2024 IEEE 63rd Conference on Decision and Control (CDC)
December 16-19, 2024. MiCo, Milan, Italy

Remarks on the Gradient Training of Linear Neural Network Based Feedback for the LQR Problem

Arthur Castello B. de Oliveira[1], Milad Siami[1], and Eduardo D. Sontag[1,2]

***Abstract*— Motivated by the growing use of Artificial intelligence (AI) tools in control design, this paper takes steps toward bridging results from Direct Gradient methods for the Linear Quadratic Regulator (LQR), and neural networks. More specifically, it looks into the case where one wants to find a Linear Feed-Forward Neural Network (LFFNN) feedback that minimizes the LQR cost. This work develops gradient formulas that can be used to implement the training of such networks and derives an important conservation law of the system. This conservation law is then leveraged to prove the global convergence of solutions and invariance of the set of stabilizing networks under the training dynamics. These theoretical results are followed by an extensive analysis of the simplest version of the problem (the "scalar case") and by numerical evidence of faster convergence of the training of general LFFNNs when compared to traditional direct gradient methods.**

I. Introduction

Artificial Intelligence (AI) and Machine Learning (ML) tools are being increasingly used in control design [1]–[7], particularly in data-driven applications, where a plant model may not be available [8], [9]. In such scenarios, an "oracle" or "digital twin" might be queried to estimate the cost associated with a specific control law implemented by a candidate feedforward network, as illustrated in Fig. 1. This network has adjustable parameters (or "weights"), which are updated through the gradient of the estimated cost, typically employing gradient descent or some other numerical optimization method.

Understanding the convergence of such learning techniques is challenging because both plant and controller are typically nonlinear. To start studying this problem, this paper proposes the analysis of Linear Feed-Forward Neural Networks (LFFNN) – that is feed-forward networks with linear activation functions – to solve Linear Quadratic Regulator (LQR) problems. At first glance, this might seem like an uninteresting problem. However, to obtain LQR solutions via Riccati equations one needs to know the system matrices, which are not available in data-driven control problem formulations, such as the one depicted in Fig. 1. In contrast, gradient methods can be used for unknown or uncertain plants, as long as estimates of the gradient can be obtained.

The analysis of direct gradient methods for LQR is an area of active research, made difficult because of a non-convex optimization landscape, and traces its origins to pioneering work by Levine and Athans starting in the late 1960s [10]. Recent publications have established global convergence properties [8], [11] as well as input-to-state stability (ISS) properties [9], [12] when gradients are subject to possible errors (seen as input disturbances).

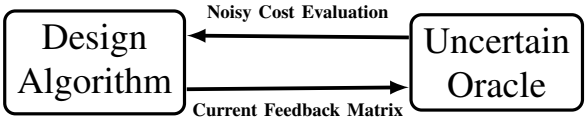

Fig. 1. System overview of the control design for an unknown plant with an uncertain oracle. The control designer attempts to find a feedback matrix that minimizes the output of the oracle, which in turn provides a (possibly noisy) estimate of the cost function every time it receives a candidate feedback matrix.

This paper deals with LFFNNs, in which the presence of hidden layers implies an overparameterization of the parameters' expression, which changes the gradient dynamics. Mathematically, feedback implemented by a feedforward neural network (FFNN) is written in an overparametrized form, as a composition $K = K_N \circ \theta \circ K_{N-1} \circ \dots \theta \circ K_1$, where the linear operator K_i represents the weights in the ith layer of the network and θ is a diagonal operator (activation function acting on each coordinate). Recent literature indicates that even the simpler problem of linear activations ($\theta =$ identity, so $K = K_N K_{N-1} \dots K_1$ resulting in linear FFNN's, or LFFNN) can have properties that make this formulation interesting, besides being a useful intermediary step for understanding deep learning [13]–[24]. The results known for direct gradient methods do not immediately generalize to LFFNNs, prompting the need to study the convergence of gradient training done on the full set of parameters. Also closely related to this paper, in [24] the authors studied how regularization can affect the critical points of an optimization solved through LFFNNs.

For the context of solving a static supervised learning problem, not only have powerful "almost everywhere" convergence results been obtained [13]–[19], but an associated ISS problem was studied as well [20], and, perhaps surprisingly, the optimization on the individual matrices K_i can result is much faster convergence than optimization on a single matrix K [21]–[23].

This paper investigates properties that can be derived for the gradient training of LFFNNs for solving the LQR problem. An advantage of this approach is that the optimal solution to the LQR problem is known to be a linear transformation, and as such no generality is lost by assuming

This work was partially supported by ONR Grant N00014-21-1-2431, NSF Grant 2121121, and AFOSR Grant FA9550-21-1-0289.

[1]Department of Electrical & Computer Engineering, Northeastern University, Boston, MA 02115 USA (e-mails: {castello.a, m.siami, e.sontag}@northeastern.edu).

[2]Department of Bioengineering, Northeastern University, Boston, MA 02115 USA.

linear activation functions. Explicit formulas for the necessary gradients are provided and used to derive an important conservation law of the dynamics. This conservation law is leveraged to prove global convergence of all solutions to critical points of the gradient dynamics, and invariance of the set of stabilizing LFFNNs along solutions. The simplest possible example, named the "scalar case," is then explored to illustrate the intuition behind overparameterization. Finally, simulations show that the speed of convergence of the gradient training can outperform more traditional direct gradient/gradient flow methods used for LQR, depending on the value of the initialization of the parameters. These results indicate both the theoretical value of studying such a problem, and the practical value of LFFNNs as design tools for data-driven control applications. Almost all proofs are omitted in this paper due to space limitations but can be viewed in the extended version [25].

II. Theoretical Background

In this paper, let $\mathbb{R}$ and $\mathbb{R}_+$ be the set of real numbers and nonnegative real numbers, respectively, and $\mathbb{I}$ be the set of imaginary numbers. The absolute value and norms are given by $|\cdot|$ and $\|\cdot\|_s$ respectively, with $s = \infty$ being the infinity norm, $s = 2$ being the two norm (vector or matrix) and $s = F$ being the Frobenius norm. Unless explicitly said otherwise, capital letters represent matrices and lower-case ones represent vectors. The identity matrix is represented by I, with the dimension clear from the context.

