Using Euler's Method to Prove the Convergence of Neural Networks

Jawher Jerray, Adnane Saoud, Laurent Fribourg

Using Euler's Method to Prove the Convergence of Neural Networks

Jawher Jerray[1] Adnane Saoud[2] Laurent Fribourg[3]

***Abstract*— It was shown in the literature that, for a fully connected neural network (NN), the gradient descent algorithm converges to zero. Motivated by that work, we provide here general conditions under which we can derive the convergence of the gradient descent algorithm from the convergence of the gradient flow, in the case of NNs, in a systematic way. Our approach is based on an analysis of the error in Euler's method in the case of NNs, and relies on the concept of local strong convexity. Unlike existing approaches in the literature, our approach allows to provide convergence guarantees without making any assumptions on the number of hidden nodes of the NN or the number of training data points. A numerical example is proposed, showing the merits of our approach.**

I. Introduction

Theoretical conditions to guarantee the convergence of first order algorithms to train neural networks (NNs) have attracted attention of researchers in the recent years, where different tools have been explored to show the convergence, ranging from landscape analysis in [1], to optimal transport theory in [2] and dynamical analysis in [3], [4][1].

In a recent work, [3] has shown that for a shallow fully connected NN with ReLU activation functions, if number m of hidden nodes is sufficiently large and if the initial values of the weights of the NN are chosen according to a Gaussian distribution, then the gradient descent converges to a globally optimal solution, with a given probability. Indeed, showing the convergence of the gradient descent algorithm to a globally optimal solution boils down to showing the convergence of the error (i.e, the error between the outputs of the real data, and the outputs generated by the NN) to zero. The proof in [3] is based on the analysis of the error dynamics and is made of two steps: first, the authors in [3] start by writing the dynamics of the error in terms of the gradient flow, i.e., gradient descent with an infinitesimal step size, and have shown the convergence of the error dynamics to zero. In a second step, they rely on the proof of the convergence of the gradient flow, to generate a proof of convergence of the gradient descent algorithm to zero, while using a constant step size.

Motivated by the work of [3], in this paper, we provide sufficient conditions allowing to show how to go, in a systematic way, from the convergence of the gradient flow to the convergence of the gradient descent algorithm, while using a constant step size. Interestingly, we also provide an explicit bound on the convergence rate for the gradient descent algorithm in the case of NNs. Our approach is inspired from the works in [5] and [6, Theorem 1] by analyzing a function δ bounding the mismatch between the solutions to the gradient flow and the gradient descent algorithm.

Related work: In spirit, the closest work in the literature is [3]. The authors in [3] make the following assumptions:

(H1) $\exists \lambda_0 > 0$ such that $\lambda_{\min}(H^\infty) > \frac{\lambda_0}{2}$, where the matrix H^∞[2] is the Gram matrix induced by the activation function and the random initialization of the weights of the NN, and $\lambda_{\min}(A)$ is the minimal eigenvalue of a matrix A.

(H2) The number of hidden nodes m satisfy $m = \Omega(\frac{n^6}{(\lambda_0)^4 \varepsilon^3})$ and the constant step size $h = O(\frac{\lambda_0}{n^2})$, where n is the number of training data points and $\varepsilon > 0$ is a failure probability.

Under these conditions, the authors in [3] provide a linear convergence rate for the gradient flow method, and then use it to generate a linear convergence rate of the gradient descent algorithm for the case of NNs.

Besides [3], we are inspired here by the works of [7] and [8]. In [7] it is shown that, under some conditions, gradient descent converges to a local minimizer, almost surely with random initialization. In [8], it is shown that, when the nonlinear activation function of a one-hidden layer NN satisfies a property of "local strong convexity", then gradient descent converges at a linear rate under a resampling rule.

