A Geometrical Analysis of Global Stability in Trained Feedback Networks

Francesca Mastrogiuseppe
f.mastrogiuseppe@ucl.ac.uk
Laboratoire de Neurosciences Cognitives et Computationelles, INSERM U960, and Laboratoire de Physique Statistique, CNRS UMR 8550, Ecole Normale Supérieure–PSL Research University, Paris 75005, France

Srdjan Ostojic
srdjan.ostojic@ens.fr
Laboratoire de Neurosciences Cognitives et Computationelles, INSERM U960, Ecole Normale Supérieure–PSL Research University, Paris 75005, France

Recruent neural networks have been extensively studied in the context of neuroscience and machine learning due to their ability to implement complex computations. While substantial progress in designing effective learning algorithms has been achieved, a full understanding of trained recurrent networks is still lacking. Specifically, the mechanisms that allow computations to emerge from the underlying recurrent dynamics are largely unknown. Here we focus on a simple yet underexplored computational setup: a feedback architecture trained to associate a stationary output to a stationary input. As a starting point, we derive an approximate analytical description of global dynamics in trained networks, which assumes uncorrelated connectivity weights in the feedback and in the random bulk. The resulting mean-field theory suggests that the task admits several classes of solutions, which imply different stability properties. Different classes are characterized in terms of the geometrical arrangement of the readout with respect to the input vectors, defined in the high-dimensional space spanned by the network population. We find that such an approximate theoretical approach can be used to understand how standard training techniques implement the input-output task in finite-size feedback networks. In particular, our simplified description captures the local and the global stability properties of the target solution, and thus predicts training performance.

1 Introduction

Trained networks have been used as a test bed for understanding how complex computations can be performed by large ensembles of elementary nonlinear units (Rosenblatt, 1958; Tsoi & Back, 1997; LeCun, Bengio, & Hinton, Neural Computation 31, 1139–1182 (2019) © 2019 Massachusetts Institute of Technology
doi:10.1162/neco_a_01187
Trained recurrent networks, moreover, can be interpreted as models of cortical circuits that one can easily probe and analyze in an in silico setup (Mante, Sussillo, Shenoy, & Newsome, 2013; Sussillo, 2014; Rajan, Harvey, & Tank, 2016; Wang, Narain, Hosseini, & Jazayeri, 2018; Barak, 2017). Because of the long temporal dependencies generated by the intricate connectivity, designing efficient learning algorithms for recurrent networks is a difficult task. Only recently have advances in technology and machine learning algorithms (Atiya & Parlos, 2000; Martens & Sutskever, 2011; Pascanu, Mikolov, & Bengio, 2013) made it possible to effectively train large recurrent networks by overcoming severe stability issues (Doya, 1992; Bengio, Simard, & Frasconi, 1994).

A first strategy to circumvent the problem of dealing with recurrent temporal dependencies was proposed by Jaeger and Haas (2004) and Maass, Joshi, and Sontag (2007). In these frameworks, external feedback loops were implemented to control and reshape the disordered dynamics that spontaneously appear in large random networks. In such feedback architectures, the task-specific component of the recurrent dynamics is totally specified by the feedback input from the readout unit to the random reservoir. As a result, learning can be restricted to the readout weights, which are trained by clamping the feedback input to the target of the task (Jaeger, 2001; Lukosevicius, 2012) or by fast weights modification within an online training scheme (Jaeger, 2002; Sussillo & Abbott, 2009).

From a mathematical perspective, feedback loops can be interpreted as rank-one perturbations of the originally random connectivity. Such a low-rank perturbation can significantly modify the characteristics of the original matrix (Tao, 2013) and generates the articulate dynamics required by complex tasks like pattern generation and decision making (Sussillo & Abbott, 2009). In order to obtain arbitrary activity profiles, the entries of the rank-one term are often tightly fine-tuned to the entries of the original random matrix. As learning potentially relies on amplifying the correlations between the readout vector and the finite-size random bulk, developing a formal understanding of the resulting network dynamics has proved to be a difficult theoretical task (Sussillo & Barak, 2013; Massar & Massar, 2013; Rivkind & Barak, 2017).

In this work, we focus on a simple yet not fully explored scenario: the feedback network is trained to produce a stationary output signal in response to a constant pattern of external inputs (Rivkind & Barak, 2017). To characterize the problem from a theoretical perspective, we focus on a restricted class of simple training solutions for which a complete analytical description can be developed. These training solutions correspond to readouts that are strongly aligned with the external and feedback input weights but uncorrelated with respect to the random synaptic weights within the reservoir. For such readouts, we derive a mean-field theory that captures the network behavior in terms of a few representative macroscopic variables (Sompolinsky, Crisanti, & Sommers, 1988; Mastrogiuseppe & Ostojic, 2018) and allows us to isolate the relevant geometrical components of the
readout vector that dominantly contribute to the final network dynamics. In particular, we find that different combinations of these components can be mapped into different output states of the final feedback network.

In a second step, we consider the more general training solutions that are found by standard learning techniques, such as least-squares (LS) (Jaeger, 2001; Lukosevicius, 2012) and recursive least-squares (RLS) algorithms (Sussillo & Abbott, 2009; Laje & Buonomano, 2013). We use our mean-field framework to derive approximate analytical descriptions of such solutions, where the full trained readout vector is replaced by its projection on the hyperplane spanned by the external and feedback input vectors. We show that such simplified portraits predict with good precision the output states that are displayed by trained networks, together with their stability: our simplified description completely neglects the correlations between the readout vector and the random bulk but still accurately predicts how the network phase space is reshaped by training. Specifically, we show that learning through the rank-one feedback can induce severe dynamical instabilities even in an extremely simple task. We use our theoretical framework to unveil and compare the strategies that the LS and RLS algorithms adopt to solve the task and to clarify the reasons of their failures and successes.

2 The Setup

We consider a rate-based network model where every unit \( i \) is characterized by a continuous activation variable \( x_i \) and its nonlinear transform \( \phi(x_i) \) (Sompolinsky et al., 1988). In line with many classical studies (Jaeger, 2001; Sussillo & Abbott, 2009; Laje & Buonomano, 2013; Rivkind & Barak, 2017), we chose \( \phi(x) = \tanh(x) \). The case of rate networks characterized by a threshold-linear activation function is discussed in appendix C.

To begin, we consider a random reservoir network driven by an external input consisting of an \( N \)-dimensional stationary pattern \( I \). Its dynamics read

\[
\dot{x}_i(t) = -x_i(t) + g \sum_{i=1}^{N} \chi_{ij} \phi(x_j(t)) + I_i, \tag{2.1}
\]

where we have rescaled time to set the time constant to unity. The parameter \( g \) quantifies the strength of the random connectivity \( \chi \), whose entries are generated from a centered gaussian distribution of variance \( 1/N \) (Sompolinsky et al., 1988). The network output is defined at the level of a linear readout,

\[
z(t) = \sum_{i=1}^{N} n_i \phi(x_i(t)), \tag{2.2}
\]

where vector \( n \) sets the output readout direction.
A task specifies how the network needs to map the external input $\mathbf{I}$ into the output signal $z(t)$. The task is solved when an appropriate readout vector $\mathbf{n}$ has been learned, such that the readout $z$ correctly matches the target. Here we consider a specific fixed-point task in which the network has to hold an arbitrary stationary output signal: $z(t) = A$.

The random reservoir in equation 2.1 admits a unique input-driven stable state (Kadmon & Sompolinsky, 2015; Mastrogiuseppe & Ostojic, 2017), where the activity of single units is a combination of the feedforward input and the recurrent dynamics generated by random synapses. For reasons that will become clear, we refer to this attractor as the feedback-free activity state. Once the network activity has relaxed into such a state, the task can be solved by tuning the readout weights $n_i$ until the network output matches the target $A$. This training protocol results into a feedback-free implementation of the task, for which the dynamics of the reservoir is independent of the choice of the readout weights.

Although feedback-free reservoirs are suitable for implementing many simple input-output associations, they have a limited range of applications. As pointed out in a series of seminal studies (Jaeger, 2001; Maass et al., 2007; Sussillo & Abbott, 2009), more flexible architectures can be obtained by using the readout signal $z(t)$ as an additional external input, which acts as an external feedback to the random reservoir (see Figure 1a). The dynamics of the network obey

$$\dot{x}_i(t) = -x_i(t) + g \sum_{j=1}^{N} (\chi_{ij} \phi(x_j(t)) + m_i z(t)) + I_i,$$

(2.3)

where the $N$-dimensional vector $\mathbf{m}$ defines the input direction of the feedback signal. In standard training procedures (Jaeger, 2001; Sussillo & Abbott, 2009), vector $\mathbf{m}$ is considered to be fixed across learning and is generated with random entries.

The feedback loop from the readout to the reservoir introduces a novel, one-dimensional element of recurrent dynamics in the network. Substituting equation 2.2 into equation 2.3 allows verifying that the final feedback architecture can be described by an equivalent connectivity matrix, which contains the original random bulk together with a rank-one matrix specified by the feedback vectors:

$$\dot{x}_i(t) = -x_i(t) + \sum_{i=1}^{N} (g \chi_{ij} + m_i n_j) \phi(x_j(t)) + I_i.$$

(2.4)

As observed in a variety of studies (Jaeger, 2001; Lukosevicius, 2012), such a low-rank matrix perturbation can deeply reshape the dynamical landscape of the reservoir network. The number of activity states and their stability
properties, which directly affect training performance in practical applications, strongly depend on the choice of the readout vector \( \mathbf{n} \). Building an exact causal relationship between the different readout solutions and the corresponding emerging dynamics is, however, a nontrivial theoretical problem (Massar & Massar, 2013; Rivkind & Barak, 2017; Landau & Sompolinsky, 2018).

In this work, we characterize the dynamics emerging in feedback networks (see equation 2.3) trained to solve the fixed-point task. The input vectors \( \mathbf{m} \) and \( \mathbf{I} \) determine the fixed geometry of the task by specifying two preferred directions in the \( N \)-dimensional phase space spanned by the reservoir population. In our framework, vectors \( \mathbf{m} \) and \( \mathbf{I} \) are generated as random patterns of variance \( \sigma_m^2 \) and \( \sigma_I^2 \). Their entries are extracted from joint gaussian distributions of null mean. The two vectors can thus be constructed as

\[
\mathbf{m} = \sigma_m \left( \rho \xi + \sqrt{1 - \rho^2} \eta_m \right), \\
\mathbf{I} = \sigma_I \left( \rho \xi + \sqrt{1 - \rho^2} \eta_I \right)
\]  

(2.5)

where \( \xi \), \( \eta_m \), and \( \eta_I \) are standard gaussian vectors. The parameter \( \rho \) quantifies the strength of correlations between the input vectors \( \mathbf{m} \) and \( \mathbf{I} \), defined onto the overlap direction \( \xi \) (alternatively, the overlap direction \( \xi \) can be taken to be parallel to the unitary vector \( \mathbf{u} = (1, \ldots, 1) \), resulting in input vectors \( \mathbf{m} \) and \( \mathbf{I} \) characterized by random and uncorrelated entries of nonzero mean). From this setup, the feedback-free implementations can be retrieved as a special case by setting the feedback inputs to zero through \( \sigma_m = 0 \).