A. Direct Gradient Formulations for LQR

Consider a linear system $\dot{x} = Ax + Bu$ with $y = Cx$, where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{o\times n}$ are the system matrices. The objective is to determine an output feedback $u = Ky$ that minimizes

$$J(K) = \mathbb{E}_{x_0\in\mathcal{X}_0}\left[\int_0^\infty x(t)^\top Qx(t) + u(t)^\top Ru(t)\,dt\right],$$

with given positive definite cost matrices $R \in \mathbb{R}^{m\times m}$ and $Q \in \mathbb{R}^{n\times n}$, and for all solutions initialized in a set $\mathcal{X}_0 \subseteq \mathbb{R}^n$.

In [26], the authors provide, through Theorem 3.2, the gradient of the cost with respect to the feedback matrix K as $\nabla_K J(K) = -2(B^\top P_K + RKC)L_K C^\top$, where, for any K such that $A + BKC$ is Hurwitz, P_K and L_K are the unique positive definite solutions of $P_K(A + BKC) + (A + BKC)^\top P_K + C^\top K^\top RKC + Q = 0$ and $L_K(A + BKC)^\top + (A + BKC)L_K + \Sigma_0 = 0$, respectively. The matrix $\Sigma_0 = \mathbb{E}_{x_0\in\mathcal{X}_0}[x_0 x_0^\top]$ depends on the distribution of initial conditions $\mathcal{X}_0$, being equal to the identity if the system to be stabilized is initialized with any state in the unit sphere, or being equal to the covariance matrix if it is initialized from a zero-mean Gaussian distribution.

Despite its reliance on the knowledge of the system matrices, the gradient $\nabla_K J(K)$ holds great value for analysis, as demonstrated in [8], [11], where it forms the basis for theoretical guarantees regarding convergence rate and accuracy in model-free scenarios. To expand on this literature, this work explores the substitution of state feedback with a linear neural network. The goal is to investigate potential advantages in solving the original problem without compromising convergence guarantees.

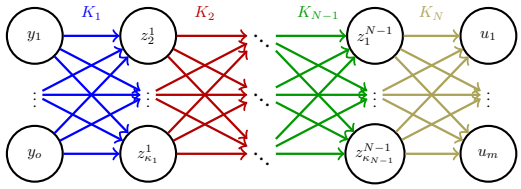

Fig. 2. Graphical representation of a Linear Feed-forward Neural Network with input layer $y \in \mathbb{R}^o$, hidden layers $z^i \in \mathbb{R}^{\kappa_i}$ and output layer $u \in \mathbb{R}^m$. The computation of the network is done for each layer as $z^i = K_i z_{i-1}$, with $z^0 = y$ and $z^N = u$, where the matrices K_i represent, in the figure, the presence and weight of edges between neurons of layer $i-1$ and layer i. The resulting input-output expression for the LFFNN then becomes $u = K_N \dots K_1 y$.

B. Neural Networks - Properties and Formulation

The optimization landscape for the gradient flow for neural networks is usually studied in terms of least-square/linear regression problems, stated as follows: let $Y = [y_1, y_2, \dots, y_k]$ and $U = [u_1, u_2, \dots, u_k]$ be the column concatenation of (possibly noisy) k input-output pairs sampled from an unknown function $\bar{\mathbf{K}}$ one wants to approximate using a neural network. That is, for every i between 1 and k, $u_i = \bar{\mathbf{K}}(y_i)$.

For some search space of neural networks $\mathcal{K}$, defined as appropriate to the problem, the optimal neural network $\mathbf{K}^*$ is the one in $\mathcal{K}$ that minimizes $J(\mathbf{K}) = \|U - \mathbf{K}(Y)\|$, where $\mathbf{K}(Y) = [\mathbf{K}(y_1), \mathbf{K}(y_2), \dots, \mathbf{K}(y_k)]$, and for some norm $\|\cdot\|$. In the specific case of a LFFNN, and being $K_i \in \mathbb{R}^{\kappa_{i-1}\times\kappa_i}$ the i-th layer parameter matrices, the function to be minimized becomes $J(K_1, \dots, K_N) = \|U - K_N \dots K_1 Y\|$, and the overall structure of a linear FFNN is illustrated in Fig. 2.

For the problem just described, and under some reasonable assumptions on the rank of Y and U, and on the dimensions of the K_is (see [13], and Assumptions 1 and 2 in [17] and references therein, or a previous work from the authors [20]), the following can be summarized from the literature about the optimization landscape of this problem [13], [15]–[17]:

Proposition 1: Consider a Linear Regression problem solved with a linear FFNN with N layers and trained through gradient flow. Assume U and Y are full column rank and that all hidden layers are wider than the number of inputs and outputs, then:

- the problem is non-convex and non-concave;
- all local minima are global minima;
- all non-minima critical points are saddles;
- in the special case $N = 2$, all non-minima critical points are strict saddles (i.e. with at least one strictly unstable direction associated with it);
- for the problem initialized at any initial condition, the solution always exists and converges to a critical point of the dynamics;
- if $N = 2$, the solutions converge to a global optimum for all initializations but a set of measure zero.

Furthermore, other works in the literature establish useful properties of overparameterized linear neural networks, when compared to equivalent non-overparameterized formulations. In [21]–[23] the authors study the speed of convergence for the gradient flow in overparameterized linear neural networks solving linear regressions, showing that depending on the initialization of the algorithm, the convergence rate can be drastically increased.

In [27] the authors study the occurrence of "benign overfitting" in a similar setup, which is the phenomenon in which a much higher number of regression parameters leads to increased accuracy for the same input-output dataset.

In [20], a previous work of the authors of this paper, we provide some insights on the loss of robustness in training overparameterized neural networks, and show how judicious restrictions on the set of initializations might circumvent this problem.

The literature is rich with properties for overparameterized neural networks that could be useful if held in the context of feedback control design. Motivated by these results, the next section looks at how one can extend these important results for Linear Quadratic cost functions, and consequently, feedback control design.