We give here a general framework in order to generalize these various results. Roughly speaking, we show here that, if

- The gradient descent converges to a local minimizer of the cost function $\mathcal{L}$ (see (C1) below),
- The cost function $\mathcal{L}$ is locally strongly convex around the minimizers (see (C3) below), and
- All the eigenvalues of the matrix $H[w(t)]$, describing the continuous dynamics of the gradient flow, are greater than some positive value λ^* (see (C2) below),[3]

then the gradient descent algorithm makes the training error converge to 0 at a linear rate (see Theorem 2).

The advantages of our approach with respect to the one of [3] are as follows:

[1]Jawher Jerray is with the university Sorbonne Paris Nord, LIPN, CNRS, UMR 7030, F-93430, Villetaneuse, France jerray@lipn.univ-paris13.fr

[2] Adnane Saoud is with Laboratoire des Signaux et Systèmes, CentraleSupélec, Université Paris Saclay, Gif-sur-Yvette, France. adnane.saoud@centralesupelec.fr

[3]Laurent Fribourg is with the university Paris-Saclay, CNRS, ENS Paris-Saclay, LMF, F-91190 Gif-Sur-Yvette, France fribourg@lsv.fr

This work was supported by ANR PIA funding: ANR-20-IDEES-0002.

[1]The current paper uses a dynamical analysis perspective to provide convergence guarantees.

[2]See [3, Assumption 3.1] for more details about the matrix H^∞.

[3](C2) is proven in [3] as a consequence of Assumptions (H1)-(H2).

- Our approach is general and does not depend on the algorithm giving the discrete evolution (and continuous evolution) of the weights of the NN, nor on the used activation functions.
- Our approach makes it possible to provide convergence of the gradient descent algorithm without making any assumptions on the number of data points n and number of hidden nodes m.
- While the approach in [3] uses a constant step size depending on the training data n. In this paper, we use a constant step size that do not depend on the number of training data.

Plan of the paper

Section II recalls the explicit Euler discretization approach. Section III recalls the gradient descent approach for NNs. Section IV gives the main result of the paper, by showing the convergence of the gradient descent algorithm, for the case of NNs. Finally, Section V provides a numerical example illustrating the theoretical results of the paper.

A. Notation

We denote by $\mathbb{R}$ and $\mathbb{N}$ the set of real and natural numbers, respectively. These symbols are annotated with subscripts to restrict them in the usual way, e.g., $\mathbb{R}_{>0}$ denotes the positive real numbers. We denote by $\mathbb{R}^n$ an n-dimensional Euclidean space and by $\mathbb{R}^{n\times m}$ a space of real matrices with n rows and m columns. For a matrix A, we denote by $\lambda_{\min}(A)$ its minimal eigenvalue. The Euclidean norm is denoted by $\|\cdot\|$. The set $\{1,\dots,m\}$ is denoted by $[m]$.

II. Preliminaries

Consider a differential system of the form

$$\dot{x}(t) = g(x(t)) \tag{1}$$

with initial condition $x_0 \in \mathbb{R}^n$, where $g : \mathbb{R}^n \to \mathbb{R}^n$ is a differentiable Lipschitzian function. For the sake of simplicity, we will denote by $x(t)$, the solution of (1) at time t. For $k \in \mathbb{N}$, We denote by $\tilde{x}_k$ the (explicit) Euler discretization of (1) at time $t = t_k$ with $t_k = kh$, for $k \geqslant 0$. Given an initial condition $\tilde{x}_0 \in \mathbb{R}^n$, $\tilde{x}_k$ is defined, for $k \geqslant 1$, by:

$$\tilde{x}_k = \tilde{x}_{k-1} + hg(\tilde{x}_{k-1}) \tag{2}$$

We then obtain a continuous function $\tilde{x} : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^n$ satisfying for all $k \in \mathbb{N}$, $\tilde{x}(t_k) = \tilde{x}_k$ by linear interpolation between points $\tilde{x}_k$, $k \in \mathbb{N}$.