3 General Classes of Solutions

As a first step, we approach the problem from a theoretical perspective. We analyze the space of all the possible task solutions by focusing on the limit of a large random bulk (\( N \to \infty \)).

The specific task we consider requires a constant readout signal, which is trivially obtained in stationary network states. Equilibrium solutions, which are obtained by setting \( \dot{x}_i(t) = 0 \) in equation 2.3, can be written as

\[
x_i = m_i z + I_i + \delta_i
\]  

(3.1)

where the term \( \delta_i \) corresponds to self-consistent recurrent input generated by the random part of the connectivity:

\[
\delta_i = g \sum_{j=1}^{N} \chi_{ij} \phi(x_j(t)).
\]  

(3.2)
The value of the readout $z$ can thus be reexpressed as

$$z = \sum_{i=1}^{N} n_i \phi(m_i z + I_i + \delta_i),$$

$$= N \langle n_i \phi(m_i z + I_i + \delta_i) \rangle,$$

(3.3)

where the notation $\langle \cdot \rangle$ indicates an average over the population of units.

For a broad range of $\phi(x)$ functions, the output $z$ is thus determined by the correlations between the readout $n$ and the vectors defining the direction of the network activity: $m$, $I$, and $\delta$. While vectors $m$ and $I$ are considered to be fixed, the direction of the residual input $\delta$ varies from one realization to the other of the random connectivity $\chi$. For a given matrix $\chi$, furthermore, the direction of vector $\delta$ varies in the different dynamical states.
admitted by the network. For these reasons, the residual input $\delta$ represents the hardest term to evaluate and control.

One approach consists of fixing the random connectivity matrix and computing $\delta$ in the open-loop configuration, obtained by clamping the readout signal to the target: $z(t) = A$ (Jaeger, 2001; Rivkind & Barak, 2017). Such a value for $\delta$, however, holds only locally—that is, in the vicinity of the target fixed point. As a consequence, this approach does not allow us to control the full, global dynamics that the readout $n$ imposes on the final feedback network. For example, it does not allow predicting whether additional, spurious attractors are created, together with the desired output state.

A different approach is used in classical mean-field studies of purely random networks (Sompolinsky et al., 1988; Molgedey, Schuchhardt, & Schuster, 1992; Cessac, Doyon, Quoy, & Samuelides, 1994; Schuecker, Goedeke, Dahmen, & Helias, 2016; Martí, Brunel, & Ostojic, 2018). In that case, the

---

Figure 1: Implementing the fixed-point task: network model and mean-field analysis in three example geometries. (a) Architecture of the feedback network (see equation 2.4). (b) Sample from the dynamics of two finite-size networks, $N = 1000$. In the left example, the output state, which corresponds to the target fixed point, is locally stable, and the readout signal $z(t)$ converges to the correct value. In the right example, the target instead corresponds to a locally unstable state, and the readout converges to an incorrect value. Gray traces indicate the time course of the activation variable $x$ for five randomly selected units. The readout trajectory is instead displayed in blue. The top-right insets display the mean-field prediction for the stability eigenspectrum of the target fixed point (see appendix A). Local instabilities are due to the outlier eigenvalue crossing the instability boundary (dashed lines). The parameters used in the left and the right examples are indicated by colored arrows in panel e. (c–e) Mean-field characterization of the network fixed points for three readout geometries (see the text). Continuous (resp. dashed) lines correspond to locally stable (resp. unstable) mean-field solutions. Shaded areas indicate the values of the target $A$ for which the association is locally unstable. Top row: Value of the readout signal. Bottom row: Position of the outlier eigenvalue in the stability eigenspectra. Gray lines indicate the value of the radius of the eigenvalues’ bulk. The instability boundary is indicated on the ordinate axis by the black arrows. The results of simulations are displayed as dots ($N = 3000$, average over eight network realizations). We integrate numerically the dynamics of finite-size networks where the readout vector $n$ is normalized through equation 3.12. In order to reach different solutions, we initialize the network dynamics in two different initial conditions, centered around $n$ and $-n$. Note that the position of the outlier eigenvalue cannot be measured when the outlier is absorbed in the circular set of the eigenspectrum. In order to ease the comparison between theory and simulations, we thus chose a small $g$ value. Parameters: $g = 0.3$, $\sigma_m = 1.2$, $\sigma_I = 0.5$, $\rho = 0.5$. In panel e, we take $p = p_m = 1$, $p_I = 0.3$. Here and in the following figures, error bars, when visible, correspond to the standard error.
recurrent input $\delta$ is approximated by a vector of uncorrelated white noise of self-consistent amplitude. Such an approximation is justified by the disorder in the random connectivity, which effectively decorrelates the input to different units. This procedure washes out any dependence on the specific instantiation of the random matrix $\chi$, so that classical mean-field theories cannot be easily used to describe solutions where the readout vector $n$ is tightly correlated with $\delta$.

For the simple task we consider here, one can focus on the restricted class of solutions where the readout $n$ is correlated only with the input vectors $m$ and $I$. We show in the next paragraph that the classical mean-field approaches can be easily used in this case to map any readout vector into a full description of the network output states. Because they do not rely on correlations with the random bulk $\chi$, readout solutions in this class generate an output that is robust with respect to changes in the random part of the connectivity.

From equation 2.5, a minimal and general solution in this class can be written as

$$n = \frac{c}{N} (p \xi + p_m \eta_m + p_I \eta_I).$$

(3.4)

The geometry of the readout vector is determined by the relative weights of the three coefficients $(p, p_m, p_I)$, which quantify how much the readout direction is distributed on the three axes defined by the nontrained part of the network architecture.

When the readout vector is aligned with one of the three axes $\xi$, $\eta_m$, or $\eta_I$, the population average in equation 3.3 takes finite $O(1)$ values (see below). As a consequence, training solutions in the form of equation 3.4 need to scale as the inverse of the network size ($n_i \sim O(1/N)$) in order to prevent diverging activity and outputs. The weights $(p, p_m, p_I)$ are furthermore normalized through the constant $c$, which fixes the output $z$ to the exact target value specified by the task.

In the remainder of this section, we characterize analytically the dynamics emerging in trained networks where the readout vector belongs to the restricted class defined by equation 3.4. To this end, we introduce the key equations of the mean-field description and analyze in detail three simple geometries $(p, p_m, p_I)$ that satisfy the task through rather different output dynamics. In sections 4 and 5, we exploit our simplified mean-field framework to analyze the more general readout solutions that are generated through learning by standard training algorithms.

### 3.1 Mean-Field Description.

For every fixed readout vector in the form of equation 3.4, we follow Mastrogiuseppe and Ostojic (2018) to derive a macroscopic, effective description of the network activity. As in standard mean-field theories for random recurrent networks (Sompolinsky et al.,
1988; Rajan, Abbott, & Sompolinsky, 2010; Kadmon & Sompolinsky, 2015), we approximate the equilibrium activation vector $\mathbf{x}$ by the statistical distribution of its elements. In particular, every element $x_i$ can be thought of as extracted from a gaussian distribution of variance $\Delta_i \equiv \langle x_i^2 \rangle - \langle x_i \rangle^2$, where $\langle \cdot \rangle$ indicates an average over the population and $\langle \cdot \rangle$ an average across different realizations of the random bulk.

From here on, we consider the overlap direction $\xi$ to be generated as a standard gaussian vector. In this case, the mean of the distribution of $x$ vanishes, as $\langle m_i \rangle = \langle I_i \rangle = \langle \chi_{ij} \rangle = 0$. As shown in Mastrogiuseppe and Ostojic (2018), if the overlap direction $\xi$ coincides instead with the unitary vector, the mean-field equations take a slightly different form, which includes finite mean values but results in qualitatively identical results.

In order to derive a self-consistent expression for $\Delta$, we first consider the average over the bulk connectivity $\chi$. By direct averaging of equation 3.1, we get

$$\mu_i = [x_i] = m_i [z] + I_i,$$

$$\Delta_i = [x_i^2] - [x_i]^2 = m_i^2 ([z^2] - [z]^2) + [\delta_i^2].$$

(3.5)

As in standard mean-field derivations (Sompolinsky et al., 1988), we have $[x_i x_j] = [x_i] [x_j]$ and

$$[\delta_i^2] = g^2 \sum_{j=1}^{N} \sum_{k=1}^{N} [\chi_{ij} \chi_{ik}] [\phi(x_j) \phi(x_k)] = g^2 \sum_{j=1}^{N} [\chi_{ij}^2] [\phi(x_j)^2] = g^2 \langle [\phi(x_i)^2] \rangle,$$

(3.6)

since $[\chi_{ij}^2] = 1/N$. In the thermodynamic limit, furthermore, the variance of $z$ vanishes, as $n_i$ is characterized by a weaker scaling with $N$ (from equation 3.4, $n_i^2 \sim \mathcal{O}(1/N^2)$). The readout signal $z$ is thus self-averaging in large networks. For this reason, we drop the brackets in $[z]$ and compute the readout signal self-consistently through

$$z = c \{ p \xi_i + p m_i \eta_{mi} + p I_i \eta_{Ii} \} [\phi(x_i)].$$

(3.7)

Note that equation 3.5 allows effectively replacing the recurrent input $\delta$ with white noise of self-consistent amplitude,

$$x_i = \mu_i + \delta_i \sim \mu_i + \sqrt{\Delta_i} w_i,$$

(3.8)

where $\Delta_i = g^2 \langle [\phi(x_i)^2] \rangle$ and $w_i$ is a standard gaussian variable.
By averaging again across units, we find that the population distribution of the equilibria \( x_i \) obeys the following statistics:

\[
\begin{align*}
\mu &= \langle x_i \rangle = 0, \\
\Delta &= \langle x_i^2 \rangle - \langle x_i \rangle^2 = g^2 \langle [\phi(x_i)^2] \rangle + \sigma_m^2 z^2 + 2 \sigma_m z + \sigma_I^2, \\
\end{align*}
\]

(3.9)

where \( \sigma_{mI} = \langle m_i I_i \rangle = \rho^2 \sigma_m \sigma_I \).

In order to obtain a closed form for our equations, we derive a self-consistent expression for the average quantities \( z \) and \( \langle [\phi(x_i)^2] \rangle \). As in standard mean-field theories, we reexpress averages through gaussian integrals,

\[
\langle [\phi(x_i)^2] \rangle = \int D w \phi^2 (\sqrt{\Delta} w),
\]

(3.10)

where we used the shorthand notation \( D w = d w e^{-w^2/2\sqrt{2\pi}} \). By combining equations 2.5 and 3.7 and integrating by parts over \( \xi_i, \eta_{mi} \) and \( \eta_{Ii} \), we furthermore get (Mastrogiuseppe & Ostojic, 2018)

\[
z = c \left\{ p(\sigma_m \rho z + \sigma_I \rho) + p_m \sigma_m \sqrt{1 - \rho^2} z + p_I \sigma_I \sqrt{1 - \rho^2} \right\} \langle [\phi'(x_i)] \rangle,
\]

(3.11)

where the average \( \langle [\phi'(x_i)] \rangle \) can be evaluated similar to equation 3.10. For any fixed set of readout weights \( (p, p_m, p_I) \), equations 3.9 and 3.11 can be solved together through standard numerical techniques to return the values of \( \Delta \) and \( z \), which characterize the different output fixed points.