III. Feedback Control through LFFNNs

Let $\mathbf{K} = (K_1, K_2, \dots, K_N)$ be a linear FFNN with $N-1$ hidden layers, an input layer, and an output layer. Let $K_1, K_2, \dots, K_N$ be the weight matrices of each layer with $K_1 \in \mathbb{R}^{o \times \kappa_1}$, $K_2 \in \mathbb{R}^{\kappa_1 \times \kappa_2}$, and so forth, with $K_N \in \mathbb{R}^{\kappa_{N-1} \times m}$, where $\kappa_i \in \mathbb{R}^+$ is the dimension of the i-th hidden layer. For an input $y \in \mathbb{R}^o$ of the FFNN, its output $u \in \mathbb{R}^m$ is given by $u = \mathbf{K}(y) = K_N K_{N-1} \cdots K_2 K_1 y$, and its structure is as depicted in Fig. 2.

By choosing $\mathbf{K}$ as the output feedback law, the closed-loop dynamics of the LTI system becomes $\dot{x} = Ax + B\mathbf{K}(Cx) = (A + BK_N \cdots K_1 C)x$, and the LQ cost becomes

$$\begin{aligned} J(\mathbf{K}) &= \mathbb{E}_{x_0 \in \mathcal{X}_0} \left[\int_0^\infty x(t)^\top Q x(t) + u(t)^\top R u(t)\, dt \right] \\ &= \operatorname{trace}(P_{\mathbf{K}} \Sigma_0), \end{aligned} \tag{1}$$

where, for a given $\mathbf{K}$, $P_{\mathbf{K}}$ is the unique solution of

$$\begin{aligned} &P_{\mathbf{K}}(A + BK_N \dots K_1 C) + (A + BK_N \dots K_1 C)^\top P_K \\ &+ (K_N \dots K_1 C)^\top R K_N \dots K_1 C + Q = 0. \end{aligned} \tag{2}$$

With this consider:

Definition 1: Let $\mathbf{K}$ be a linear FFNN. Define $\mathcal{K} := \{\mathbf{K} \mid (A + BK_N \cdots K_1 C) \text{ is Hurwitz}\}$ and let R and Q be given positive definite matrices. Solving the LQR problem using LFFNNs consists in finding a $\mathbf{K}^* \in \mathcal{K}$ that solves

$$\begin{aligned} \min_{\mathbf{K} \in \mathcal{K}} \quad & J(\mathbf{K}) \\ \text{s.t.} \quad & (2). \end{aligned}$$

A *Gradient Flow training* for the LQR problem with LFFNNs can be obtained, for each $i = 1, 2, \dots, N$ and any fixed "learning rate" $\eta > 0$, by selecting any $\mathbf{K}_0 = \mathbf{K}|_{t=0} \in \mathcal{K}$ and imposing the following dynamics for the parameter matrices K_i that compose $\mathbf{K}$:

$$\dot{K}_i = -\eta \frac{\partial J}{\partial K_i}. \tag{3}$$

It is evident that an equilibrium of the Gradient Flow dynamics is not necessarily the global optimum of the problem, and a better understanding of the landscape of the problem is necessary before one can discuss the optimality of an obtained solution. Nonetheless, $\dot{K}_i = 0$ is a necessary condition for global optimality, which makes the equilibria of (3) natural candidates for the optimal solution. Moving forward in this paper it is assumed that $\eta = 1$, although future works might want to consider how changing the value of η affects the convergence rate of the gradient flow.

Regarding the computation of the gradients of J, consider the following theorem:

Theorem 1: Let $B_i := BK_N \dots K_{i+1}$ and $R_i := K_{i+1}^\top \dots K_N^\top R K_N \dots K_{i+1}$ for $i \in \{1, \dots, N-1\}$, $C_i := K_{i-1} \dots K_1 C$ for $i \in \{2, \dots, N\}$, $B_N := B$, $C_1 := C$, and $R_N := R$. Then

$$\nabla_{K_i} J = 2[B_i^\top P_{\mathbf{K}} + R_i K_i C_i] L_{\mathbf{K}} C_i^\top, \tag{4}$$

where $P_{\mathbf{K}}$ is the solution of (2), L_K is the solution of

$$L_{\mathbf{K}}[A + BK_N \dots K_1 C]^\top + [A + BK_N \dots K_1 C] L_{\mathbf{K}} + \Sigma_0 = 0, \tag{5}$$

and Σ_0 relates to the distribution of initial conditions, being equal to the covariance matrix if the initialization is random Gaussian with zero mean, or equal to the identity for random points at the unit sphere.

Remark 1: Notice that both Lyapunov equations (2) and (5) need only to be computed once for computing the gradient for all i, meaning that performing the feedback through a neural network incurs no large gains in complexity compared to solving the LQR problem via gradient flow.

Moving forward, it is assumed full state feedback for the system ($C = I$) and initializations in the unit sphere ($\Sigma_0 = I$). The next section looks at what can be said regarding convergence guarantees for the proposed problem.

A. A Conservation Law for LQR with LFFNNs

Notice that, relative to the weight matrix of each hidden layer, the derivative of the parameter matrices given by (4) follow an iterative structure that allows the characterization of invariant quantities of the solution in a very similar form as the ones characterized for linear regressions and LFFNNs. This property is stated in the following theorem:

Theorem 2: For a gradient flow dynamics (3) used for finding the LFFNN that minimizes the Linear Quadratic cost (1), and for any i from 1 to $N-1$, the following quantity is invariant along any trajectory:

$$K_i K_i^\top - K_{i+1}^\top K_{i+1} = (K_i K_i^\top - K_{i+1}^\top K_{i+1})_{t=0} := \mathcal{C}_i, \tag{6}$$

where $\mathcal{C}_i$ are constant matrices of appropriate dimensions.

Notice that a similar invariance property is essential to many of the properties of the gradient flow for linear regressions and linear neural networks, as can be seen from Lemma

2.3 in [17], Lemma 1 in [21], Lemma 2.1 of [14], Definition 1 of [18], and others. The fact that such property also holds for the more general Linear Quadratic cost motivates the search of other properties of the solutions also for this case.