We now define the property of "local strong convexity". In [8], it is shown that, when the nonlinear activation function of a one-hidden layer NN satisfies this property, then gradient descent converges at a linear rate under a resampling rule.

Definition 1. Consider a function $g : \mathbb{R}^n \to \mathbb{R}^n$. The function g is said to be *locally strongly convex* on a neighborhood U of a point $z \in \mathbb{R}^n$ with a factor $\gamma \in \mathbb{R}_{>0}$ if, for all $x, y \in U$:

$$(x-y)^\top (g(x) - g(y)) \geqslant \gamma \|x-y\|^2 \tag{3}$$

Remark 1. In the literature, different names are used to refer to γ in Equation (3). For example, in [9], [10], γ is called "constant of contraction" of g with respect to the Euclidean norm. In [11], $-\gamma$ is called "least upperbound Lipschitz constant" (or "one-sided Lipschitz constant") induced by the Euclidean norm.

III. Gradient Descent for Neural Networks

We now recall from [3] some definitions regarding the application of the gradient descent algorithms to NNs. We consider a NN of the form:

$$f(w,a,x) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \sigma((w^r)^\top x) \tag{4}$$

where $x \in \mathbb{R}^d$ is the input, $w^r \in \mathbb{R}^d$ is the weight vector of the first layer, $a_r \in \mathbb{R}$ is the output weight and $\sigma(\cdot)$ is the ReLU activation function: $\sigma(z) = z$ if $z \geqslant 0$ and $\sigma(z) = 0$ if $z < 0$.

We focus on the empirical risk minimization problem with a quadratic loss. Indeed, given a training data set $\{(x_i, y_i)\}_{i=1}^n$, the objective is to minimize the loss function $\mathcal{L}(w) = \sum_{i=1}^n \frac{1}{2}(f(w,a,x_i) - y_i)^2$. To do this, the authors in [3] fix the second layer and apply the gradient descent on the first layer weights matrix:

$$\tilde{w}_{k+1} = \tilde{w}_k - h\frac{\partial \mathcal{L}(\tilde{w}_k)}{\partial w_k}. \tag{5}$$

where the gradient formula for each weight vector is

$$\frac{\partial \mathcal{L}(w)}{\partial w^r} = \frac{1}{\sqrt{m}} \sum_{i=1}^{n} (f(w,a,x_i) - y_i) a_r x_i \mathbb{I}\{(w^r)^\top x_i \geqslant 0\}. \tag{6}$$

with $r \in [m]$ and $\mathbb{I}$ is the indicator on whether the event $(w^r)^\top e^i \geqslant 0$ happens. Note that the ReLU activation function is not continuously differentiable and one can view $\frac{\partial \mathcal{L}(w)}{\partial w^r}$ as a convenient notation for the right hand side of (6).

The discrete equation (5) corresponds to the Euler discterization, with a step size h, of the set of ordinary differential equations defined for $r \in [m]$ by:

$$\frac{dw^r(t)}{dt} = -\frac{\partial \mathcal{L}(w(t))}{\partial w^r(t)} \tag{7}$$

IV. Asymptotic Error in Euler's Method

As mentioned in the introduction, showing the convergence of the gradient descent algorithm, for the case of NNs, to a globally optimal solution boils down to showing the convergence of the error (i.e. the error between the outputs of the real data, and the outputs generated by the NN) to zero. For this reason, in the rest of paper we will focus on the analysis the dynamics of the error $v : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^n$, defined by $v = f(w,a,x) - y$.

As shown in [3, Section 3], the continuous dynamics of the error v can be written in a compact way

$$\frac{d}{dt}v(t) = -H[w(t)]v(t), \quad v(0) = v_0 \tag{8}$$

where $H[w] : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^{n\times n}$ is a symmetric time-varying matrix.[4]

The discrete version resulting from the Euler discretization of (8), and which corresponds to the gradient descent algorithm of $\tilde{v}$, writes:

$$\tilde{v}_{k+1} - \tilde{v}_k = -hH[\tilde{w}_k]\tilde{v}_k, \tag{9}$$

is generated using the step size h, with $\tilde{v}(0) = \tilde{v}_0$.