To conclude, we observe that the equations above apply only to networks that satisfy the task through stationary output states. In general, stationary outputs do not require stationary reservoir activity. For example, solutions in the form of equation 3.4 can be used to solve the task even when activity in the bulk is chaotic, but fluctuations cancel out at the level of the readout (Mastrogiuseppe & Ostojic, 2018). In that case, the mean-field description above can easily be extended to take temporal fluctuations into account (Sompolinsky et al., 1988). Standard training techniques, however, typically fail to converge when the target of the task is not strong enough to suppress chaotic activity in the bulk (Rivkind & Barak, 2017). For this reason, here and in the following, we concentrate on fixed-point dynamics.

### 3.2 Determining Task Solutions.

For every fixed set of weights \( (p, p_m, p_I) \), first the normalization parameter \( c \) needs to be determined. We fix \( c \) by requiring that in the final network output, the readout value exactly matches the target \( A \). To this end, we consider the open-loop formulation of equations 3.9 and 3.11, obtained by setting \( z = A \), and we solve the system
for $c$ and $\Delta^{ol}$. This gives

$$c = \frac{A}{\left\{ p(\sigma_m \rho A + \sigma_I \rho) + p_m \sigma_m \sqrt{1 - \rho^2 A + p_I \sigma_I \sqrt{1 - \rho^2}} \right\} \langle \phi'(x^{ol}_i) \rangle},$$

(3.12)

where $\langle \phi'(x^{ol}_i) \rangle$ is computed as a gaussian integral over a distribution of variance $\Delta^{ol}$ (see equation 3.10).

Once the normalizing factor has been derived, the full mean-field equations can be solved. Because of the normalization, the mean-field system of equations always admits a stationary solution, which satisfies the task for $N \to \infty$ (i.e., its readout obeys $z = A$). This solution, however, is not guaranteed to be locally stable from the point of view of the full feedback dynamics (see Figure 1b). Apart from the target solution, furthermore, the mean-field equations might admit other solutions corresponding to stable fixed points. In those cases, depending on the initial conditions, the dynamics of the final network might converge to spurious output states where the readout $z$ significantly deviates from the target.

For every solution of the mean-field equations, we predict local stability by evaluating the eigenspectrum of the linear stability matrix of the corresponding fixed point. For the class of readouts that we consider, the eigenspectra consist of a dense set of eigenvalues, distributed within a circle in the complex plane, together with a single real outlier (see Figure 1b) (Mastrogiuseppe & Ostojic, 2018; Tao, 2013; Rivkind & Barak, 2017). The radius of the circular set and the position of the outlier eigenvalue can be evaluated within the mean-field framework (Mastrogiuseppe & Ostojic, 2018); their value is determined by the interplay between the amplitude of the random bulk and the relative geometrical arrangement of vectors $n, m, and I$.

Details of the calculations are provided in appendix A.

### 3.3 Categories of Readout Solutions.

We specifically look at the results obtained for three readout vectors characterized by different geometries.

In the first example we analyze (see Figure 1c), the readout $n$ overlaps solely with the feedback input $m$ along $\eta_m$. This configuration corresponds to setting $p = p_I = 0$ and can be realized also in the absence of the external input vector. In this case, the network admits three stationary states for every value of the target $A$. Among them, only two states are locally stable. Bistability is due to the strong overlap between the feedback vectors $m$ and $n$ and emerges through a pitchfork bifurcation induced by the outlier eigenvalue in the stability eigenspectrum of the feedback-free fixed point (Hopfield, 1982; Mastrogiuseppe & Ostojic, 2018). As a consequence, stationary states tuned to small target values are characterized by long relaxation timescales. For most of the target values, the solution corresponding to the target (i.e., $z = A$) is locally stable. However, the network admits a
second stable fixed point, characterized by a different readout value. This additional stable state is reached when the network is initialized within the appropriate basin of attraction. The position of the unstable fixed point (dashed lines in Figure 1) can be used to estimate the size of the basins of attraction of the two stable fixed points, which in this case is approximately equal. Within a small parameter region corresponding to very small target values, the target fixed point is built-in correspondence of the intermediate, locally unstable branch of the mean-field solutions. The mean-field theory indicates that the local instability is due to the outlier eigenvalue of the fixed-point stability matrix lying above the critical line (see Figure 1b, right). The amplitude of the instability region is controlled by the overlap between the two fixed vectors \( \mathbf{m} \) and \( \mathbf{I} \) and disappears when \( \rho = 0 \) (Rivkind & Barak, 2017).

In the second example we consider (see Figure 1d), the readout \( \mathbf{n} \) overlaps solely with the external input weights \( \mathbf{I} \) along \( \eta \). We thus set \( p = p_m = 0 \). Analogous to a feedback-free network, aligning the readout vector with the external input generates a unique stationary solution, corresponding to the target of the task (\( z = A \)). Such a solution is always stable, and it is characterized by negative values of the outlier eigenvalue.

As a last example, we consider the general case where the readout displays nonvanishing overlaps with both vectors \( \mathbf{m} \) and \( \mathbf{I} \) (see Figure 1e). Such a category includes the simplest configuration where the readout \( \mathbf{n} \) is aligned with the overlap direction \( \xi \) (\( p_m = p_I = 0 \)), but more general solutions characterized by nonzero projections along \( \eta_m \) and \( \eta_I \) are included as well. As in the first example we considered (see Figure 1c), the components of the readout that overlap with the feedback input \( \mathbf{m} \) tend to generate three different fixed points. Because of the components along the external input \( \mathbf{I} \), however, these solutions are no longer characterized by a strong symmetry around zero. As a result, the unstable intermediate fixed point is always characterized by finite output values. This unstable state corresponds to the task solution within a finite parameter region at small and negative target values. This instability area is delimited by a critical target value \( A^* \), where the readout amplitude \( c \), together with the stable fixed points, diverges:

\[
A^* = -\frac{p_m \sigma_I \rho + p_I \sigma_I \sqrt{1 - \rho^2}}{p_m \sigma_m \rho + p_m \sigma_m \sqrt{1 - \rho^2}}.
\]  

(3.13)

Note that this geometry produces a unique and stable input-output association within a parameter window corresponding to small and positive target values. For large (positive or negative) target values, instead, the projection along \( \mathbf{m} \) dominates, and a second stable fixed point exists.

To conclude, we characterized the output states generated by readout solutions in the class defined by equation 3.4. We found that for different solutions belonging to that class, the task is satisfied through output states, which are characterized by different local and global stability properties.
Stability can be predicted via the mean-field theory by examining the projections of the readout \( \mathbf{n} \) on the input vectors \( \mathbf{m} \) and \( \mathbf{I} \). In particular, an optimal solution, characterized by both local and global stability, can be obtained by aligning the readout \( \mathbf{n} \) along the nonshared direction of the external input \( \mathbf{I} \). This solution requires a nonvanishing input pattern and generates a feedback network that is dynamically analogous to the feedback-free implementation. Components along the feedback input \( \mathbf{m} \), on the other hand, tend to generate additional stable solutions, which can attract the dynamics to different readout values. Combining both readout components along \( \mathbf{m} \) and \( \mathbf{I} \), finally, generates nonsymmetric bistable solutions, which result in broad regions of local and global instability.

4 Least-Squares Training

In section 3, we derived a theoretical framework that allows us to directly map the geometry of the readout vector into the final output states of the feedback network. This description applies to networks that obey three closely related assumptions (see equation 3.4): (1) the readout is a linear combination of the input axes \( \xi, \eta \); (2) no correlations exist between the readout and the random bulk; and (3) the scaling of the solution with the network size \( N \) is weak (i.e., \( n_i \) is of order \( 1/N \)).

Importantly, these assumptions might not hold in finite-size networks obtained through standard training techniques. Trained networks might indeed converge to more complex, nonlinear solutions, where strong correlations with the random connectivity significantly influence the output dynamics. In sections 4 and 5, we investigate how far our approximate theory can be used to describe trained networks. To this end, we construct for every training solution a crude approximation in the form of equation 3.4, for which the mean-field equations can be solved and the global dynamics of the resulting network can be predicted. Since in the mean-field description, correlations with the random bulk are neglected, comparing mean-field predictions with training performance allows us to quantify how much training relies on a fine-tuning of the readout to the random connectivity \( \chi \). In cases where the theory predicts that no stable solutions exist, furthermore, our analysis allows investigating whether finite-size trained networks converge to more effective solutions that the simplified theoretical description cannot capture.

We consider two specific cases, corresponding to two classical training protocols. The first training procedure we consider is the simple batch update through least-squares (LS) inversion (Jaeger, 2001; Lukosevicius, 2012; Rivkind & Barak, 2017). To begin, a finite-size architecture defined by a random bulk \( \chi \) and two vectors \( \mathbf{m} \) and \( \mathbf{I} \) is generated. The activity of the whole population in the open-loop configuration is then simulated, yielding

\[
x_{ol}^i = m_i A + I_i + \delta_{ol}^i.
\]  

(4.1)
The task imposes a unique constraint on the readout, namely, \( \phi(x^i)_T n = A \). This equation represents an underdetermined linear system for the \( N \) entries of vector \( n \). The minimum norm LS solution can be computed through the pseudo-inverse of \( \phi(x^i)_T \) (Bretscher, 2009),

\[
n = \frac{A}{N(\phi^2(x^i))} \phi(x^i), \tag{4.2}
\]

and thus scales with the network size as \( 1/N \).

Through the open-loop activation vector \( x^d \), the LS readout inherits three \( N \)-dimensional components along vectors \( m \), \( I \), and \( \delta^d \) (see equation 4.1). The nonlinear activation function \( \phi(x) \), which is applied to \( x^d \) (see equation 4.2), mixes and reshapes these three directions by generating a continuum of additional components along orthogonal axes. Furthermore, the component of \( x^d \) aligned with the recurrent input \( \delta^d \) introduces finite, \( O(1) \) correlations between the readout and the finite-size instantiation of the random bulk used for training. Note that a fraction of those correlations is used by the LS algorithm to exactly match the readout \( z \) to the target \( A \) within network instantiations of any size. Mean-field solutions (see equation 3.4), instead are independent of the random bulk and imply small readout errors (of the order of \( 1/\sqrt{N} \)), which disappear only in the thermodynamic limit.

4.1 Mean-Field Approximation. We construct an approximate description of \( n \) (see equation 4.2), which conserves the relative strength of its components along the input vectors \( m \) and \( I \) (see equation 3.4), thus totally neglecting the components originating from the recurrent input \( \delta^d \) and the nonlinearity. To this end, we extract the linear projections of the readout vector along the orthogonal axes \( \xi \), \( \eta_m \), and \( \eta_i \), resulting in a set of weights \( (p, p_m, p_I) \) that effectively replace the LS solution with an approximate readout in the form of equation 3.4. For such an approximate readout, the network output states can be exactly predicted.

As we show in the next paragraphs, corrections to the mean-field description due to both neglected terms \( \delta^d \) and \( \phi(x) \) in this case can be exactly computed (Rivkind & Barak, 2017). A minimal approximation is, however, of interest per se, as it directly generalizes to more complex readout vectors for which analytical expressions do not exist, so that the exact form of correlations and the shape of nonlinear components are not known a priori (see section 5).