With this, the following result regarding the global convergence of training for LFFNN in the context of LQR can be stated.

Theorem 3: Any gradient flow solution initialized in $\mathcal{K}$ exists and remains in $\mathcal{K}$ for all time, and converges to a critical point of the gradient flow dynamics.

Notice that the assumption of a stabilizing initialization for the feedback gain is standard and necessary in the literature of gradient methods for LQR.

This result not only guarantees invariance of the set of stabilizing neural networks and global convergence of solutions but also demonstrates how the invariance obtained in Theorem 2 can be used to extend results from the literature of neural networks for linear regressions to the context of neural networks for feedback design.

However, Theorem 3 does not provide any guarantees regarding convergence to the target set (solution of the original, non-overparametrized problem). More can be said regarding the optimality of the solution by leveraging some results on convergence of general nonlinear systems. Consider a general differential equation

$$\dot{x} = f(x) \tag{7}$$

evolving on an open subset $\mathbb{X} \subseteq \mathbb{R}^n$. We assume that $f : \mathbb{X} \to \mathbb{R}^n$ is continuously differentiable. The solution $x(t) = \phi(t, \xi)$ of (7) with initial state $\xi \in \mathbb{X}$ is defined (and in $\mathbb{X}$) on a maximal interval $t \in (T_\xi^{\min}, T_\xi^{\max})$, where $-\infty \leq T_\xi^{\min} < 0 < T_\xi^{\max} \leq +\infty$. The $n \times n$ Jacobian matrix of f evaluated at a point $x \in \mathbb{X}$ is denoted by $J_f(x)$.

For any subset $S \subseteq \mathbb{X}$ define the finite-time domain of attraction $\mathcal{D}_{\text{F}}(S)$ of S as the set of all $\xi \in \mathbb{X}$ such that $T_\xi^{\max} = +\infty$ and there is some $\tau_\xi \geq 0$ such that $\phi(t, \xi) \in S$ for all $t \geq \tau_\xi$.

We say that $\bar{x} \in \mathbb{X}$ is a *strict saddle equilibrium* of (7) if

1) $f(\bar{x}) = 0$ and
2) $J_f(\bar{x})$ has at least one eigenvalue with positive real part and at least one eigenvalue with non-positive real part.

The following theorem and corollary generalize results for discrete-time gradient iterations that were given in [28], which in turn generalized a result from [29] that restricted to discrete sets of strict saddles.

Theorem 4: Suppose that $\bar{x} \in \mathbb{X}$ is a strict saddle equilibrium of (7). Then there exists an open neighborhood $B \subseteq \mathbb{X}$ of $\bar{x}$ such that $\mathcal{D}_{\text{F}}(B)$ has Lebesgue measure zero.

Corollary 1: Suppose that $E \subseteq \mathbb{X}$ is a set consisting of strict saddle equilibria of (7). Then the set $\mathcal{C}_E$ of points $\xi \in \mathbb{X}$ whose trajectories converge to points in E has measure zero.

Therefore, if one were to find conditions that guarantee that the sub-optimal critical points are strict saddles, one could obtain almost everywhere convergence guarantees. We present the proofs of Theorem 4 and Corollary 1 in the appendix. In the full version of this paper [25] we follow up on a sufficient condition for which our problem satisfies the assumptions in Theorem 4.

The next section explores the simplest "scalar" version of the problem to provide some intuition on the effects of LFFNN on the training of optimal control feedback.

IV. A Simple Example

To provide some intuition behind the behavior of the parameters under training, in this section the simplest possible case is studied. Assume $N = 2$, $o = n = m = 1$, and $\kappa := \kappa_1 = 1$. The case where the parameters take these values is referred to as the "scalar case."

For the scalar case, we consider a linear system with $A \in \mathbb{R}$, $B \in \mathbb{R}$, $C \in \mathbb{R}$, $x, u, y : \mathbb{R}_+ \to \mathbb{R}$. Without loss of generality assume $B = C = 1$ and $A = a$. Furthermore, assume the scalar weights for the cost (1) are $Q = q > 0$ and $R = r > 0$ and the parameters for optimization are also written as $K_1 = k_1$ and $K_2 = k_2$. Assuming a feedback of the form $u = k_2 k_1 x$, with $a + k_2 k_1 < 0$ results in

$$\begin{aligned} J(k_1, k_2) &= \mathbb{E}_{x_0 \in \mathcal{X}_0} \left[\int_0^\infty x(t)^2 q + u(t)^2 r dt \right] \\ &= -\frac{(q + k_2^2 k_1^2 r)}{2(a + k_2 k_1)}. \end{aligned}$$

Taking the gradient with respect to k_1 and k_2 gives

$$\begin{aligned} \nabla_{k_1} J(k_1, k_2) &= -\frac{r k_2^2 k_1^2 + 2ar k_2 k_1 - q}{2(a + k_2 k_1)^2} k_2 \qquad (8) \\ &= f(k_1, k_2) k_2 \\ \nabla_{k_2} J(k_1, k_2) &= f(k_1, k_2) k_1, \qquad (9) \end{aligned}$$

which, in turn, results in $\dot{k}_1 = -f(k_1, k_2) k_2$, and $\dot{k}_2 = -f(k_1, k_2) k_1$.

Notice that, similar to the observation made in [20] for the scalar case in the linear regression, the scalar dynamics of this problem is a simple nonlinear reparameterization of linear dynamics. This means that, inside the set where $a + k_2 k_1 < 0$, the phase plane should be that of a saddle, with an inversion in the direction of the flow whenever $f(k_1, k_2) < 0$ and an extra equilibrium set given by $f(k_1, k_2) = 0$. This can be observed numerically in the plot given by Fig. 3.