We are now ready to formalize the main problem of the paper:

Problem 1. *Given the discrete time system in (9), provide conditions on the matrix $H[w] : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^{n\times n}$, the loss function $\mathcal{L} : \mathbb{R}^d \to \mathbb{R}_{\geqslant 0}$ and the step size h to ensure the convergence of $\tilde{v}_k$, to zero, together with an explicit bound on its convergence rate.*

In the rest of the paper, we provide a solution to Problem 1 under the following assumptions:

(C1) The gradient descent algorithm for the update of the weights of the NN in equation (5) converges to a local minima w^*, i.e. $\frac{\partial \mathcal{L}(\tilde{w}_k)}{\partial w}$ converges to 0 as k goes to the infinity.

(C2) There exist $\lambda^* > 0$ and $t_0 \geqslant 0$ such that, for all $t \geqslant t_0$: $\lambda_{\min}(H[w(t)]) \geqslant \lambda^*$, where the time-varying matrix $H[w(t)]$ is given in (8). (cf [3, Lemma 3.2 and Lemma 3.3])

(C3) $\mathcal{L}$ is locally strongly convex around every local minimizer w^* of $\mathcal{L}$, that is: for every local minimizer w^*, there is a neighborhood around w^* on which $\mathcal{L}$ is strongly convex.

Remark 2. Let us mention that the local strong convexity property is equivalent to the fact that $-\frac{\partial \mathcal{L}(\tilde{w}_k)}{\partial w}$ belongs to a contractive region, i.e, there exists $k_0 \in \mathbb{N}$, a convex region D and a positive real $\gamma > 0$ such that $\tilde{w}_k \in D$ for all $k \geqslant k_0$, and

$$\langle \frac{\partial \mathcal{L}(w)}{\partial w} - \frac{\partial \mathcal{L}(w')}{\partial w}, w - w' \rangle \geqslant \gamma \|w - w'\|^2$$

for all $w \in D$, $w' \in D$. This is also equivalent to the positive definiteness of the Hessian of $\mathcal{L}$ in the neighborhood of each local minimizer w^*. Moreover, in view of [10], the strong convexity corresponds to the special case of contracting gradient descent in the identity metric.

Before providing our proof of the error dynamics $\tilde{v}_k$ to zero, let us first provide an analysis of the conservatism of the Assumptions (C1)-(C2)-(C3) and compare them to the Assumptions (H1) and (H2), used in [3].

- Assumption (C1) is satisfied when the gradient descent algorithm reaches a local minima, from [7] this condition is satisfied almost surely with a random initialisation of the algorithm, when the step size of the Euler discretization h is chosen such that $h < \frac{1}{L}$ where L is the Lipschitz constant of the loss function $\mathcal{L}$;
- Assumption (C3) is satisfied when the loss function $\mathcal{L}$ is strongly convex on a the neighborhood of each minimizer. This condition can be always satisfied by intializing the parameters so that they fall into the basin of the local strong convexity region, and which can be done for example by following the tensor-based approach proposed in [8].
- Assumption (C2) is a direct consequence of Assumptions (H1) and (H2) used in [3] (See Proposition 1 below).

Proposition 1. *Assumption (C2) follows from (H1)-(H2) when the map f is defined by (4).*

The proof of the proposition follows immediately from Lemmas 3.2 and 3.3 in [3].

From the discussion above and the result of Proposition 1 it follows that Assumptions (H1) and (H2) used in [3] are in general more conservative than the Assumptions (C1), (C2) and (C3) used in this paper.

