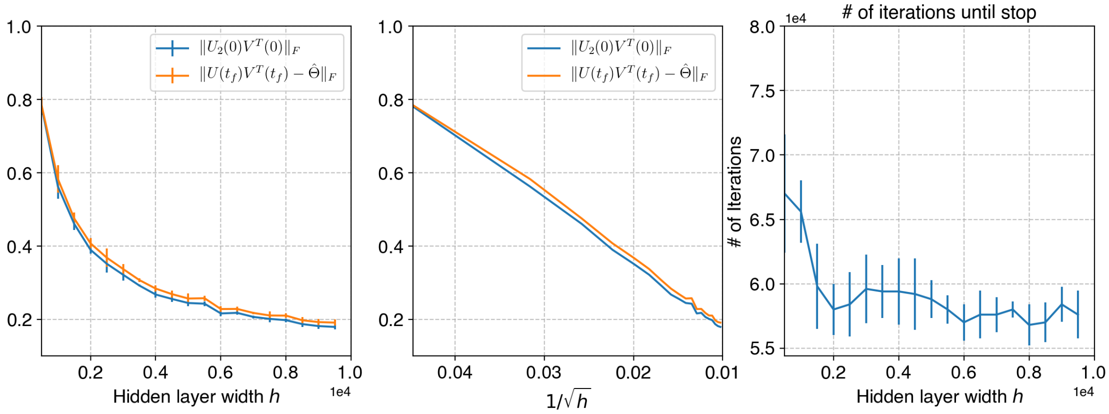

*Figure 1.* Implicit bias of wide single-hidden-layer linear network under random initialization. The line is plotting the average over 5 runs for each $h$, and the error bar shows the standard deviation. The gradient descent stops at iteration $t_f$.

the choice of $Q$ does not affect the error trajectory $E(t)$, hence the loss $\mathcal{L}(t)$.

From Fig.2, we see that given fixed step size, the convergence rate is improved as we increase the level of the imbalance at initialization and the balanced initialization is the slowest among all cases. Notably, in case 3, our bound is almost tight. For case 2, our bound is tight in characterizing the asymptotic rate of convergence. Our analysis does not provide convergence guarantee for the balanced case, while Arora et al. (2018a;b) have shown linear convergence for certain cases with zero imbalance at initialization. This suggests the need of a unified convergence analysis, that is applicable for both balanced and imbalanced initialization, to obtain tighter convergence guarantees. This is subject of current research.

Note that the goal of this experiment is to verify the improved convergence rate achieved by gradient flow initialized with a high level of imbalance. To this end, we approximate the continuous dynamics using gradient descent with a fixed small step size. However, this does not imply that one can always accelerate gradient descent by increasing the level of imbalance at initialization. This is because the step size for gradient descent is sometimes chosen to be close to the largest possible for convergence, but it is unknown how the level of imbalance affects such choice.

**Implicit Bias of Wide Linear Networks**: For the case of wide linear networks with random initialization considered in Section 3.2, when we set $\alpha = 1/2$, Theorem 2 suggests that $\|U(\infty)V^T(\infty)-\hat{\Theta}\|_F \sim \mathcal{O}(h^{-1/2})$[1]. We verify it by training linear networks with varying hidden layer width. We randomly initialize the network as in Section 3.2 and

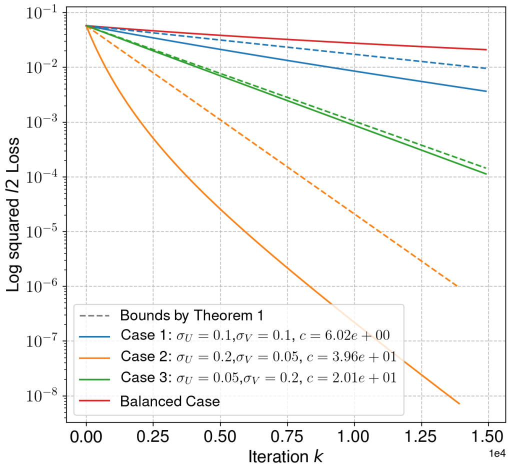

*Figure 2.* Convergence of single-hidden-layer linear networks under different level of imbalance $c$. The dashed line represents the bound provided by Theorem 1.

train it using gradient descent with a fixed small step size. The algorithm stops when the loss is below some fixed tolerance. We only vary the width $h$ (from 500 to 10000) for different experiments and repeat 5 runs for each $h$.