The projection of \( n \) on \( \xi \) can be computed as

\[
p = \frac{A}{N(\phi^2(x^i))} \langle \xi; \phi(x^i) \rangle \]

\[
= \frac{A}{N(\phi^2(x^i))} \langle \xi; \phi \left( \sigma_m (\rho \xi_i + \sqrt{1 - \rho^2} \eta_{mi}) A \right) \rangle
\]
\[ + \sigma_I (\rho \xi_i + \sqrt{1 - \rho^2} \eta_i) + \delta^a_i \].

(4.3)

We reexpress the population average as a gaussian integral and integrate by parts over \( \xi_i \), yielding
\[
p = A N(\phi^2(x^a_i)) \rho (\sigma_m A + \sigma_I) (\phi'(x^a_i)).
\]

(4.4)

Similar calculations return the values of \( p_m \) and \( p_I \). The final set of weights is given by
\[
p = \gamma \rho (\sigma_m A + \sigma_I),
p_m = \gamma \sigma_m \sqrt{1 - \rho^2} A,
p_I = \gamma \sigma_I \sqrt{1 - \rho^2},
\]

(4.5)

where we defined the common multiplicative factor \( \gamma = A \langle \phi'(x^a_i) \rangle / N(\phi^2(x^a_i)) \).

As in Figure 1e, the LS solution thus includes nonvanishing components along the three axes \( \xi, \eta_m, \) and \( \eta_I \). In contrast to Figure 1, however, the values of \( (p, p_m, p_I) \) that have been derived from the LS solution display an explicit dependence on the target value \( A \). For fixed vectors \( m \) and \( I \), the geometrical arrangement of the readout vector thus changes according to the target of the task.

Starting from equation 4.5, we evaluate the normalization factor (see equation 3.12), and we solve the system of mean-field equations (see equations 3.9 and 3.11). In Figure 2, the solutions and their stability properties are illustrated.

We find that the shape of the mean-field solutions critically depends on the geometrical arrangement of the nontrained part of the network architecture. Specifically, the network output is strongly influenced by the value of the overlap between the input vectors \( m \) and \( I \), quantified by the parameter \( \rho \).

When the input vectors \( m \) and \( I \) are orthogonal or share a weak overlap, the theory admits a continuous stable solution—on which the target output is built—together with two symmetric stable branches at large target values (see Figure 2a). In this case, the theory predicts that training always generates locally stable dynamics (Rivkind & Barak, 2017), but spurious fixed-point attractors are generated at large \( A \). As the overlap \( \rho \) is increased, the branch of the solution corresponding to the target merges with one of the two additional fixed points, generating a region of local instability at small and negative target values (see Figure 2b). In the limit where the input vectors \( m \) and \( I \) are fully aligned (\( \rho = 1; \) see Figure 2c), finally, mean-field solutions are similar to the results in Figure 1e. The network is characterized by a region of local instability, which is delimited by a critical target value.
Figure 2: The least-squares solution: approximate mean-field description. Mean-field characterization for three different configurations of the nontrained input vectors $\mathbf{m}$ and $\mathbf{I}$. In panel a, we take $\rho = 0.6$; in panel b, $\rho = 0.97$; in panel c, $\rho = 1$. Continuous and dashed lines indicate the solutions of the simplified mean-field description, where the readout is approximated by a vector in the form of equation 3.4. Details are as in Figure 1. The light green line indicates the exact value of the outlier eigenvalue, which measures the local stability of the target solution (Rivkind & Barak, 2017); details are provided in appendix B. Dots display the results of simulations from finite-size networks ($N = 3000$, averages over eight network realizations). We integrate numerically the dynamics of finite-size networks where the readout vector $\mathbf{n}$ is given by the LS solution (see equation 4.2). In order to reach different solutions, we initialize the network dynamics in two different initial conditions, centered around $\mathbf{n}$ and $-\mathbf{n}$. Parameters as in Figure 1.

$A^*$ where the dynamics diverge. The normalization factor $c$ is in this case given by:

$$c = \frac{A}{\rho \left( \sigma_I + \sigma_mA \right)^2 \langle \phi'(x_{ol}^d) \rangle}, \quad (4.6)$$

which diverges at $A^* = -\sigma_I/\sigma_m$. 

Downloaded from http://www.mitpressjournals.org/doi/pdfplus/10.1162/neco_a_01187 by guest on 16 May 2021
In Figure 2, mean-field predictions are compared with the outcome of simulations performed in finite-size networks trained through LS inversion. Despite the approximate nature of the theory, a good agreement between the two is found. As predicted by the theory, output states of trained networks are strongly affected by the value of the input overlap $\rho$. In particular, the theoretical approximation correctly predicts the existence of an area of local instability for large $\rho$ values. Within this parameter region, training generates an output $z(t)$ that immediately diverges from the desired target. For every value of $\rho$, furthermore, the theory captures the existence of a second, spurious fixed point at large target values.

The mean-field prediction for the outlier stability eigenvalue can be further compared with an exact analytical expression derived by following the analysis in Rivkind and Barak (2017). This expression, provided in appendix B, includes the components of the readout $n$, which are neglected in the simplified mean-field description: correlations with the bulk $\chi$ and nonlinear geometry. We find that apart from minor quantitative discrepancies, the exact outlier prediction is well matched by the approximate one. Note that this exact analytical expression provides only a measure of local stability, as it can be evaluated only in correspondence to the target solution.

To conclude, we found that dynamics in LS trained networks can be explained by a highly simplified, approximate theoretical description that conserves the projection of the readout vector on the hyperplane spanned by the feedback and external input vectors. The approximate mean-field theory specifically predicts that LS readouts often generate bistable fixed points, while unstable solutions can be encountered for large values of the overlap $\rho$. Note that as long as the input vectors $m$ and $I$ are not completely parallel (i.e., $\rho < 1$), the mean-field framework indicates that an optimal readout solution can be designed by aligning the readout $n$ with the non-shared component of $I$ (see Figure 1d). We conclude that even for the simple task we consider, LS training does not lead to an optimal solution that prevents dynamical instabilities.

Apart from an overall quantitative agreement, mismatches between simulations and mean-field solutions can be observed. Their size depends on the value of the target and the network parameters. A brief description of the role and impact of the different sources of mismatch is presented in section 4.2.

4.2 Nonlinear Corrections. Together with its components along the input axes $\xi$, $\eta_m$, and $\eta_I$, the full LS readout (see equation 4.2) includes a continuum of orthogonal directions that can potentially affect the network dynamics and output values. We first focus on the orthogonal components, which are generated when the nonlinear function $\phi(x)$ is applied to $x^d$ (see equation 4.2).

We build a more precise mean-field approximation, which conserves the nonlinearity of the LS solution (see Figure 3). Details of the analysis
are provided in appendix D. By solving the novel set of mean-field equations, we altogether find that nonlinear solutions lie very close to the ones we obtained for linearly approximated readouts (see Figure 3, left). The simple linear approximation therefore appears to capture well the readout geometry that is relevant for the final network dynamics. At a finer scale, we find that nonlinear solutions in some cases explain with higher precision the output states observed in trained networks. In particular, we observe that the agreement significantly improves in the parameter regions corresponding to large and positive target values, where the dynamics are strongly nonlinear and admit bistable fixed points (see Figure 3, right).

4.3 The Effect of Correlations. Our approximate mean-field description neglects a second set of orthogonal readout directions, generated by the recurrent input \( \delta^o \) (see equation 4.1). When this vector is replaced with gaussian noise, correlations between the LS readout and the random part of the connectivity are effectively washed out.

In order to boost the effect of correlations, we increase the strength of random connections \( g \) up to the critical point where dynamics become chaotic (see Figures 4a and 4b; note that the rank-one structure and the external input pattern shift the critical coupling from \( g = 1 \) to slightly larger values; Rajan et al., 2010; Schuecker, Goedeke, & Helias, 2018). For every value of \( g \), we observe that the approximate mean-field description correctly captures performance in LS-trained networks when the two input vectors \( \mathbf{m} \) and \( \mathbf{I} \) are not completely aligned (\( \rho < 1 \); see Figure 4a). On the other hand, we find that the theory fails to fully describe degenerate architectures with parallel input vectors (\( \rho = 1 \); see Figure 4b): a significant qualitative mismatch is
Figure 4: The least-squares solution: effect of correlations. (a, b) Comparison between the linear mean-field predictions and the trained networks’ output states for two increasing values of $g$, corresponding to pink ($g = 0.6$) and magenta ($g = 1.1$) traces. We consider network architectures characterized by two different values of the input vectors overlap. In panel a, we take $\rho = 0.6$; in panel b, $\rho = 1$. Note that especially in panel a, the results for the two values of $g$ strongly overlap. A significative mismatch between theory and simulations is observed in panel b, for $g = 1.1$, within the instability area at small and negative target values. (c) For $\rho = 1$, we fix the value of the target $A$ within the instability window (gray arrows in panel b), and we measure the mismatch between mean-field predictions and trained networks as the random strength $g$ is increased. The mismatch is measured at the level of the readout $z$ in correspondence of the spurious fixed point characterized by positive readout values. Parameters and simulations settings are as in Figure 2.

observed within the instability region corresponding to small and negative target values. In such a region, nonlinear corrections have very little effect (see Figure 3), and the mismatch is mostly due to correlations between the readout vector and the random bulk. The mismatch indeed increases with the random strength $g$ as the recurrent input deriving from the random connectivity becomes large with respect to the feedback input (see Figure 4c).

A detailed analysis of degenerate feedback architectures ($\rho = 1$) is provided in Figure 5. In the top row, we select an intermediate value of $g$. In this case, dynamics in trained networks diverge at the border of the instability window as predicted by the mean-field theory (see Figure 5a). In the bottom row, we consider a larger value of $g$, lying right below the instability to chaos. In this case, the trained network output appears to obey a different qualitative pattern: the largest stable solution, which is predicted to diverge at $A^*$, converges instead at the border of the instability window.

Our approximate mean-field description predicts, furthermore, a large instability window, wider than the instability region predicted by the exact analytical expression (Rivkind & Barak, 2017); (light-shaded regions in
Figure 5: The least-squares solution: effect of correlations; detailed analysis of the case $\rho = 1$ (see Figure 4). The results for the two different values of $g$ are displayed separately in the top ($g = 0.6$) and in the bottom ($g = 1.1$) panels. (a) Magnified view of Figure 4b. Arrows on the abscissa axis indicate the target values used in panel c. (b) Exact prediction for the outlier eigenvalue of the stability matrix of the target fixed point (in green; see appendix B). The outlier eigenvalue lies above the instability boundary within a parameter region corresponding to the exact instability area, indicated in light gray. The instability area predicted by the approximate mean-field theory is indicated in dark gray. Gray lines indicate the value of the radius of the eigenvalues bulk. (c) Probing the basin of attraction of the target solution for two different values of $A$, indicated by arrows in panel a. We simulate the post-training network dynamics starting from initial conditions given by gaussian vectors of variable amplitude added on top of the open-loop activity $x^0$. The absolute error is then measured at the level of the readout $z$ and is averaged over 40 trained networks. Parameters are as in Figure 2.