In order to analyse the convergence properties of the NN, we use the result of [5] that allows to bound the sequence $\delta_k = \|\tilde{w}_k - w_k\|$, $k \in \mathbb{N}$, where $w_k = w(kh)$, h is a constant step size and $w : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^m$ is the solution to (7). Intuitively, the sequence δ_k characterizes the mismatch between the solutions to the gradient flow in (7) and the gradient descent algorithm in (5) when using the Euler method.

Theorem 1. *[5] Under assumptions (C1) and (C3), if the the step size h satisfies $h < \frac{2}{L}$, where L is the Lipschitz constant of the loss function $\mathcal{L}$, then the sequence $\|\tilde{w}_k - w_k\|$, $k \in \mathbb{N}$ converges to 0, where $\tilde{w}_k$ is defined by (5) and $w_k = (w^1(kh), \dots, w^m(kh))$ with $w^r(t)$ defined for $r \in [m]$ by (7).*

We also have the following auxilliary result

Lemma 1. *Under assumptions (C1), (C2) and (C3), if the the step size h satisfies $h < \frac{2}{L}$, where L is the Lipschitz constant of the loss function $\mathcal{L}$, then there exist $\lambda^* > 0$ and $k_0 \in \mathbb{N}$ such that, for all $k \geqslant k_0$*

$$\lambda_{\min}(H[\tilde{w}(kh)]) \geqslant \frac{\lambda^*}{2}.$$

where $\tilde{w}_k$ is defined by (5).

Proof. From Assumption (C2), we have the existence of $t_0 \geqslant 0$ such that fot all $t \geqslant t_0$, $\lambda_{\min}(H[w(t)]) \geqslant \lambda^*$. Moreover, we have from Theorem 1 that the sequence $\|\tilde{w}_k - w_k\|$ converges to zero. Hence, by continuity of the map $t \mapsto \lambda_{\min}(H[\tilde{w}(t)])$[5], we conclude the existence of $k_0 \in \mathbb{N}$ that

$$\lambda_{\min}(H[\tilde{w}(kh)]) \geqslant \frac{\lambda^*}{2}$$

for all $k \geqslant k_0$. □

[4]For f given by (4), $H[w]$ is the (symmetric positive definite matrix) defined for $w = (w^1, ..., w^m) \in \mathbb{R}^{d\times m}$ as the $n \times n$ matrix with (i, j)-th entry $H_{ij}[w] = \frac{1}{m}(e^i)^\top e^j \Sigma_{r=1}^m \mathbb{I}\{(e^i)^\top w^r \geqslant 0, (e^j)^\top w^r \geqslant 0\}$.

[5]The function $t \mapsto \tilde{w}(t)$ is continuous from the Euler's construction, which implies the continuity of the map $t \mapsto \lambda_{\min}(H[\tilde{w}(t)])$.

We are now ready to provide the main result of the paper, showing the convergence of the gradient descent algorithm for NNs.

Theorem 2. *Under assumptions (C1), (C2) and (C3), if the the step size h satisfies $h < \frac{2}{L}$, where L is the Lipschitz constant of the loss function $\mathcal{L}$, then there exist $\lambda^* > 0$ and $k_0 \in \mathbb{N}$ such that, for all $k \geqslant k_0$*

$$\|\tilde{v}_k\| \leqslant \|\tilde{v}_{k_0}\|(1 - \frac{1}{2}\lambda^* h)^{k-k_0} \tag{10}$$

where $\tilde{v}_k$ is defined by (9).

Proof. We have by (9):

$$\tilde{v}_{k+1} = \tilde{v}_k - hH[\tilde{w}_k]\tilde{v}_k.$$

which implies that

$$\|\tilde{v}_{k+1}\| = \|(I - hH[\tilde{w}_k])\tilde{v}_k\| \leqslant \|I - hH[\tilde{w}_k]\|\|\tilde{v}_k\|.$$

Using the fact that $I - hH[\tilde{w}_k]$ is a symmetric matrix, we have:

$$\|\tilde{v}_{k+1}\| < (1 - h\lambda_{\min}(H[\tilde{w}(t_k)]))\|\tilde{v}_k\|.$$