Fig.1 clearly shows that the distance between the trained network and the min-norm solution, $\|U(t_f)V^T(t_f)-\hat{\Theta}\|_F$, decreases as the width $h$ increases and the middle plot verifies the asymptotic rate $\mathcal{O}(h^{-1/2})$. Besides, we also plot the initial distance in $\text{span}(\Phi_2)$ between the network and the min-norm solution as

$$\|U_2(0)V^T(0)\|_F = \|\Phi_2\Phi_2^T(U(0)V^T(0)-\hat{\Theta})\|_F\,.$$

A small $\|U_2V^T\|_F$ is the exact property we want for a solution to be close to the min-norm solution. We see that the large width together with random initialization guarantees $\|U_2(0)V(0)\|_F \sim \mathcal{O}(h^{-1/2})$, and more importantly,

[1] In Theorem 2, we provide the bound for spectral norm, but the proof essentially derives the same bound for Frobenius norm. We refer readers to Min et al. (2021) for details

since the initialization does not exactly fall into the invariant set defined by (15), $\|U_2V\|_F$ will deviate from its initial value. However, the deviation is well-controlled by the fast convergence of the error, i.e. as shown in the plot, $\|U_2(t_f)V^T(t_f)\|_F \simeq \|U(t_f)V^T(t_f) - \hat{\Theta}\|_F \sim \mathcal{O}(h^{-1/2})$.

## 5. Conclusion

In this paper, we study the explicit role of initialization on controlling the convergence and implicit bias of single-hidden-layer linear networks trained under gradient flow. We first show that initialization with sufficient level of imbalance leads to the exponential convergence of the squared error loss. We then show that proper initialization enforces the trajectory of network parameters to be exactly (or approximately) constrained in a low-dimensional invariant set, over which minimizing the loss yields the min-norm solution. Combining those results, we obtain a novel non-asymptotic bound regarding the implicit bias on wide linear networks under random initialization towards the min-norm solution. Our analysis, although on a simpler overparametrized model, connects overparametrization, initialization, and optimization. We think it is promising for future research to translate some of the concepts such as the imbalance, and the constrained learning concept to multi-layer linear networks, and eventually to neural networks with nonlinear activations.

## Acknowledgements

The authors thank the support of the NSF-Simons Research Collaborations on the Mathematical and Scientific Foundations of Deep Learning (NSF grant 2031985), the NSF HDR TRIPODS Institute for the Foundations of Graph and Deep Learning (NSF grant 1934979), the NSF AMPS Program (NSF grant 1736448), and the NSF CAREER Program (NSF grant 1752362).

## References

Allen-Zhu, Z., Li, Y., and Liang, Y. Learning and generalization in overparameterized neural networks, going beyond two layers. In *Advances in neural information processing systems*, pp. 6158–6169, 2019a.

Allen-Zhu, Z., Li, Y., and Song, Z. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pp. 242–252. PMLR, 2019b.

Arora, S., Cohen, N., Golowich, N., and Hu, W. A convergence analysis of gradient descent for deep linear neural networks. In *International Conference on Learning Representations*, 2018a.

Arora, S., Cohen, N., and Hazan, E. On the optimization of deep networks: Implicit acceleration by overparameterization. In *35th International Conference on Machine Learning*, 2018b.

Arora, S., Cohen, N., Hu, W., and Luo, Y. Implicit regularization in deep matrix factorization. *Advances in Neural Information Processing Systems*, 32, 2019a.

Arora, S., Du, S., Hu, W., Li, Z., and Wang, R. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *International Conference on Machine Learning*, pp. 322–332, 2019b.

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R. R., and Wang, R. On exact computation with an infinitely wide neural net. In *Advances in Neural Information Processing Systems*, pp. 8141–8150, 2019c.

Bartlett, P., Helmbold, D., and Long, P. Gradient descent with identity initialization efficiently learns positive definite linear transformations by deep residual networks. In *International conference on machine learning*, pp. 521–530. PMLR, 2018.

Bartlett, P. L., Long, P. M., Lugosi, G., and Tsigler, A. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 2020.

Buchanan, S., Gilboa, D., and Wright, J. Deep networks and the multiple manifold problem. *arXiv preprint arXiv:2008.11245*, 2020.