The new equilibrium set given by $f(k_1, k_2) = 0$ can be studied explicitly. This condition is satisfied for any (k_1, k_2) that solves $r(k_2 k_1)^2 + 2ark_2 k_1 - q = 0$ and maintains the assumption that $a + k_1 k_2 < 0$. The only viable solution is $k_2 k_1 = -a - \sqrt{a^2 + q/r} = k_-^*$, which coincides with the optimal LQR solution for the scalar system. Furthermore, notice that $f(k_1, k_2) > 0$ for all $k_2 k_1 > k^*$ that are such that $a + k_2 k_1 < 0$, since the positive root $k_+^* = -a + \sqrt{a^2 + q/r} > 0$ is such that $a + k_+^* > 0$, and the concavity of the parabola is negative. Next in this analysis, consider the following proposition:

Proposition 2: Let $\Phi_J(t, (k_1, k_2))$ be the solution of the cost function for the scalar case initialized at (k_1, k_2). For two initializations at $(\tilde{k}_1, \tilde{k}_2)$ and $(\bar{k}_1, \bar{k}_2)$ respectively such that $\tilde{k}_1 \neq \tilde{k}_2$ and $\bar{k}_1 \neq \bar{k}_2$, that $J(\tilde{k}_1, \tilde{k}_2) = J(\bar{k}_1, \bar{k}_2)$, and that $|\tilde{c}| = |\tilde{k}_1^2 - \tilde{k}_2^2| > |\bar{k}_1^2 - \bar{k}_2^2| = |\bar{c}| > 0$, then for all time

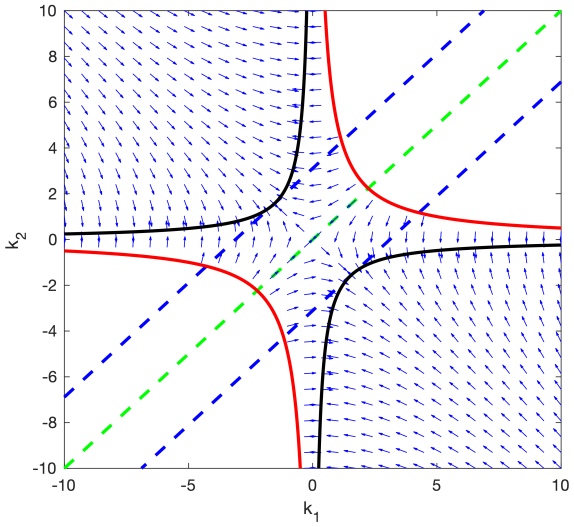

Fig. 3. Phase Plane for the dynamics of training for the scalar case described in Section IV, drawn for a stable A. The blue arrows depict the vector field at different points of the state space. The black hyperbolas are the new equilibria introduced by the condition $f(k_1, k_2) = 0$, with $f(\cdot)$ as in (8) and (9). The red hyperbolas are the borders of the set of (k_1, k_2) such that $a + k_2 k_1 < 0$, that is, such that the closed loop is stable. Note that the vector field is undefined outside this set since the cost function is undefined for closed-loop unstable systems.

$t > 0$, $\Phi_J(t, (\tilde{k}_1, \tilde{k}_2)) \leq \Phi_J(t, (\bar{k}_1, \bar{k}_2))$. In other words, the cost converges faster to the minimum value for solutions initialized with a larger imbalance $|c|$.

Still in regards to the effects of imbalance, one can notice graphically from Fig. 3 that as c increases, the associated equilibrium gets closer to the border of the set of stabilizing controllers, i.e. the red and black hyperbolas meet at infinity. That is, let $\delta\mathcal{K}$ be the border of $\mathcal{K}$ (i.e. the red hyperbolas), (k_1, k_2) be such that $k_2 k_1 = k^*$, and $k_1^2 - k_2^2 = c$, then as $c \to \infty$, $\text{dist}((k_1, k_2), \delta\mathcal{K}) \to 0$. This can be easily shown algebraically by computing the limits of each hyperbola at infinity and seeing that they indeed match.

This does not mean, however, that any disturbance during the training could take the feedback matrix to instability. In fact, from (8) and (9) note that as $(k_1, k_2) \to \delta\mathcal{K}$, $|f(k_1, k_2)| \to \infty$, with its direction being away from the border. This means that only a disturbance of infinite magnitude on the training dynamics could take a solution initialized on the set of stable closed loops into the set of unstable ones. One can even prove the following robustness result regarding the training of FFNNs for LQR for the scalar case:

Proposition 3: For the scalar case, if the solutions are initialized in $\mathcal{K}$ and are such that $|k_1 - k_2|_{t=0} > \alpha$, $\alpha \in (0, 2\sqrt{|k^*|})$ if $a < 0$ or $\alpha \in (2\sqrt{a}, 2\sqrt{|k^*|})$ if $a > 0$, and if the dynamics are disturbed as $\dot{k}_1 = -\nabla_{k_1} J + u$, and $\dot{k}_2 = -\nabla_{k_2} J + v$, where $u, v : \mathbb{R}^+ \to \mathbb{R}$, then as long as

$$\|u - v\|_\infty \leq -\alpha \frac{r\alpha^4 - 8ar\alpha^2 - 16q}{2(-\alpha^2 + 4a)^2}, \tag{10}$$

where $\|\cdot\|_\infty$ is the infinity norm of the function, and the training dynamics is ISS.

Remark 2: To see that the bound (10) is not empty, consider the polynomial $\mathcal{P}(\alpha) = -\alpha(r\alpha^4 - 8ar\alpha^2 - 16q)$ taken from the numerator of the r.h.s. of the equation. Its five roots are $\alpha_{1,2} = \pm 2\sqrt{a + \sqrt{a^2 + q/r}} \in \mathbb{R}$, $\alpha_{3,4} = \pm 2i\sqrt{-a + \sqrt{a^2 + q/r}} \in \mathbb{I}$ and $\alpha_0 = 0 \in \mathbb{R}$.