Therefore, using Lemma 1, we have the existence of $k_0 \in \mathbb{N}$ such that for all $k \geqslant k_0$:

$$\|\tilde{v}_{k+1}\| < (1 - \frac{1}{2}\lambda^* h)\|\tilde{v}_k\|.$$

Hence, for all $k \geqslant k_0$:

$$\|\tilde{v}_k\| \leqslant \|\tilde{v}_{k_0}\|(1 - \frac{1}{2}\lambda^* h)^{k-k_0}.$$

□

Remark 3. The proposed approach in the paper is not restricted to the form of f given in (4). Indeed, other forms of f, and therefore other types of NNs with potentially different activation functions, can be used under Assumptions (C1)-(C2)-(C3).

Remark 4. Let us mention that although we focused on deterministic guarantees in the paper, the result of Theorem 2 can be generalized using the same approach to the probabilistic case. That is if Assumptions (C1)-(C2)-(C3) hold with a probability p on random initialization, then Equation (10) also holds with probability p, and which corresponds to the result obtained in [3].

V. Numerical example

Consider the numerical example described in [3, Section 5], where the objective is to train a NN to fit a given data set. The training has been conducted using $n = 10$ data points and $m = 200$ hidden nodes. We initialize $w^r \sim N(0, I)$ and $a_r \sim unif[\{-1, 1\}]$ for $r \in [m]$ (initial first layer weight vector w^r drawn from a normal distribution and initial output weight a_r drawn uniformly on $\{-1, 1\}$).

We focus here on a typical simulation for a given initialization (but all the simulations we made are similar). Using the software ORBITADOR (see [12]), we compute the Lipschitz constant $L = 9.7$ of $\mathcal{L}$, and take $h = 0.15$ as a step size (thus ensuring $h < \frac{2}{L}$).[6] We then compute γ_k, which is the constant appearing in the right-hand side of Equation (3) on a neighborhood U_k of $\tilde{w}_k$ (see Figure 1). We find $\gamma_k > 0.18$ for all $k \geqslant 0$. It follows that $\mathcal{L}$ is strongly convex of factor $\gamma \geqslant 0.18$, which is in accordance with Assumption (C3).

We also compute δ_k with initial value $\delta_0 = 1$ (see Figure 2), $\lambda_{\min}(H[\tilde{w}_k])$ (see Figure 3) and $\tilde{v}_k$ with initial value $\tilde{v}_0 = 4$ (see Figure 4). We check that δ_k converges to 0, which is in accordance with Theorem 1. We also check that, for all $k \geqslant 0$, $\lambda_{\min}(H[\tilde{w}_k]) \geqslant \frac{\lambda^*}{2}$ with $\lambda^* = 0.26$, which is in accordance with Lemma 1. Finally, we check on Figure 4 that $\|\tilde{v}_k\|$ converges to 0 at a linear rate μ, which is consistent with the result of Theorem 2. The rate $\mu \approx 0.23$ is again in accordance with the theoretical lower bound given by Theorem 2, which is $\frac{\lambda^*}{2} \approx 0.13$. [7]

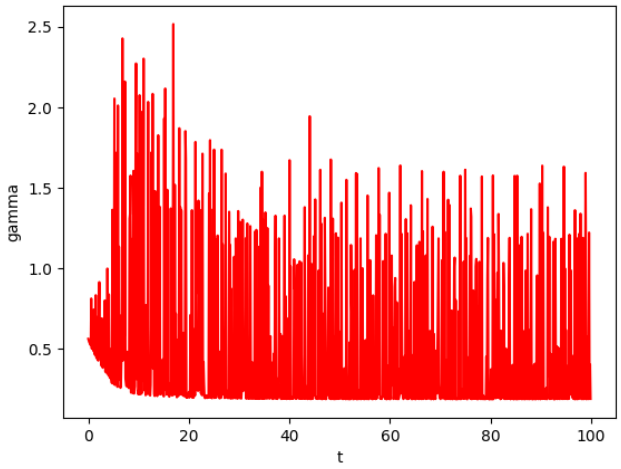

Fig. 1. Evolution of γ_k with time ($t = t_k$)