Cao, Y. and Gu, Q. Generalization bounds of stochastic gradient descent for wide and deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 10836–10846, 2019.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, pp. 2937–2947, 2019.

Du, S. and Hu, W. Width provably matters in optimization for deep linear neural networks. In *International Conference on Machine Learning*, pp. 1655–1664, 2019.

Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pp. 1675–1685, 2019a.

Du, S. S., Hu, W., and Lee, J. D. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Du, S. S., Zhai, X., Poczos, B., and Singh, A. Gradient descent provably optimizes over-parameterized neural networks. In *International Conference on Learning Representations(ICLR), 2019*, 2019b.

Gidel, G., Bach, F., and Lacoste-Julien, S. Implicit regularization of discrete gradient dynamics in linear neural networks. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32, pp. 3202–3211. Curran Associates, Inc., 2019.

Graves, A., Mohamed, A.-r., and Hinton, G. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pp. 6645–6649. IEEE, 2013.

Gunasekar, S., Woodworth, B., Bhojanapalli, S., Neyshabur, B., and Srebro, N. Implicit regularization in matrix factorization. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 6152–6160, 2017.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pp. 8571–8580, 2018.

Ji, Z. and Telgarsky, M. Gradient descent aligns the layers of deep linear networks. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Li, Y. and Liang, Y. Learning overparameterized neural networks via stochastic gradient descent on structured data. In *Advances in Neural Information Processing Systems*, pp. 8157–8166, 2018.

Lyu, K. and Li, J. Gradient descent maximizes the margin of homogeneous neural networks. In *International Conference on Learning Representations*, 2019.

Mei, S. and Montanari, A. The generalization error of random features regression: Precise asymptotics and double descent curve. *arXiv preprint arXiv:1908.05355*, 2019.

Min, H., Tarmoun, S., Vidal, R., and Mallada, E. On the explicit role of initialization on the convergence and implicit bias of overparametrized linear networks. *arXiv preprint arXiv:2105.06351*, 2021.

Rawat, W. and Wang, Z. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, 2017.

Saxe, A. M., Mcclelland, J. L., and Ganguli, S. Exact solutions to the nonlinear dynamics of learning in deep linear neural network. In *International Conference on Learning Representations*, 2014.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484, 2016.

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

Tarmoun, S., França, G., Haeffele, B. D., and Vidal, R. Implicit acceleration of gradient flow in overparameterized linear models, 2021. URL https://openreview.net/forum?id=D9pSaTGUemb.

Vinyals, O., Ewalds, T., Bartunov, S., Georgiev, P., Vezhnevets, A. S., Yeo, M., Makhzani, A., Küttler, H., Agapiou, J., Schrittwieser, J., et al. Starcraft ii: A new challenge for reinforcement learning. *arXiv preprint arXiv:1708.04782*, 2017.

# A. Numerical Verification

The scale of the linear regression problem we consider in the numerical section is $D = 400$, $n = 100$, and $m = 1$.

## A.1. Convergence of single-hidden-layer linear network via imbalanced initialization

**Generating training data** The synthetic training data is generated as following:

1) For data matrix $X$, first we generate $X_0 \in \mathbb{R}^{n\times D}$ with all the entries sampled from $\mathcal{N}(0, 1)$, and take its SVD $X_0 = W\Sigma^{1/2}\Phi_1$. Then we let $X = W\Phi_1$, hence we have all the singular values of $X$ being 1. Here $r = \text{rank}(X) = n = 100$.

2) For $Y$, we first sample $\Theta \sim \mathcal{N}(0, D^{-1}I_D)$, and $\epsilon \sim \mathcal{N}(0, 0.01^2 I_n)$, then we let $Y = X\Theta + \epsilon$.

**Initialization and Training** We set the hidden layer width $h = 500$. We initialize $U(0), V(0)$ with

$$U(0) = \sigma_U U_0,\ V(0) = \sigma_V V_0, \qquad [U_0]_{ij}, [V_0]_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)\,.$$

and we consider three cases of such initialization: 1) $\sigma_U = 0.1,\ \sigma_V = 0.1$; 2) $\sigma_U = 0.5,\ \sigma_V = 0.02$; 3) $\sigma_U = 0.05,\ \sigma_V = 0.2$. Such setting ensures the initial end-to-end function are identical for all cases but with different levels of imbalance. For these three cases, we run gradient descent on the averaged loss $\tilde{L} = \frac{1}{n}\|Y - XUV^T\|_F^2$ with step size[2] $\eta = 5e - 4$.