The interval of interest for the analysis is inside (sometimes equal to) the interval between $\alpha = 0$, where the line $|k_1 - k_2| = 0$ contains the point $k_1 = k_2 = 0$ which is a spurious equilibrium of the system, and $\alpha = 2\sqrt{|k_-^*|} = \alpha_1$, where the line $|k_1 - k_2| = 2\sqrt{|k_-^*|}$ contains the point $k_1 = -k_2 = \sqrt{|k_-^*|}$ which is part of the target equilibrium set of the system. In other words, $\alpha > 0$ guarantees that the set $|k_1 - k_2| > \alpha$ does not contain the spurious equilibrium at the origin, and $\alpha < 2\sqrt{|k_-^*|}$ guarantees that the set $|k_1 - k_2| > \alpha$ contains the entirety of the target set $k_2 k_1 = k_-^*$. These two lines are indicated in Fig. 3 by the green and blue lines, respectively.

To evaluate the sign of $\mathcal{P}(\alpha)$ between α_0 and α_1, evaluate its derivative at α_0: $\mathcal{P}'(\alpha_0) = -(r\alpha^4 - 8ar\alpha^2 - 16q) - \alpha(4r\alpha^3 - 16ar\alpha) = 16q > 0$. Therefore, for $\alpha \in (\alpha_0, \alpha_1) = (0, 2\sqrt{|k_-^*|})$, $\mathcal{P}(\alpha) > 0$ which means that the bound $\|u - v\|_\infty < \mathcal{B}(\alpha)$ is not empty has a maximum value inside $(0, 2\sqrt{|k_-^*|})$.

Through this simple example, one can see how interesting and rich the problem discussed in this paper can be, as well as capture some of its intuition in a simpler context. The next section investigates numerically whether the increased speed of convergence, proven for the scalar case here, might still hold for the general problem.

V. Numerical Results On the Speed of Convergence

This section investigates numerically how the use of LFFNNs for solving the LQR problem can affect the speed of convergence of training when compared to traditional gradient methods for LQR. This analysis is similar in principle to the discussion held in [21]–[23], where it was shown and proved that, in the context of linear regressions, an overparameterized formulation might converge to the optimum much quicker than a non-overparameterized one, depending on the initialization of the system.

The simulations are performed for a random stable linear system and find the optimal feedback matrix using an arbitrarily wide, single hidden-layer linear neural network structure, initialized with parameters with varying levels of "imbalance." The results are shown in Fig. 4 together with the convergence for the simple gradient LQR, without using linear neural networks.

The experiments in Fig. 4 first use the system matrices to compute the optimal LQR matrix K^*, then computes an SVD $K^* = \Psi\Sigma\Phi^\top$, generates a random $k \times k$ orthogonal matrix Γ, and defines the initialization $K_2 = \mu\Psi\Sigma^{1/2}\Gamma^\top$ and $K_1 = (1/\mu)\Gamma\Sigma^{1/2}\Phi^\top$, where $\mu \geq 1$ is the level of imbalance.

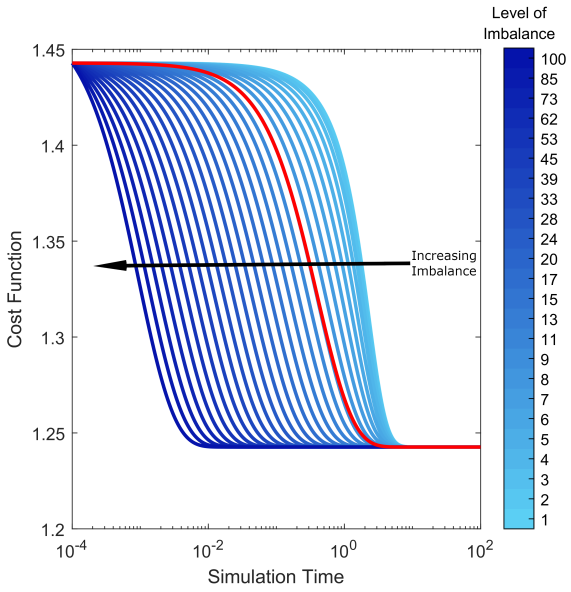

Fig. 4. Simulation results for training a LFFNN with a single wide hidden layer (blue gradient) and for a standard gradient dynamics (red) for LQR. Notice that the convergence of the LFFNN to the optimal value of the cost function $J(K)$ can be quicker or slower depending on the "imbalance" μ of the initialization. In the graph, μ is varied logarithmically from 1 (light blue) to 100 (dark blue). This experiment reproduces, for Linear Quadratic cost functions, the results presented in [23] for linear regression.

Evidently, if one had access to the value of the optimal feedback matrix K^*, the training loses its purpose, however, this simulation serves as a demonstration of regions of the state space where the convergence is much faster or slower than the non-overparameterized case. Furthermore, those regions are defined by the magnitude of $\mathcal{C}$, defined in (6), which is independent of K^*.

The simulation also presents a non-overparametrized gradient flow convergence for the same problem and initialization (red), illustrating that the imbalance can both accelerate and decelerate convergence, depending on its value.

VI. Conclusions

This paper investigates the use of Linear Feed-Forward Neural Networks (LFFNNs) for training an optimal LQR feedback. The theoretical exploration conducted yielded several important results, as summarized below.

In Theorem 1, the explicit computation of the gradient of the Linear Quadratic (LQ) cost as a function of the parameter matrix of each layer of the linear neural network is given. Building upon this, Theorem 2 establishes a crucial conservation law along the solutions of the gradient flow. Leveraging this property, Theorem 3 proves global convergence of solutions to critical points of the gradient dynamics. This guarantees the existence and boundedness of all solutions initialized in the set of LFFNNs that stabilize the closed-loop system in question, which is also shown to be invariant under gradient flow. Then the "scalar case" is studied. This simplest case is thoroughly explored to illustrate and provide the intuition behind the effects of overparameterization on the training dynamics of the parameters as well as indicate the usefulness of the problem.

Finally, this paper provides numerical evidence that other properties of linear neural networks for linear regression can be extended to the linear neural networks for the LQR problem. Specifically, it was observed that initializing training with a high-norm value to the invariants, defined in Theorem 2, potentially improves convergence rates when compared to the gradient LQR case, despite the increased number of parameters.