VI. Conclusion

In this paper, we proposed an approach to show the convergence of the gradient descent algorithm for NNs. Our approach is based on an analysis on the error in Euler's method. The proposed approach makes it possible to provide deterministic convergence guarantees, without making any assumptions on the number of hidden nodes of the NN. Moreover, our approach is generic, since it does not take into account the algorithmic procedure used to update the parameters of the NN, which makes our method potentially applicable to other optimization problems that go beyond the training of NNs. Moreover, we also believe that proposed approach can be used to explore the convergence properties of the stochastic gradient descent algorithm by following for example the approach proposed in [13].

[6] Note that the constraint $h < \frac{2}{L}$ is independent of n while [3] requires $h = O(\frac{\lambda_0}{n^2})$ with $\lambda_0 \approx 0.3$ and $n = 10$ here.

[7] Note that our proof of convergence does not involve the number m of hidden nodes, but assumes that (C2) holds. In contrast, the proof of [3] assumes a number $m = \Omega(\frac{n^6}{\lambda_0^4 \varepsilon^3})$ (which is huge here ($\approx \Omega(10^{13})$ for a probability of failure $\varepsilon = 0.1$), but guarantees the satisfaction of (C2).

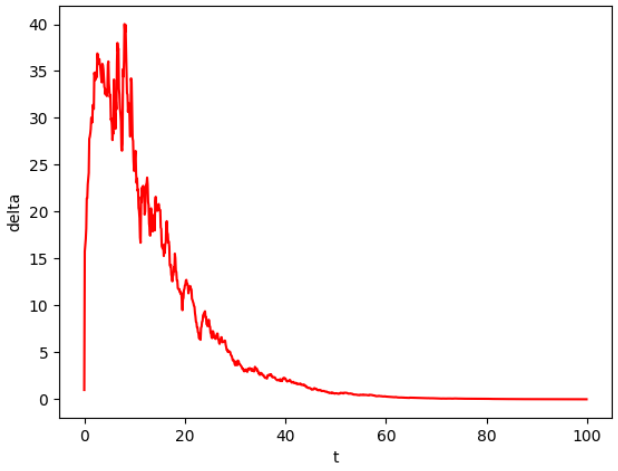

Fig. 2. Evolution of δ_k (upper bound of $\|\tilde{w}_k - w_k\|$) with time $t = t_k$

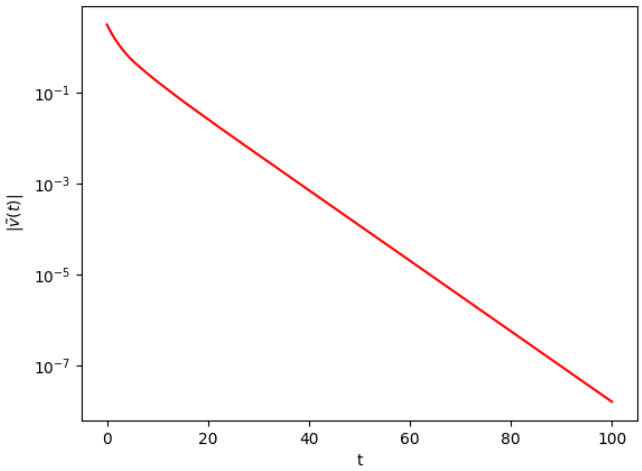

Fig. 4. Log-scale evolution of $\|\tilde{v}_k\|$ with time ($t = t_k$)