Figures 5a and 5b). The target fixed point remains thus locally stable within a parameter region where the approximate theory predicts instability. Within the same window, however, the target solution is characterized by a stability eigenvalue lying very close to the instability boundary and a very narrow basin of attraction (Figures 5b and 5c). Trained networks indeed converge to the target fixed point only when the initial conditions
of the dynamics are very carefully tuned to the open-loop activation vector $x^\text{OL}$. Since the dynamics systematically diverge from the target solution when the initial conditions are perturbed along random directions, Figure 5c indicates that the basin of attraction of such a fixed point is particularly narrow with respect to any axis of the phase space.

To conclude, we found that correlations affect the agreement between mean-field theory and trained networks within a restricted region of parameters, corresponding to almost parallel input vectors. In that region, a quantitative and qualitative match cannot be obtained when the ratio $g/A$, which measures the relative strength of the recurrent over the feedback dynamics, is large. The approximate mean-field description, however, successfully captures the presence of an instability area, where the target fixed point is unstable or confined to a very narrow basin of attraction.

5 Recursive Least-Squares Training

Our analysis of least-squares solutions revealed the presence of a finite parameter region where the target fixed point is constructed on a locally unstable state. Dynamical feedback instabilities have been observed in a variety of studies where, as in the LS case we examined so far, the readout vector $n$ is fixed through a unique and batch weights update (Jaeger, 2001; Lukosevicius, 2012; Reinhart & Steil, 2011).

A convenient strategy to overcome local instabilities consists of training feedback networks with an online algorithm so that the effect of synaptic changes is immediately propagated and tested on the network dynamics. FORCE (first-order reduced and controlled error) learning (Sussillo & Abbott, 2009) is a popular example that relies on an online and recursive formulation of the least-squares algorithm (RLS; Liu, Principe, & Haykin, 2010; Jaeger, 2002). This algorithm includes a regularization term, whose strength is controlled by a parameter $r$, which penalizes large-amplitude readouts. Details of the training procedure are reported in appendix E.

In this section, we train feedback networks with FORCE and apply our theoretical framework to derive an approximate but effective description of the solutions reached. Similar to the previous section, we can investigate how the algorithm implements the task by replacing the trained readout vector with an approximation in the form of equation 3.4. Specifically, we are interested in understanding whether—and, eventually, how—RLS solutions overcome the local instabilities that are encountered by the batch version of the LS algorithm.

5.1 Parallel Geometry. We start our analysis from a specific case, corresponding to a particularly simple network architecture: the two input vectors $m$ and $I$ are taken to be parallel ($\rho = 1$). In this case, the mean-field theory for uncorrelated solutions (see equation 3.4) predicts that a unique solution exists, for which the readout vector $n$ is parallel to the overlap axis.
Figure 6: Recursive least-squares solutions ($\rho = 1$): analysis of the readout vector. We consider a set of $N$ unitary and orthogonal vectors $\{v^k\}_{k=1,\ldots,N}$, which includes the input direction $\xi$, and we decompose the trained readout $n$ on such an orthonormal basis. We construct an histogram by collecting the strengths of the projections of $n$ along the different basis vectors. Different histogram colors indicate three different training trials, where the norm of the initial guess for $n$ has been varied. The initial guess is generated as a gaussian random vector of SD $Ns$. The value of $s$ thus controls the amplitude of the initial guess with respect to the network size. The inset reports a magnified view of the histogram for small ordinate values. The gray arrow indicates the average strength of the component of $n$ along the input axis $\xi$. The amplitude of the projections along the remaining basis vectors, instead, is strongly modulated by the initial amplitude $s$. Parameters: $g = 0.3$, $\sigma_m = \sigma_l = 1.2$, $\rho = 1$, $A = 1.6$.

$\xi$, while orthogonal readout directions do not affect the output $z$. As in Figures 1e and 2c, such a readout unavoidably generates regions of local and global instability. Here we ask whether RLS solutions implement the task through a similar strategy or whether alternative and more effective solutions are found outside the restricted class of readouts which are described by the mean-field theory (see equation 3.4).

To start, we run the RLS algorithm up to convergence and then analyze numerically the geometry of the reached solution $n$ (see Figure 6). We find that regardless of the network size, the value of the parameters, and the initial conditions, the readout vector is systematically characterized by a large component aligned with the input direction $\xi$ (gray arrow in the inset of Figure 6). The readout vector, however, also includes strong components oriented in orthogonal directions; such additional components are induced by nonlinearities, correlations with the random bulk, and initial conditions of training. In particular, since training convergence is often very fast, a large fraction of orthogonal components is directly inherited from the geometry of the initial guess for the readout $n$, which is typically generated at random (see Figure 6). As a consequence, the amplitude of orthogonal components is strongly affected by the norm of the random vector which is used as an initial guess for $n$. In Figure 6, the initial readout is generated as a gaussian vector of standard deviation $Ns$, so that the amplitude of
Orthogonal components is controlled by parameter $s$, which measures the amplitude of the initial guess with respect to the bulk size.

As a second step, we train feedback networks to different target values, and we attempt to predict performance through the help of the mean-field framework (see Figure 7). As in the previous section, our approximate theoretical description conserves the component of $n$ which is parallel to $\xi$, and discards any orthogonal direction. In equation 3.4, we thus set $p = 1$, $p_m = p_I = 0$.

As in Figures 1e and 2c, the mean-field theory predicts an area of local instability at small and negative target values (gray shaded regions in Figure 7). In order to evaluate the local stability of the fixed point generated by FORCE, we compute the post-training dynamics by using the activation vector from the last training step ($x^{end}$) as an initial condition, and we finally measure the error at the level of the readout $z$. Remarkably, we find that the RLS algorithm often fails to converge to a locally stable solution when the target $A$ is taken within the instability window predicted by the mean-field theory (see Figures 7a–7c). The failure of the algorithm is thus correctly explained by our crude theoretical approximation, which replaces the complex readout emerging from training with a unique component along $\xi$.

Although some variability exists, this result is robust with respect to a broad choice of network and training parameters. In particular, the average post-training error decays but remains positive for both small and large values of the random strength $g$ (see Figure 7b). Interestingly, training evolves differently in the two cases (see Figure 8). For small values of $g$, the algorithm never converges (see Figure 8a). The readout error remains large during training, while the amplitude of the weights modification imposed by the algorithm smoothly decays. When the value of $g$ is large and close to the instability to chaos, instead, continuous weight modifications make the readout $z$ stably converge to the target within the majority of the trials (see Figure 8b, bottom), resulting in a small average error (see Figure 7b). These strongly correlated training solutions, which allow feedback networks to overcome the dynamical instability at small and negative target values, cannot be captured within our approximate theoretical framework. We conclude that large $g$ values, corresponding to strong random connectivities, help the algorithm find fine-tuned solutions whenever weak and uncorrelated readout vectors (see equation 3.4) cannot generate stable fixed points. In certain trials, however, the algorithm fails to converge (see Figure 8b, top) or generates fine-tuned solutions that diverge from the target after learning. As a consequence, the average post-training error never completely vanishes (see Figure 7b).

As in Figures 1e and 2c, the mean-field approximation further predicts the existence of an additional stable fixed point for large—positive and negative—target values (red-shaded regions in Figures 7d–f). Such a fixed point is characterized by a readout value of opposite sign and is responsible for large readout errors whenever the dynamics is initialized within
Figure 7: Recursive least-squares solutions ($\rho = 1$): comparison between mean-field predictions and trained networks. The region of local instability predicted by the theory is displayed in gray. The regions where the mean-field theory predicts the existence of a second stable fixed point are instead indicated in panels d–f by horizontal red stripes. In panels a–c, we quantify the local stability of the training solution. To this end, the post-training activity is simulated for 50 normalized time units, where the initial condition is set by the activation variable during the last training step $x^\text{end}$. The error is finally measured as $|z - A|$. In panels d–f, we test whether the network admits two bistable states. We simulate dynamics starting from $x^\text{end}$ and $-x^\text{end}$ as initial conditions ($n$ and $-n$ give similar results), and we compare the final value of the readout $z$ in the two cases. The trained dynamics are considered to be bistable if these two values have opposite signs. Dots show results averaged over 50 network realizations of size $N = 600$. Here and in the following, continuous lines do not indicate analytical results but are drawn to guide the eye. (a, d) Training results for different values of the regularization parameter $r$ (details provided in appendix E). (b, e) Training results for different values of the random strength $g$. (c, f) Training results for different values of the scaling $s$ of the initial guess for the readout $n$. Parameters as in Figure 6.

its basin of attraction. In order to assess bistability, we integrate the post-training dynamics starting from initial conditions centered around $x^\text{end}$ and $-x^\text{end}$. Again, a good agreement with the mean-field prediction is found: trained networks converge to a second stable fixed point when the target
Figure 8: Recursive least-squares solutions (\( \rho = 1 \)): evolution of training for a target value within the instability window (\( A = -1.1 \)). The top rows display the time course of the readout variable \( z \), while the bottom rows show the amplitude of the synaptic modifications in \( n \) imposed by the algorithm. The training window corresponds to the shaded region, and the target \( A \) is indicated by the dashed black line. (a) \( g = 0.3 \); (b) \( g = 1.1 \). In panel b, we show a nonconverging (top) and a converging (bottom) trial. Parameters as in Figure 6.

value is taken within the parameter regions where the approximate mean-field network is bistable.

The systematic agreement between theory and simulations (see Figure 7) altogether indicates that the readout component along \( \xi \) plays a major role in shaping the dynamical landscape of trained networks. Orthogonal readout directions, which are artificially included in the readout vector during training, can have strong amplitude (see Figure 6) but contribute little to the network output states.

As a final step, we look at the eigenvalues of the effective connectivity matrix \( \chi + mn^T \) (see equation 2.4). The theory predicts that for a solution in the form of equation 3.4, one real outlier appears in the eigenspectrum. In a large network, its value corresponds to the overlap between the feedback vectors \( mn^T \) (Tao, 2013; Mastrogiuseppe & Ostojic, 2018). We find that when the initial guess for the readout \( n \) is small, the amplitude of those additional components is weak (see Figure 6), and the eigenspectrum is close to the theoretical prediction (see Figure 9a, top). Specifically, the eigenspectrum
includes a single real outlier whose position is well approximated by the scalar product $m^Tn$. The outlier eigenvalue gets closer to the predicted value as the training parameter $r$, which penalizes large readout weights, is increased (see Figures 9a and 9b). When the initial guess is characterized by strong scaling, the shape of the eigenspectrum becomes very variable from one trial to the other. Typically, the eigenspectrum includes more than one outlier, the position of which fluctuates strongly independently of the theoretical prediction (see Figure 9a, bottom).

Taken together, those results suggest that the connectivity eigenspectrum may bear little information about the overall dynamics of the trained network. The eigenspectrum of the effective connectivity matrix $\chi + mn^T$ indeed appears to be very sensitive to the readout components, which are spuriously introduced by the training algorithm but contribute little to the final output states. Our mean-field description, which specifically ignores such components, successfully captures the fundamental traits of local and global dynamics in trained networks but fails to predict the
shape of the complex eigenspectra displayed by the synaptic connectivity matrices.

5.2 Arbitrary Geometry. We complete our analysis by considering the RLS solutions that emerge from training in feedback architectures characterized by arbitrary initial geometries ($\rho < 1$).