The simulations presented, alongside Theorem 3, provide evidence of how the invariance condition established in Theorem 2 represents an initial step in formalizing theoretical guarantees from linear neural network theory for linear regression literature to the LQR case. Future directions for research involve a deeper analysis of critical points in the gradient dynamics, and characterizing the optimization landscape. One can also be interested in investigating almost everywhere convergence to the optimum, particularly in the single hidden layer case. Additionally, it is possible to explore how the advantageous properties of faster convergence and increased accuracy inherent in overparameterized neural networks can benefit practical applications of LFFNNs to LQR problems.

Appendix

Proof of Theorem 4

Pick any equilibrium point $\bar{x} \in \mathbb{X}$. Next modify the vector field f to a vector field g so that g coincides with f on an open neighborhood of U of $\bar{x}$ and g vanishes outside a compact set $K \subseteq \mathbb{X}$. Since g has compact support, solutions of $\dot{x} = g(x)$ are defined for all $t \in \mathbb{R}$, and the map $G : x \mapsto \gamma(1, x)$ (time-1 map for g, where γ is the flow of g) is a C^1 diffeomorphism. Since $\gamma(1, \bar{x}) = \bar{x}$, it follows that $G(\bar{x}) = \bar{x}$, and since G is a diffeomorphism, there is some neighborhood V of $\bar{x}$ in which $G = F$, where F is the time-1 map for f. The Center-Stable Manifold Theorem as, for example, stated in [30], Theorem III.7, applied G restricted to V, gives the existence of an open subset B of V and a local center stable manifold W of dimension equal to the number of eigenvalues with nonpositive real part, with the property that for any $x \in B$ such that $G^\ell(x) \in V$ for all $\ell \in \mathbb{Z}_+$ necessarily $x \in W$. Since $F = G$ on V, the same property is true for F.

Pick any point $\xi \in \mathcal{D}_{\text{F}}(B)$ and pick $k = \tau_\xi \geq 0$, without loss of generality a positive integer, such that $\phi(t, \xi) \in B$ for all $t \geq k$. Let $x = \phi(k, \xi)$. Then $F^\ell(x) = \phi(k + \ell, \xi) \in B$ for all $\ell \in \mathbb{Z}_+$, and therefore necessarily $x \in W$. We have established that for each $\xi \in \mathcal{D}_{\text{F}}(B)$ there is some k such that F^k, the time-k map of the flow f, is defined at ξ and satisfies $F^k(\xi) \in W$. It follows that $\mathcal{D}_{\text{F}}(B)$ is the union of the (countably many) sets S_k consisting of those points $x \in \mathbb{X}$ such that $F^k(x) \in W$. Thus it will suffice to show that each set S_k has measure zero. Note that F^k is a local diffeomorphism, it being a time-k map for a differentiable vector field. (It is not necessarily a global diffeomorphism, so we cannot argue that $(F^k)^{-1}(W)$

is diffeomorphic to W. In fact, preimages may not even belong to $\mathbb{X}$.) Thus, there is an open neighborhood N_ξ of ξ in $\mathbb{X}$ that maps diffeomorphically by F^k into an open neighborhood M_ξ of $F^k(\xi)$. By uniqueness of solutions in time $-k$, the preimage of M_ξ is exactly N_ξ. Note that S_k is included in the union N_k over $\xi \in \mathbb{X}$ of the sets N_ξ. Also, for each ξ, $N_\xi \cap S_k$ maps diffeomorphically onto $M_\xi \cap W$, and therefore $N_\xi \cap S_k$ has measure zero (because W has measure zero and diffeomorphisms transform null sets into null sets). Recall that Lindelöf's Lemma (see e.g [31]) ensures that every open cover of any subset S of $\mathbb{R}^n$ (or more generally, of any second-countable space) admits a countable subcover. Applied to N_k, we have a countable subcover by sets N_{ξ_k}, and for each of these $N_{\xi_k} \cap S_k$ has measure zero, so $N_k \cap S_k = S_k$ has measure zero as well. ∎

Proof of Corollary 1

For each $\bar{x} \in E$, we may pick by Theorem 4 a ball $B_{\bar{x}} \subseteq \mathbb{X}$ of $\bar{x}$ such that $\mathcal{D}_{\text{F}}(B_{\bar{x}})$ has measure zero. The union of the sets $B_{\bar{x}}$ covers E. By Lindelöf's Lemma applied to $S = E$, we conclude that there is a countable subset of balls $\{B_{\bar{x}_k}, k \in \mathbb{Z}_+\}$ which covers E. We claim that $\mathcal{C}_E \subseteq \bigcup_k \mathcal{D}_{\text{F}}(B_{\bar{x}_k})$. Since a union of measure zero sets has measure zero, this will establish the claim. So pick any $\xi \in \mathcal{C}_E$. Thus, $\phi(t,\xi) \to \bar{x}$ for some $\bar{x} \in E$. Since $E \subseteq \bigcup_k B_{\bar{x}_k}$, it follows that $\bar{x} \in B_{\bar{x}_k}$ for some k. Since $B_{\bar{x}_k}$ is a neighborhood of $\bar{x}$, this means that there is some $\tau_\xi \geq 0$ such that $\phi(t,\xi) \in B_{\bar{x}_k}$ for all $t \geq \tau_\xi$. Therefore $\xi \in \mathcal{D}_{\text{F}}(B_{\bar{x}_k})$. This completes the proof. ∎

References

[1] M. K. Wafi and M. Siami, "A comparative analysis of reinforcement learning and adaptive control techniques for linear uncertain systems," in *2023 Proceedings of the Conference on Control and its Applications (CT)*. SIAM, 2023, pp. 25–32.

[2] M. Alali and M. Imani, "Reinforcement learning data-acquiring for causal inference of regulatory networks," in *American Control Conference (ACC), IEEE*, 2023.

[3] R. Zandi, H. Salehinejad, K. Behzad, E. Motamedi, and M. Siami, "Robot motion prediction by channel state information," in *2023 IEEE 33rd International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2023, pp. 1–6.

[4] E. Motamedi, K. Behzad, R. Zandi, H. Salehinejad, and M. Siami, "Robustness evaluation of machine learning models for robot arm action recognition in noisy environments," *arXiv preprint arXiv:2401.09606*, 2024.