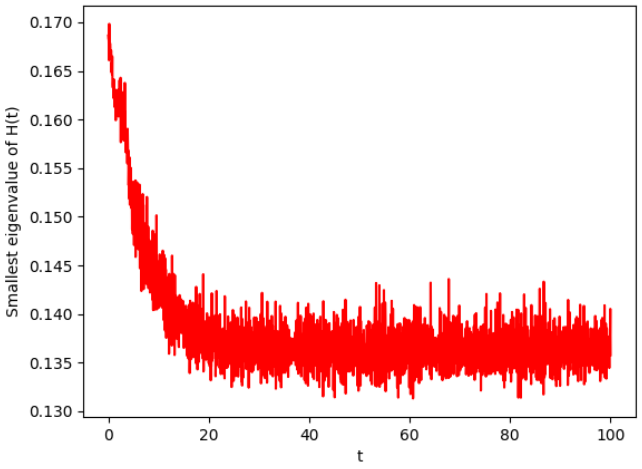

Fig. 3. Evolution of $\lambda_{\min}(H[\tilde{w}(t)])$ with time ($t = t_k$)

REFERENCES

[1] I. Safran and O. Shamir, "On the quality of the initial basin in over-specified neural networks," in *International Conference on Machine Learning*. PMLR, 2016, pp. 774–782.

[2] L. Chizat and F. Bach, "On the global convergence of gradient descent for over-parameterized models using optimal transport," *Advances in neural information processing systems*, vol. 31, 2018.

[3] S. S. Du, X. Zhai, B. Póczos, and A. Singh, "Gradient descent provably optimizes over-parameterized neural networks," *CoRR*, vol. abs/1810.02054, 2018.

[4] Y. Li and Y. Liang, "Learning overparameterized neural networks via stochastic gradient descent on structured data," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[5] J. Jerray, A. Saoud, and L. Fribourg, "Asymptotic error in Euler's method with a constant step size," in *European Control Conference, ECC 2022, Oxford, UK*.

[6] A. Le Coënt, J. Alexandre Dit Sandretto, A. Chapoutot, L. Fribourg, F. De Vuyst, and L. Chamoin, "Distributed control synthesis using Euler's method," in *RP*, ser. LNCS, M. Hague and I. Potapov, Eds., vol. 247. Springer, 2017, pp. 118–131.

[7] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht, "Gradient descent only converges to minimizers," in *29th Annual Conference on Learning Theory*, ser. Proceedings of Machine Learning Research, V. Feldman, A. Rakhlin, and O. Shamir, Eds., vol. 49. Columbia University, New York, New York, USA: PMLR, 23–26 Jun 2016, pp. 1246–1257. [Online]. Available: https://proceedings.mlr.press/v49/lee16.html

[8] K. Zhong, Z. Song, P. Jain, P. L. Bartlett, and I. S. Dhillon, "Recovery guarantees for one-hidden-layer neural networks," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 2017, pp. 4140–4149. [Online]. Available: http://proceedings.mlr.press/v70/zhong17a.html

[9] W. Lohmiller and J.-J. E. Slotine, "On contraction analysis for nonlinear systems," *Automatica*, vol. 34, no. 6, pp. 683–696, 1998.

[10] P. Wensing and J.-J. Slotine, "Beyond convexity-Contraction and global convergence of gradient descent," *PLoS One*, vol. 15, no. 8, 2020.

[11] G. Söderlind, "The logarithmic norm. History and modern theory," *BIT Numerical Mathematics*, vol. 46, no. 3, pp. 631–652, Sep 2006. [Online]. Available: https://doi.org/10.1007/s10543-006-0069-9

[12] J. Jerray, L. Fribourg, and É. André, "Robust optimal periodic control using guaranteed Euler's method," in *American Control Conference, ACC, New Orleans, LA, USA, May 25-28, 2021*. IEEE, 2021, pp. 986–991.

[13] J. Latz, "Analysis of stochastic gradient descent in continuous time," *Statistics and Computing*, vol. 31, no. 4, pp. 1–25, 2021.