In section 4, we found that the least-squares solution generates a region of local instability when the overlap between the two input vectors $m$ and $I$ is larger than a finite limit value (see Figure 2). However, our theoretical framework suggests that as long as the two input vectors are not completely aligned ($\rho < 1$), a stable mean-field solution always exists. The optimal readout $n$ includes a strong component along $\eta_I$—the direction of the external input $I$, which is not shared with the feedback input vector $m$ (see Figure 1d).

In Figure 10a, we select a value of the target for which the LS solution is unstable within a finite parameter region at large $\rho$ values. We then test the stability of the fixed point generated by FORCE training by continuously varying the value of the overlap parameter $\rho$. In agreement with the mean-field theory, we find that the RLS algorithm systematically converges to a locally stable solution whenever the value of the overlap between the two input vectors is significantly smaller than one. Specifically, the algorithm converges inside and outside the shaded region where the least-square solution is unstable.

In order to understand how the instability is overcome, we fix the value of the overlap $\rho$ within the instability window and systematically compare the geometry of the LS and the RLS solutions (see Figures 10b–10d, top). As in section 4, we extract the dominant readout geometries by projecting the readout vectors onto the three orthogonal axes $\xi$, $\eta_m$, and $\eta_I$ (see equation 3.4), resulting in sets of weights ($p, p_m, p_I$) from which dynamics can be predicted.

We find that the LS solution contains strong components along $\xi$ and $\eta_m$, together with a weak and positive component along the nonshared input direction $\eta_I$ (see Figure 10b, top). Importantly, because of equation 4.5, the value of $p_I$ is constant with respect to the target value $A$. This solution generates a window of local instability at small target values (see Figure 2b) because of the interaction between the strong components along $\eta_m$ and $\xi$, which tend to produce symmetric and bistable solutions, and the small component along $\eta_I$, which weakly suppresses bistability and disrupts such a symmetry. In particular, the instability area appears at negative target when the value of $p_I$ is positive.

The RLS algorithm, on the other hand, converges to a readout vector that includes a strong component along $\eta_I$ (see Figure 10c, top). The value of $p_I$, furthermore, is not constant but varies with the value of the target $A$. In particular, the sign of $p_I$ coincides with the sign of the target. The instability region thus gets pushed to positive target values when $A$ is
a.

Figure 10: Recursive least-squares solutions (arbitrary $\rho$): comparison between mean-field predictions and trained networks. (a) Local stability of the target fixed point in feedback networks trained through the RLS algorithm. Details are as in Figure 7a. We change the initial geometry of the network architecture by increasing values of the overlap $\rho$. The target value is fixed to $A = -1.1$. The yellow shaded area indicates the parameter region where the LS solution is unstable. (b) Geometry of the LS solution, characterized in terms of the analytical projections of the readout $n$ onto the three axes $\xi$, $\eta_m$, and $\eta_I$ (see equation 4.5). The top and the bottom rows display results for two different values of $\rho$, indicated in panel a by the black arrows. (c) Geometry of the RLS solution, computed numerically by projecting the readout $n$ onto the three axes $\xi$, $\eta_m$, and $\eta_I$. (d) Fraction of bistable trained networks, compared with the mean-field prediction extracted from the readout weights in panel c. Details are as in Figure 7d. Results are displayed for three different values of the regularization parameter $r$. Dots show results averaged over 15 network realizations. Parameters as in Figure 6.

negative, and vice versa. The approximate mean-field description we derive from the RLS weights (see Figure 10c, top) confirms that such a region never emerges. Consistently, training always results in a locally stable fixed
point. The mean-field description further predicts the existence of a second, locally stable solution at large and positive target values (see Figure 10d, top). In agreement, we find that a large fraction of trained networks displays bistability when the target $A$ is taken within such a window.

As a last step, we examine a network architecture characterized by a smaller value of the overlap $\rho$, taken outside the LS instability window (see Figures 10b–10d, bottom). We observe that for smaller values of $\rho$, the bistability region predicted by the mean-field theory at large target values disappears (see Figure 10d, bottom). Bistability is suppressed because in correspondence of the same target values, the readout component along the nonshared input direction $\eta_I$ is characterized by large relative weight (see Figure 10c, bottom). Consistent with the theory, the fraction of trained networks displaying bistability at large $A$ values significantly decreases (see Figure 10d, bottom).

To conclude, we tested the RLS algorithm on feedback networks characterized by arbitrary geometry, where our simplified theoretical framework predicts the existence of globally stable solutions. We found that the algorithm is systematically able to find readout solutions that generate the stable dynamics suitable to solve the task. Stability is achieved thanks to the flexibility of the online algorithm, which reshapes the geometry of the readout vector until a final optimal configuration is found. In contrast with the LS solution, such an optimal readout includes a strong and structured component along the nonshared input direction $\eta_I$, which suppresses bistability and prevents local instabilities.

6 Conclusion

Predicting activity in trained recurrent networks is a challenging theoretical problem due to the disordered synaptic structure and the large number of units (Doya, 1992; Barak, 2017). Through the recurrent dynamics, global bifurcations can appear as a result of minimal modifications to the synaptic weights and lead to dramatic changes in the network dynamical landscape. Among all the possible architectures, feedback networks represent a class of particularly simple recurrent networks. In these networks, training is restricted to a set of $N$ synaptic weights out of $N^2$, and the task-specific part of the recurrent dynamics is specified by the readout signal alone.

In this work, we examined how feedback networks can be trained to perform a single, stationary input-output association. Such a task is simple enough to be implemented in a purely feedforward setup, where no recurrent feedback exists and the reservoir dynamics is driven solely by the external input pattern. In such a case, which corresponds to setting $\sigma_m = 0$ in equation 2.5, the mean-field theory suggests that a task implementation can be constructed by aligning the readout vector $n$ with the external input $I$. This simple feedforward solution implements the task through a single stable fixed point and thus possesses optimal stability properties.
Throughout the letter, we investigated how more complex solutions, which involve a rank-one recurrent feedback, can be constructed. To this end, we derived an approximate mean-field framework that captures the geometry of the $N$-dimensional readout vector $n$ with respect to the input vectors $m$ and $I$. In this theoretical setup, the random bulk connections are treated as quenched noise, so that correlations between the readout vector and the specific instantiation of the random matrix $\chi$ are neglected. Note that the amplitude of the bulk connections $g$ affects the solutions of our mean-field equations only through minor quantitative differences. As a consequence, the space of solutions that our theory describes corresponds to network models where the global connectivity matrix is in practice rank-one (Mastrogiuseppe & Ostojic, 2018). We used the mean-field theory to show that implementing such a simple fixed-point task in a feedback network can generate output states characterized by nontrivial stability properties. Specifically, we showed that the task can be solved if the readout $n$ picks and amplifies the feedback component of the input, which is parallel to the vector $m$ (see Figure 1c). Similar to Hopfield (1982), this implementation strongly relies on the nonlinearity of the dynamics and produces additional spurious attractors. Alternatively, the task can be solved by aligning the readout vector with the external input $I$ (see Figure 1d). Such a configuration induces a hidden feedforward structure in the feedback network, which generates a single and stable output fixed point. In appendix C, we showed that this class of solutions also performs optimally in feedback networks characterized by a threshold-linear activation function.

The mean-field framework was further used to approximate the more complex, disordered solutions that are obtained by training with standard algorithmic techniques. Specifically, we used the theory to predict the local and global stability of the target fixed point, and thus to understand how network parameters and target values affect training performance. Despite implying very strong assumptions, we found that the mean-field approximation correctly describes trained networks in a surprisingly broad range of parameters. The approximation fails to predict the value of the spurious output states in LS training only when the architecture is strongly degenerate ($\rho \sim 1$) and the value of the random strength $g$ is large. This nontrivial result indicates that the dynamical mechanisms underlying the input-output association in LS and RLS trained networks mimic the way the same computation emerges in a simple rank-one network (Mastrogiuseppe & Ostojic, 2018).

As a result of our theoretical analysis, we found that a common training algorithm, which relies on the least-square inversion of the constraint $\phi(x)^Tn = A$, returns suboptimal readout solutions under certain conditions. This happens because such a constraint, exactly like our system of mean-field equations (see equation 3.9), possibly admits multiple solutions but is blind with respect to the underlying dynamics that control stability. Furthermore, we found that both LS and RLS approaches systematically fail to
converge to a locally stable solution when the geometry of the initial network architecture is degenerate and the two input vectors $\mathbf{m}$ and $\mathbf{I}$ are parallel. Such a failure is explained by the approximate mean-field theory, which suggests that the unique task solution in the form of equation 3.4 is characterized by locally unstable dynamics. The mean-field framework, moreover, suggests that introducing extra directions in the network geometry through the nonshared axes $\eta_m$ and $\eta_I$ expands the space of possible read-out solutions. In Figure 10, we showed that an online training algorithm like FORCE can take advantage of such a broader set of solutions by selecting the appropriate readout vector that generates stable output states. Degenerate architectures consisting of strongly overlapping input vectors are not common in practical applications, where vectors $\mathbf{m}$ and $\mathbf{I}$ are typically generated at random. Note, however, that strong overlaps can be accidentally introduced in the network architecture when the two vectors $\mathbf{m}$ and $\mathbf{I}$ are uniform or generated through random distributions of large mean. In such cases, the overlap direction coincides with the unitary vector $\mathbf{u}$.

Our analysis indicates a possibly important role for the parameter $g$, which scales the amplitude of the random bulk of connectivity. We found that increasing the value of $g$ does not significantly influence the structure of the feedback instabilities, which are mostly controlled by the geometry of the input vectors. Furthermore, large $g$ values have essentially no effect on training performance when the target fixed point can be implemented through uncorrelated, mean-field solutions. On the other hand, large random bulks significantly help the RLS algorithm converge to strong, correlated solutions when uncorrelated solutions cannot satisfy the task. Correlated solutions are likely to play a major role in feedback networks trained to more complex tasks beyond the single input-output association. Understanding the mechanism through which large $g$ values actively support training is thus an open but crucial question. As a hint, classical studies indicate that convergence of the RLS algorithm is guaranteed under the condition of persistent excitation (Bittanti, Bolzern, & Campi, 1990; Kubin, 1988). This condition requires that activity in the bulk, which serves as a set of basis functions, samples a large portion of the $N$-dimensional space spanned by the network population during training. From this perspective, large random bulks contribute to improve excitation in two different directions: they slow the decay of network activity, thus enlarging the training sample, and they increase the amplitude of the activity components that are orthogonal to the input vectors, thus increasing the sampling variance.

Our simplified mean-field description is complementary to the approach proposed in Rivkind and Barak (2017), who studied a similar fixed-point task. The mean-field analysis they developed is exact and takes into account both the nonlinear components of the readout vector and its correlations with the random bulk. Such an approach requires an analytical expression for trained readouts and returns the local stability of the target fixed point. Our analysis is approximate but extends more easily to any readout vector.
obtained through algorithmic training. Our approach furthermore captures dynamics on a global scale and allows us to predict the existence of spurious, bistable fixed points.

The mean-field framework we adopt is flexible enough to be directly extended to network architectures where the input weights are independently generated from non-Gaussian probability distributions or the activation function $\phi(x)$ has a different shape (see appendix C). Significant additional work will be required to extend our results to networks of spiking units, for which different training algorithms have been proposed (Eliasmith & Anderson, 2004; Boerlin, Machens, & Deneve, 2013; Thalmeier, Uhlmann, Kappen, & Memmesheimer, 2016; Kim & Chow, 2018).