[5] A. Ravari, S. F. Ghoreishi, and M. Imani, "Optimal recursive expert-enabled inference in regulatory networks," *IEEE Control Systems Letters*, vol. 7, pp. 1027–1032, 2022.

[6] ——, "Optimal inference of hidden Markov models through expert-acquired data," *IEEE Transactions on Artificial Intelligence*, 2024.

[7] M. Sznaier, A. Olshevsky, and E. D. Sontag, "The role of systems theory in control oriented learning," in *25th International Symposium on Mathematical Theory of Networks and Systems*, 2022.

[8] M. Fazel, R. Ge, S. M. Kakade, and M. Mesbahi, "Global Convergence of Policy Gradient Methods for the Linear Quadratic Regulator," Mar. 2019, arXiv:1801.05039 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1801.05039

[9] L. Cui, Z.-P. Jiang, and E. D. Sontag, "Small-disturbance input-to-state stability of perturbed gradient flows: Applications to LQR problem," *Systems & Control Letters*, vol. 188, p. 105804, 2024.

[10] W. Levine and M. Athans, "On the determination of the optimal constant output feedback gains for linear multivariable systems," *IEEE Transactions on Automatic Control*, vol. 15, no. 1, pp. 44–48, Feb. 1970. [Online]. Available: http://ieeexplore.ieee.org/document/1099363/

[11] H. Mohammadi, A. Zare, M. Soltanolkotabi, and M. R. Jovanovic, "Convergence and Sample Complexity of Gradient Methods for the Model-Free Linear–Quadratic Regulator Problem," *IEEE Transactions on Automatic Control*, vol. 67, no. 5, pp. 2435–2450, May 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9448427/

[12] E. D. Sontag, "Remarks on input to state stability of perturbed gradient flows, motivated by model-free feedback control learning," *Systems & Control Letters*, vol. 161, p. 105138, Mar. 2022. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0167691122000056

[13] K. Kawaguchi, "Deep Learning without Poor Local Minima," in *Advances in Neural Information Processing Systems*, vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/hash/f2fc990265c712c49d51a18a32b39f0c-Abstract.html

[14] B. Bah, H. Rauhut, U. Terstiege, and M. Westdickenberg, "Learning deep linear neural networks: Riemannian gradient flows and convergence to global minimizers," *Information and Inference: A Journal of the IMA*, vol. 11, no. 1, pp. 307–353, Mar. 2022. [Online]. Available: https://academic.oup.com/imaiai/article/11/1/307/6127129

[15] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Networks*, vol. 2, no. 1, pp. 53–58, Jan. 1989. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/0893608089900142

[16] A. Eftekhari, "Training Linear Neural Networks: Non-Local Convergence and Complexity Results," in *Proceedings of the 37th International Conference on Machine Learning*. PMLR, Nov. 2020, pp. 2836–2847, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v119/eftekhari20a.html

[17] Y. Chitour, Z. Liao, and R. Couillet, "A Geometric Approach of Gradient Descent Algorithms in Linear Neural Networks," *Mathematical Control and Related Fields*, vol. 13, no. 3, pp. 918–945, 2023, arXiv:1811.03568 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1811.03568

[18] S. Arora, N. Cohen, N. Golowich, and W. Hu, "A convergence analysis of gradient descent for deep linear neural networks," *arXiv preprint arXiv:1810.02281*, 2018.

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

[20] A. C. B. de Oliveira, M. Siami, and E. D. Sontag, "Dynamics and perturbations of overparameterized linear neural networks," in *2023 62nd IEEE Conference on Decision and Control (CDC)*. IEEE, 2023, pp. 7356–7361.

[21] H. Min, S. Tarmoun, R. Vidal, and E. Mallada, "On the Explicit Role of Initialization on the Convergence and Implicit Bias of Overparametrized Linear Networks," in *Proceedings of the 38th International Conference on Machine Learning*. PMLR, July 2021, pp. 7760–7768, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v139/min21c.html

[22] H. Min, R. Vidal, and E. Mallada, "On the convergence of gradient flow on multi-layer linear models," in *International Conference on Machine Learning*. PMLR, 2023, pp. 24 850–24 887.

[23] S. Tarmoun, G. Franca, B. D. Haeffele, and R. Vidal, "Understanding the Dynamics of Gradient Flow in Overparameterized Linear models," in *Proceedings of the 38th International Conference on Machine Learning*. PMLR, July 2021, pp. 10 153–10 161, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v139/tarmoun21a.html

[24] A. Taghvaei, J. W. Kim, and P. Mehta, "How regularization affects the critical points in linear networks," *Advances in neural information processing systems*, vol. 30, 2017.

[25] A. C. B. de Oliveira, M. Siami, and E. D. Sontag, "Convergence analysis of overparametrized lqr formulations," *arXiv preprint arXiv:2408.15456*, 2024.

[26] T. Rautert and E. W. Sachs, "Computational Design of Optimal Output Feedback Controllers," *SIAM Journal on Optimization*, vol. 7, no. 3, pp. 837–852, Aug. 1997. [Online]. Available: http://epubs.siam.org/doi/10.1137/S1052623495290441

[27] P. L. Bartlett, P. M. Long, G. Lugosi, and A. Tsigler, "Benign overfitting in linear regression," *Proceedings of the National Academy of Sciences*, vol. 117, no. 48, pp. 30 063–30 070, 2020.

[28] I. Panageas and G. Piliouras, "Gradient Descent Only Converges to Minimizers: Non-Isolated Critical Points and Invariant Regions," in *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), C. H. Papadimitriou, Ed., vol. 67. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017, pp. 2:1–2:12. [Online]. Available: https://drops-dev.dagstuhl.de/entities/document/10.4230/LIPIcs.ITCS.2017.2

[29] 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

[30] M. Shub, *Global Stability of Dynamical Systems*. Springer, 2013.

[31] J. Kelley, *General Topology*, ser. Graduate Texts in Mathematics. Springer New York, 1975. [Online]. Available: https://books.google.com/books?id=-goleb9Ov3oC