For comparison, we also consider the balanced initialization that corresponds to the same end-to-end matrix. For a given $\Theta(0) = U(0)V^T(0)$, we choose an arbitrary $Q \in \mathbb{R}^{h\times m}$ with $Q^TQ = I_m$, then a balanced initialization is given by

$$U_{\text{balanced}}(0) = \Theta(0)\left[\Theta^T(0)\Phi_1\Phi_1^T\Theta(0)\right]^{-1/4} Q^T, \quad V_{\text{balanced}}(0) = \left[\Theta^T(0)\Phi_1\Phi_1^T\Theta(0)\right]^{1/4} Q\,.$$

Such initialization ensures the imbalanced is the zero matrix while keeping the end-to-end matrix as $\Theta(0)$. We note here the choice of $Q$ does not affect the error trajectory $E(t)$, hence the loss $\mathcal{L}(t)$.

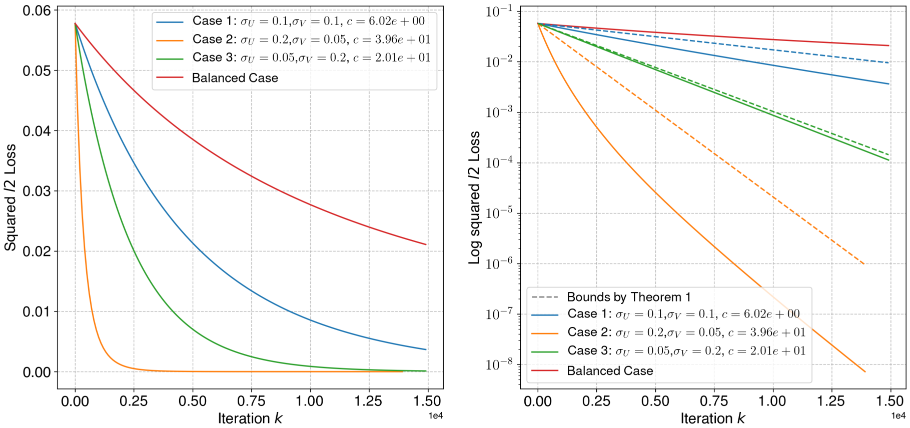

*Figure 3.* Convergence of gradient descent with different initial level of imbalance, $c := \lambda_r(\Lambda(0)) + \lambda_m(-\Lambda(0))$.

From Fig.3, we see that given fixed step size, the convergence rate is improved as we increase the level of the imbalance at initialization and the balanced initialization is the slowest among all cases. Notably, in case 3, our bound is almost tight. For case 2, our bound is tight in characterizing the asymptotic rate of convergence. Our analysis does not provide convergence guarantee for the balanced case, while Arora et al. (2018a;b) have shown linear convergence for certain cases with zero imbalance at initialization. This suggests the need of a unified convergence analysis, that is applicable for both balanced and imbalanced initialization, to obtain tighter convergence guarantees. This is subject of current research.

## A.2. Implicit regularization on wide single-hidden-layer linear network

**Generating training data** The synthetic training data is generated as following:

[2] To compute the bound from Theorem 1, the step size is scaled by $n/2$ to account for that the gradient descent uses rescaled loss function.

1) For data matrix $X$, first we generate $X \in \mathbb{R}^{n \times D}$ with all the entries sampled from $\mathcal{N}(0, D^{-1})$;

2) For $Y$, we first sample $\Theta \sim \mathcal{N}(0, D^{-1} I_D)$, and $\epsilon \sim \mathcal{N}(0, 0.01^2 I_n)$, then we let $Y = X\Theta + \epsilon$.

**Initialization and Training** We initialize $U(0), V(0)$ with $[U(0)]_{ij} \sim \mathcal{N}(0, h^{-1})$, $[V(0)]_{ij} \sim \mathcal{N}(0, h^{-1})$ and run gradient descent on the averaged loss $\tilde{L} = \frac{1}{n}\|Y - XUV^T\|_F^2$ with step size $\eta = 5e-3$. The training stops when the loss is below $1e-8$. We run the algorithm for various $h$ from 500 to 10000, and we repeat 5 runs for each $h$.