The mean-field approach we considered here is directly adapted from Mastrogiuseppe and Ostojic (2018). Although the two studies build on the same theoretical tools, they are motivated by deeply different perspectives. In Mastrogiuseppe and Ostojic (2018), the low-rank part of the network connectivity is designed by hand: the overall understanding of the dynamics that emerges from the mean-field framework is exploited to construct low-rank connectivity structures that stably and efficiently implement a variety of behavioral tasks. Crucially, the tasks are in that case defined as qualitative input-output association rules that fix the overall network behavior but not the exact value of the readout $z$. In this work, in contrast, the mean-field framework serves as a tool for understanding how an extremely simple task is implemented through training in standard network architectures. As in supervised learning applications, the task is specified by a fixed quantitative constraint on the readout value $z$. Consequently, the resulting target fixed point is not guaranteed to be a stable state for the network dynamics. In Mastrogiuseppe and Ostojic (2018), moreover, both connectivity vectors $m$ and $n$ are considered to be plastic. In the network models that are derived, the two vectors serve two different computational roles: vector $n$ selectively amplifies the external input patterns, while vector $m$ defines the output direction to be picked by the readout. In the present framework, the network architecture is instead constrained as in machine learning applications (Jaeger, 2001; Lukosevicius, 2012; Sussillo & Barak, 2013): the input weights $m$ are generated at random and considered to be fixed, while the readout direction is given by the trained vector $n$.

The mean-field analysis adopted in both studies is exact when two specific assumptions are satisfied: the readout vector $n$ obeys a weak $(1/N)$ scaling with the network size, and its entries are statistically uniform and uncorrelated with the random part of the connectivity $\chi$. If these hypotheses hold, the resulting feedback network can display only simple or bistable stationary dynamics (Mastrogiuseppe & Ostojic, 2018), which was suitable in our case for implementing a simple fixed-point task. For a feedback architecture trained to solve a more articulate task, on the other hand, the entries of $n$ need to be structured or fine-tuned to the random bulk. In that case, a mean-field description with the same characteristics would fail to
correctly describe the network dynamics. However, one can hypothesize that as long as the task is reasonably simple, the overall connectivity can still be approximated by the sum of a random noise term and a weak and uncorrelated connectivity matrix, where the rank of the latter is larger than one but still much smaller than the network size $N$. If such a low-dimensional decomposition can be extracted from the trained synaptic matrix, then a similar mean-field approach can be directly applied. Specifically, one can hope to predict dynamics by looking at the relative geometrical arrangement of a restricted number of high-dimensional vectors, which define the external inputs together with the low-rank part of the connectivity. Ongoing work indicates that this approach can be used to successfully extract the relevant dynamical mechanisms from recurrent networks trained on more complex tasks, where the spectrum of possible network implementations is not known a priori.

Appendix A: Linear Stability Analysis of Fixed Points

In this section, we provide details of the stability analysis for the stationary mean-field solutions, which correspond to the fixed points of the original dynamics (see equation 2.3). For every fixed point $x$, stability is evaluated by computing the linear stability matrix and predicting the position of its eigenvalues.

The linear stability matrix reads:

$$S_{ij} = \left( g \chi_{ij} + \frac{m_i n_j}{N} \right) \phi'(x_j). \quad (A.1)$$

As shown in Figure 1b, the eigenspectrum of $S$ consists of two distinct components: a circular compact set of eigenvalues, directly inherited from the random bulk $\chi$, and an isolated real outlier, mostly controlled by the rank-one feedback matrix $mn^T$. The radius of the circular set and the position of the outlier are evaluated by following the analysis in Mastrogiuseppe and Ostojic (2018).

The radius can be computed as in Rajan and Abbott (2006) and Harish and Hansel (2015),

$$r = g\sqrt{\langle [\phi'^2(x_i)] \rangle}, \quad (A.2)$$

where the average is evaluated as an integral over a gaussian distribution of variance $\Delta$ (see equation 3.10). In this work, we focus on fixed-point dynamics, so we fix network parameters and target values to ensure $r < 1$.

When the real outlier is not absorbed within the circular set, its position can be evaluated by computing the linearized dynamics of the average activation $\mu_i = [x_i]$ around the stationary mean-field solutions, corresponding
to network fixed points. We illustrate here the main steps of the calculation; a more detailed account can be found in Mastrogiuseppe and Ostojic (2018).

In equation 3.5, we derived that at the fixed point (which we here indicate by the apex $^0$), the average activation is given by $\mu_i^0 = m_i z_i^0 + I_i$. After a small perturbation, we can write $\mu_i(t) = \mu_i^0 + \mu_1(t)$, where the dynamics of $\mu_1(t)$ is given by

$$\dot{\mu}_1(t) = -\mu_1(t) + m_i z_1(t). \tag{A.3}$$

Note that one can formally write $\mu_1 = m_i \tilde{z}_1$, where $\tilde{z}_1$ is the low-pass filtered version of $z_1$: $(1 + d/dt)\tilde{z}_1 = z_1$. Equation A.3 indicates that the decay timescale of the mean activity is inherited by the decay time constant of $z_1$. An additional equation for the time evolution of $z_1$ thus needs to be derived.

When the network is perturbed, the firing activity $\phi_i = \phi(x_i)$ can be evaluated at the first order: $\phi_i^0 \to \phi_i^0 + \phi_i^1(t) = \phi(x_i^0) + \phi'(x_i^0) x_i^1(t)$. As a consequence, the first-order in $z$ reads

$$z_1(t) = \langle \tilde{n}_i [\phi'(x_i^0) x_i^1(t)] \rangle, \tag{A.4}$$

where $\tilde{n}_i = N n_i$ and $n_i$ is as in equation 3.4. Summing equation A.4 to its time derivative, we get

$$\dot{z}_1(t) = -z_1(t) + (1 + d/dt) \langle \tilde{n}_i [\phi'(x_i^0) x_i^1(t)] \rangle. \tag{A.5}$$

As Mastrogiuseppe and Ostojic (2018) showed, one can first compute the average over the random connectivity, yielding

$$[\phi'(x_i^0) x_i^1] = m_i \tilde{k}_1 [\phi_i^1] + \left( \frac{\Delta_0^1}{2} - \langle \mu_1^1 \mu_i^0 \rangle - \langle \mu_i^1 \rangle \langle \mu_i^0 \rangle \right) [\phi_i''], \tag{A.6}$$

where the gaussian integrals $[\phi_i^1]$ and $[\phi_i'']$ are evaluated using the fixed-point statistics.

As a second step, the average over units is performed, yielding a result in the form

$$\langle \tilde{n}_i [\phi'(x_i^0) x_i^1(t)] \rangle = \tilde{z}_1 a + \Delta_0^1 b. \tag{A.7}$$

A little algebra returns the value of the coefficients $a$ and $b$:

$$a = c(p \sigma_m \rho + p_m \sigma_m \sqrt{1 - \rho^2}) [\langle \phi'(x_i) \rangle].$$
\[ b = \frac{c}{2} \left( p_0 \rho z_0^0 + p_0 \sigma_0 \sqrt{1 - \rho^2} z_0^0 + p_1 \rho + p_1 \sigma_1 \sqrt{1 - \rho^2} \right) \langle [\phi''(x_i)] \rangle. \] (A.8)

The time evolution of \( z_1^1 \) can be finally rewritten as

\[ \dot{z}_1^1(t) = -z_1^1(t) + \left\{ az_1^1 + b \left( 1 + \frac{d}{dt} \right) \Delta_0^1 \right\}, \] (A.9)

so that the time evolution of the perturbed variance must be considered as well. By using equation 2.3, one can easily show that close to the fixed point,

\[ \dot{\Delta}_0^1 = -\Delta_0^1 + \left\{ \mu \frac{\partial G}{\partial \mu} \bigg|_0 + \Delta_0^1 \frac{\partial G}{\partial \Delta_0} \bigg|_0 + z_1 \frac{\partial G}{\partial z} \bigg|_0 \right\}, \] (A.10)

where \( G(\mu, z, \Delta_0) = g^2 \langle [\phi_i^2(t)] \rangle + \sigma_m^2 z^2 + 2\sigma_m z + \sigma_1^2 \). Explicitly computing the derivatives gives

\[ \frac{\partial G}{\partial \mu} \bigg|_0 = 2g^2 \langle [\phi_i \phi'_i] \rangle, \]
\[ \frac{\partial G}{\partial \Delta_0} \bigg|_0 = g^2 \left\{ \langle [\phi_i^2] \rangle + \langle [\phi_i \phi''_i] \rangle \right\}, \]
\[ \frac{\partial G}{\partial z} \bigg|_0 = 2\sigma_m^2 z^0 + 2\sigma_m I. \] (A.11)

Furthermore, as the input vectors and the bulk connectivity have zero mean, \( \mu_i^1(t) = -\mu_i^1(t) \).

We finally obtained that the perturbation timescale is determined by the population-averaged dynamics,

\[ \frac{d}{dt} \begin{pmatrix} \mu_1^1 \\ \Delta_0^1 \\ \kappa_1^1 \end{pmatrix} = - \begin{pmatrix} \mu_1^1 \\ \Delta_0^1 \\ \kappa_1^1 \end{pmatrix} + \mathcal{M} \begin{pmatrix} \mu_1^1 \\ \Delta_0^1 \\ \kappa_1^1 \end{pmatrix}, \] (A.12)

where the evolution matrix \( \mathcal{M} \) is defined as

\[
\mathcal{M} = \\
\begin{pmatrix}
0 & 0 & 0 \\
2g^2 \langle [\phi(x_i)\phi'(x_i)] \rangle & g^2 \left\{ \langle [\phi'(x_i)]^2 \rangle + \langle [\phi(x_i)\phi''(x_i)] \rangle \right\} & 2\sigma_m^2 z^0 + 2\sigma_m I \\
2bg^2 \langle [\phi(x_i)\phi'(x_i)] \rangle & bg^2 \left\{ \langle [\phi'(x_i)]^2 \rangle + \langle [\phi(x_i)\phi''(x_i)] \rangle \right\} & b(2\sigma_m^2 z^0 + 2\sigma_m I + a) \\
\end{pmatrix}.
\] (A.13)
When an outlier eigenvalue is present in the stability eigenspectrum, its position can be finally evaluated as the largest eigenvalue of the reduced stability matrix $\mathcal{M}$.

**Appendix B: Least-Squares Solution: Exact Outlier Prediction**

The study in Rivkind and Barak (2017) addresses the local stability of the full least-squares solution. The position of the outlier eigenvalue $\lambda$ is in that case computed by combining a large network mean-field description with control theory arguments. Following the same calculations, we compute the value of $\lambda$ in the current and more general setup, yielding

$$
\lambda = \frac{g^2}{\Delta_i} \left\{ \langle [\phi'^2(x_i)] \rangle + \langle (m_i A + \sqrt{\Delta_i \omega_i}) [\phi(x_i) \phi'(x_i)] \rangle \right\} + 1
$$

$$
= g^2 \{ \langle [\phi'^2(x_i)] \rangle + \langle (\phi(x_i) \phi(x_i)')' \rangle \}
\times \left( 1 + \frac{A}{\Delta_i} \left[ \sigma_m \rho^2 (\sigma_m A + \sigma_I) + \sigma_m^2 (1 - \rho^2) A \right] \right),
$$

(B.1)

where $\Delta_i = g^2 \langle [\phi^2(x_i)] \rangle$ as in equation 3.5. In Figure 2, this expression is evaluated for the mean-field solution corresponding to the target (light green traces).