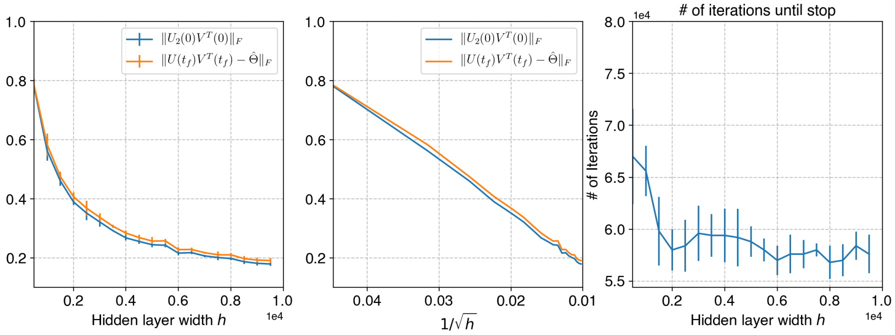

*Figure 4.* Implicit bias of wide single-hidden-layer linear network under random initialization. The line is plotting the average over 5 runs for each $h$, and the error bar shows the standard deviation. The gradient descent stops at iteration $t_f$.

Fig.4 clearly shows that the distance between the trained network and the min-norm solution, $\|U(t_f)V^T(t_f) - \hat{\Theta}\|_F$, decreases as the width $h$ increases and the middle plot verifies the asymptotic rate $\mathcal{O}(h^{-1/2})$.

## B. Comparison with the NTK Initialization for wide single-hidden-layer linear networks

In Section 3.2, we analyzed implicit bias of wide single-hidden-layer linear networks under properly scaled random initialization. Our initialization for network weights $U, V$ is different from the typical setting in previous works (Jacot et al., 2018; Du & Hu, 2019; Arora et al., 2019c). In this section, we show that under our setting, the gradient flow is related to the NTK flow by 1) reparametrization and rescaling in time ; 2) proper scaling of the network output. The use of output scaling is also used in Arora et al. (2019c).

In this paper we work with a single-hidden-layer linear network defined as $f : \mathbb{R}^D \to \mathbb{R}^m$, $f(x; V, U) = VU^T x$, which is parametrized by $U, V$. Then we analyze the gradient flow on the loss function $\mathcal{L}(V, U) = \frac{1}{2}\left\|Y - XUV^T\right\|_F^2$, given the data and output matrix $X, Y$. Lastly, in Section 4.2, we initialize $U(0), V(0)$ such that all the entries are randomly drawn from $\mathcal{N}\left(0, h^{-2\alpha}\right)$ $(1/4 < \alpha \leq 1/2)$, where $h$ is the hidden layer width.

Now we define $\tilde{U} := h^\alpha U, \tilde{V} := h^\alpha V$, then the loss function can be written as

$$\begin{aligned}
\mathcal{L}(V, U) = \tilde{\mathcal{L}}(\tilde{V}, \tilde{U}) = \frac{1}{2}\left\|Y - \frac{1}{h^{2\alpha}} X\tilde{U}\tilde{V}^T\right\|_F^2 &= \frac{1}{2}\left\|Y - \frac{\sqrt{m}}{h^{2\alpha - \frac{1}{2}}}\frac{1}{\sqrt{mh}} X\tilde{U}\tilde{V}^T\right\|_F^2 \\
&= \frac{1}{2}\sum_{i=1}^n \left\|y^{(i)} - \frac{\sqrt{m}}{h^{2\alpha - \frac{1}{2}}}\frac{1}{\sqrt{mh}}\tilde{V}\tilde{U}^T x^{(i)}\right\|_2^2 \\
&:= \sum_{i=1}^n \left\|y^{(i)} - \frac{\sqrt{m}}{h^{2\alpha - \frac{1}{2}}}\tilde{f}(x; \tilde{V}, \tilde{U})\right\|_2^2
\end{aligned}$$

Notice that $\tilde{f}(x; \tilde{V}, \tilde{U}) = \frac{1}{\sqrt{mh}}\tilde{V}\tilde{U}^T x$ is the typical network discussed in previous works (Jacot et al., 2018; Du & Hu, 2019; Arora et al., 2019c). When all the entries of $U(0), V(0)$ are initialized randomly as $\mathcal{N}\left(0, h^{-2\alpha}\right)$, the entries of $\tilde{U}(0), \tilde{V}(0)$ are random samples from $\mathcal{N}(0, 1)$, which is the typical choice of initialization for NTK analysis.