**Appendix C: Analysis of Networks with Threshold-Linear Activation**

In this section, we extend our results to feedback networks characterized by a threshold-linear activation function. We set $\phi(x) = [x - T]_+$, where $T$ is the parameter controlling the activation threshold. Such a piecewise-linear activation function is extremely popular in machine learning applications based on feedforward networks (Krizhevsky, Sutskever, & Hinton, 2012). Recent literature focusing on recurrent architectures (Kadmon & Sompolinsky, 2015; Mastrogiuseppe & Ostojic, 2017; Rivkind & Barak, 2017), on the other hand, seems to indicate that threshold-linear functions (more than sigmoids) can impair dynamical stability in simple recurrent models. As an example, we focus on negative threshold values ($T = -0.5$), which seem to guarantee better stability for the fixed-point task we consider (Rivkind & Barak, 2017).

As a first step, as in Figure 1, we select arbitrary values of $(p, p_m, p_l)$ (see equation 3.4) to construct example network implementations. We analyze these networks by solving the resulting mean-field equations to predict the output states and their stability. The mean-field equations illustrated in section 3 have been derived by assuming a generic activation function $\phi(x)$, so they directly generalize to threshold-linear networks. The gaussian integrals in equations 3.9 and A.13 can in this case be evaluated analytically.
Figure 11: Implementing the fixed-point task with a threshold-linear network \([\phi(x) = [x - T]_+, T = -0.5]\): mean-field analysis in three example geometries. (a–c) Mean-field characterization of the network output states for three different readout geometries (see Figure 1). Continuous (resp. dashed) lines correspond to locally stable (resp. unstable) mean-field solutions. Top row: value of the readout signal. Bottom row: position of the outlier eigenvalue in the stability eigenspectra. Details as in Figures 1c–1e. The results of simulations are displayed as dots \((N = 6000, \text{average over 20 network realizations})\). We integrate numerically the dynamics of finite-size networks where the readout vector \(n\) is normalized through equation 3.12. In panels a and c, we observe large finite-size effects due to the outlier stability eigenvalues lying very close to the instability boundary. Parameters as in Figure 1.

(none that integrals involving the high-order derivatives \(\phi''\) and \(\phi'''\) systematically vanish). Results are displayed in Figure 11.

Similar to Figure 1d, Figure 11b indicates that aligning the readout vector \(n\) with the private component of the external input vector \(\eta_I\) returns a unique and stable target fixed point. The position of the outlier eigenvalue in the stability eigenspectrum, which is controlled by \(\phi'''\) (see equation A.8), vanishes in the present case. We conclude that the optimal solution derived in the case of a sigmoidal activation function, which consists of a readout vector aligned with \(\eta_I\), applies to threshold-linear networks as well.
Figures 11a and 11c indicate that readout solutions that contain components along the feedback input vector \( \mathbf{m} \) are characterized by poor stability properties. When the readout is aligned with the nonshared component of \( \mathbf{m} \) (see Figure 11a), two marginally stable fixed points (characterized by an outlier eigenvalue laying exactly on the instability line) are generated. This implementation of the task is characterized by extremely long timescales and large finite-size effects and thus results in large readout errors. When the readout contains components along both input vectors \( \mathbf{m} \) and \( \mathbf{I} \) (see Figure 11c), instead, the two fixed points behave asymmetrically. The positive branch of the mean-field solution, which corresponds to the target fixed point when \( A \) is positive, is formally stable but is characterized by a very large outlier eigenvalue. The negative branch of the solution is instead systematically unstable and generates a very broad instability window at negative target values (gray area in Figure 11c). In both Figures 11a and 11c, large outlier eigenvalues are given by the average of the third-order derivative \( \langle [\phi_i'''(\cdot)] \rangle \) vanishing in equations A.13 and A.8. This term quantifies the effect of saturation and has a stabilizing effect in networks characterized by the sigmoidal activation function (see Figure 1). Note that as already observed in a previous study (Mastrogiuseppe & Ostojic, 2017), threshold-linear networks appear to be characterized by larger finite-size effects than sigmoidal ones.

In a second step, similar to Figure 2, we investigate how threshold-linear networks implement the fixed-point task when they are trained via LS inversion (see Figure 12). We find that mean-field solutions are again characterized by large values of the outlier eigenvalue. When the value of the target is negative, specifically, the target solution is predicted to be unstable from both our approximate and the exact (Rivkind & Barak, 2017) theory. This instability window further appears in both network architectures characterized by small and large values of the overlap \( \rho \) (see Figures 12a and 12b).

In Figure 12, we compare theoretical results with simulations performed in infinite-size networks trained via LS inversion. In contrast to the prediction of the theory, we find that when the value of the overlap \( \rho \) is small, finite-size networks manage to keep the readout error small. The output fixed point, however, displays extremely long timescales. When the value of \( \rho \) is large, the readout value significantly diverges from the target when \( A \) is negative.

To conclude, we briefly investigated how threshold-linear networks solve the fixed-point task. We found that the readout components aligned with the feedback vector \( \mathbf{m} \) do not robustly generate bistability, but give rise to instabilities and dramatically slow dynamics by pushing the value of the outlier eigenvalue close to the instability boundary. Finally, for such a class of networks, it could be interesting to further analyze readout solutions characterized by strongly nonlinear geometries (see section 4.2). We keep this direction open for further studies.
Figure 12: The least-squares solution in a threshold-linear network \( \phi(x) = [x - T]_+, T = -0.5 \): approximate mean-field description. (a, b) Mean-field characterization for three different configurations of the nontrained input vectors \( m \) and \( I \). In panel a, we take \( \rho = 0.6 \), in panel b, \( \rho = 1 \). Continuous and dashed lines indicate the solutions of the simplified mean-field description, where the readout is approximated by a vector in the form of equation 3.4. Details are as in Figure 2. The light green line indicates the exact value of the outlier eigenvalue, which measures the local stability of the target solution (Rivkind & Barak, 2017; details provided in appendix B). The right panel in panel b (top) is a magnified view of the area highlighted in pink in the left panel. Dots display the results of simulations from finite-size networks \( N = 6000 \), averages over 20 network realizations). We integrate numerically the dynamics of finite-size networks where the readout vector \( n \) is given by the LS solution (see equation 4.2). In order to reach different solutions, we initialize the network dynamics in two different initial conditions, centered around \( n \) and \( -n \). Parameters as in Figure 1.

Appendix D: Least-Squares Solution: Nonlinear Corrections

In this section, we consider the least-squares readout \( n \) discussed in section 4 and develop a mean-field description that takes the effect of the nonlinearity \( \phi(x) \) into account (see Figure 3). For simplicity, we consider the scenario where the input vectors \( m \) and \( I \) are parallel \( (\rho = 1) \), so that the \( \xi \) is the only dominant direction to be taken into account \( (p_m = p_I = 0) \). Such a setup is of specific interest, as it corresponds to the case where the...
mean-field approximation deviates more strongly from the results obtained through simulating dynamics in trained networks. The nonlinear mean-field description we derive here indicates which fraction of such deviations can be explained by including nonlinear components in our theoretical description.

The full LS solution is proportional to (see equation 4.2):

$$
\phi(x_{ol}^i) = \phi((\sigma_m A + \sigma_I)\xi_i + \delta_{ol}^i).
$$

We approximate $\delta_{ol}^i$ by white noise of self-consistent amplitude, thus yielding an approximate readout in the form

$$
n_i = c \phi\left((\sigma_m A + \sigma_I)\xi_i + \sqrt{\Delta_i} w_{ol}^i\right),
$$

where $w_{ol}^i$ is a standard gaussian variable. With this choice, the analytical expression for $z$ reads

$$
z = c \int D\xi \int Dw_{ol}^i \phi\left((\sigma_m A + \sigma_I)\xi + \sqrt{\Delta_i} w_{ol}^i\right)$$

$$
\times \int Dw \phi\left((\sigma_m z + \sigma_I)\xi + \sqrt{\Delta_i} w\right),
$$

which can be solved together with the usual expression for the second-order statistics (see equation 3.9). Crucially, the residual input $\sqrt{\Delta_i} w$ is not constrained to match the residual input in the open-loop condition ($\sqrt{\Delta_{ol}^i} w_{ol}^i$), as additional output states, together with the target fixed point, can exist.

As in equation 3.12, we fix the normalization factor $c$ by matching the network output with the target in the open-loop configuration. We thus impose $z = A$, $\Delta_i = \Delta_{ol}^i$, and $w = w_{ol}^i$, which gives

$$
c = \frac{A}{\int D\xi \left[\int Dw_{ol}^i \phi\left((\sigma_m A + \sigma_I)\xi + \sqrt{\Delta_i} w_{ol}^i\right)\right]^2}.
$$

Appendix E: Details on Recursive Least-Squares Algorithm

In this section, we illustrate in detail the recursive least-squares (RLS) algorithm (Sussillo & Abbott, 2009) used in section 5.

The algorithm runs online; that is, weights modifications are alternated with short time windows where the network dynamics is simulated. Specifically, we update the readout weights every $\Delta t = 0.1 \tau$, where $\tau$ is the integration time constant of a single unit, which is here normalized to one.
The time step of integration is instead taken equal to 0.01 $\tau$. As the task is extremely simple, the total training time is taken equal to 150 $\tau$. At the beginning of training, the readout vector $n$ is initialized as a gaussian vector of SD $N^s$ (see Figures 6 and 7). If not differently stated, we fix $s = -0.5$. At every training step, the components of $n$ are updated by following (Sussillo & Abbott, 2009). To begin, the error $e(t) = z(t) - A$ is computed. Then the readout vector $n$ is updated according to

$$n(t) = n(t - \Delta t) - e(t) P(t) \phi(x(t)).$$

The matrix $P$ is a running estimate of the inverse of the rate correlation, which includes a regularization term (Sussillo & Abbott, 2009). At the beginning of training, $P$ is initialized as

$$P(0) = \frac{I}{r},$$

where $I$ is the identity matrix and $r$ is a scalar that we refer to as the regularization parameter. If not differently stated, the value of $r$ is fixed to 0.1. At every training step, $P$ is updated as

$$P(t) = P(t - \Delta t) - \frac{P(t - \Delta t) \phi(x(t)) \phi(x(t))^T P(t - \Delta t)}{1 + \phi(x(t))^T P(t - \Delta t) \phi(x(t))}.$$ 

Acknowledgments

This work was funded by the Programme Emergences of City of Paris, Agence Nationale de la Recherche, grants ANR-16-CE37-0016-01 and ANR-17-ERC2-0005-01, and the program Investissements d’Avenir launched by the French government and implemented by the ANR, with the references ANR-10-LABX-0087 IEC and ANR-11-IDEX-0001-02 PSL University. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

References


Received September 7, 2018; accepted January 17, 2019.