However, the difference is that $\tilde{f}(x; \tilde{V}, \tilde{U})$ is scaled by $\frac{\sqrt{m}}{h^{2\alpha - \frac{1}{2}}}$. In previous work showing non-asymptotic bound between wide neural networks and its infinite width limit (Arora et al., 2019c, Theorem 3.2), the wide neural network is scaled by a small constant $\kappa$ such that the prediction by the trained network is within $\epsilon$-distance to the one by the kernel predictor of its NTK. Moreover, Arora et al. (2019c) suggests $\frac{1}{\kappa}$ should scale as $poly(\frac{1}{\epsilon})$, i.e., to make sure the trained network is arbitrarily close to the kernel predictor, $\kappa$ should be vanishingly small. In our setting, the random initialization implicitly enforces such a vanishing scaling $\frac{\sqrt{m}}{h^{2\alpha-\frac{1}{2}}}$, as the width of network increases.

Lastly, we show that the gradient flow on $\mathcal{L}(V,U)$ only differs from the flow on $\tilde{\mathcal{L}}(\tilde{V},\tilde{U})$ by the time scale.

Suppose $U,V$[3] follows the gradient flow on $\mathcal{L}(V,U)$, we have

$$\begin{aligned}-\frac{1}{h^\alpha}\frac{\partial}{\partial U}\mathcal{L}(V,U) =& -\frac{1}{h^\alpha}X^T(Y-XUV^T)V\\ =& -\frac{1}{h^{2\alpha}}X^T\left(Y-\frac{1}{h^{2\alpha}}X\tilde{U}\tilde{V}^T\right)\tilde{V} = -\frac{\partial}{\partial\tilde{U}}\tilde{\mathcal{L}}(\tilde{V},\tilde{U})\,,\end{aligned} \tag{21}$$

and

$$\begin{aligned}-\frac{1}{h^\alpha}\frac{\partial}{\partial V}\mathcal{L}(V,U) =& -\frac{1}{h^\alpha}(Y-XUV^T)^TXU\\ =& -\frac{1}{h^{2\alpha}}\left(Y-\frac{1}{h^{2\alpha}}X\tilde{U}\tilde{V}^T\right)^TX\tilde{U} = -\frac{\partial}{\partial\tilde{V}}\tilde{\mathcal{L}}(\tilde{V},\tilde{U})\,.\end{aligned} \tag{22}$$

Consider the gradient flow on $\mathcal{L}(V,U)$ w.r.t. time $t$, from (21), we have

$$\begin{aligned}&\frac{d}{dt}U(t) = -\frac{\partial}{\partial U}\mathcal{L}(V(t),U(t))\\ \Leftrightarrow\ &\frac{1}{h^\alpha}\frac{d}{dt}\tilde{U}(t) = -\frac{\partial}{\partial U}\mathcal{L}(V(t),U(t))\\ \Leftrightarrow\ &\frac{1}{h^\alpha}\frac{d}{dt}\tilde{U}(t) = -h^\alpha\frac{\partial}{\partial\tilde{U}}\tilde{\mathcal{L}}(\tilde{V}(t),\tilde{U}(t))\\ \Leftrightarrow\ &\frac{d}{dt}\tilde{U}(t) = -h^{2\alpha}\frac{\partial}{\partial\tilde{U}}\tilde{\mathcal{L}}(\tilde{V}(t),\tilde{U}(t))\,,\end{aligned} \tag{23}$$

Similarly from (22) we have

$$\frac{d}{dt}V(t) = -\frac{\partial}{\partial V}\mathcal{L}(V(t),U(t)) \Leftrightarrow \frac{d}{dt}\tilde{V}(t) = -h^{2\alpha}\frac{\partial}{\partial\tilde{V}}\tilde{\mathcal{L}}(\tilde{V}(t),\tilde{U}(t))\,. \tag{24}$$

From (23) and (24) we know that the gradient flow on $\mathcal{L}(V,U)$ w.r.t. time $t$ essentially runs the gradient flow on $\tilde{\mathcal{L}}(\tilde{V},\tilde{U})$ with an scaled-up rate by $h^{2\alpha}$.

[3] We write $U(t),V(t)$ as $U,V$ for simplicity. Same for $\tilde{U}(t),\tilde{V}(t)$.