Noname manuscript No.
(will be inserted by the editor)

The dynamics of Recurrent Neural Networks trained for temporal tasks and the eigenvalue spectrum (Supplementary information)

Cecilia Jarne

the date of receipt and acceptance should be inserted later

1 Supplementary data

The simulations used in this work, as well as the code used to generate them, are available in the following repository. It will remain open for anyone to see the content and the results.

https://github.com/katejarne/RRN_dynamics

To search for specific examples, with the different characteristics of dynamics, it is recommended to use this Statistical Summary. For the examples shown trained networks have been labeled. The label includes the name of the task for which it was trained, a number to identify the realization, and the initial condition of the weight distribution. These labels allow you to identify the data in the supplementary information to view the examples and to have the corresponding raw data available. For example, "$XOR\#id10\_Ortho$", shown in Figure **??**, is an example of a network trained for XOR task, with the orthogonal initial condition and the id 10 in the dataset.

The trained networks are saved in hdf5 format and can be opened using "$generate\_figures.py$" to generate all the figures corresponding to one particular realization using the corresponding testing set generator. For example, to test the "and" task, use "$generate\_data\_set\_and.py$"

Al network realizations are in individual folders that contain all training stages.

2 Software details

Simulations were generated using python 3.6.9 Tensorflow version 2.0 and Keras 2.3.1 Following the procedure described previously in [Jarne and Laje(2019)].

The analysis was performed using Scikit learn, Numpy, Scipy Python packages.

Address(es) of author(s) should be given

3 Statistical summary of network dynamics

To better interpret the obtained results, the **Supplementary Table 1** was prepared. It contains the percentage of the simulations that have a certain distribution of eigenvalues. It is separated by initial condition and by task learned.

		Dominant Eigenvalues					
Task	Initial Condition	1 Real (+)	2 or more real (+)	1 real (+) and a complex pair	1 real (+) and 2 complex pairs or more	1 real (−) with or without complex pairs	other other
AND	Orthogonal	0,00	5,88	82,35	5,88	5,88	0,35
	Randon Normal	0,00	12,50	50,00	25,00	12,50	0,00
XOR	Orthogonal	0,00	0,00	26,32	73,68	0,00	1.25
	Randon Normal	0,00	7,14	78,57	0,00	7,14	7,14
OR	Orthogonal	0,00	0,00	77,78	16,67	5,56	0,31
	Randon Normal	6,25	0,00	43,75	6,25	18,75	25,00
Flip Flop	Orthogonal	0,00	0,00	50,00	50,00	0,00	0,00
	Randon Normal	0,00	0,00	50,00	21,43	21,43	7,14

Table 1 Supplementary. Percentage of the simulations that have certain distributions in the eigenvalues of the W^{Rec} matrices.

Example figures have been made from the possible situations that were obtained. Some of them are included in the paper to explain the behavior, additional Figures can be found in the repository. The table contains an id reference number of the corresponding simulation. **Supplementary Table 2** is a summary containing an example for each of the possible situations that have been described in the main text.

		Dominant Eigenvalues: sample cases				
Task	**Initial Condition**	**1 real** (+)	**2 or more real** (+)	**1 real** (+) **and a complex pair**	**1 real** (+) **and 2 complex pairs or more**	**1 real** (−) **with or without complex pairs**
AND	Orthogonal	-	#_and_14_ortho	#_and_15_ortho (Fig 6)	#_and_08_ortho	#_and_10_ortho
	Randon Normal	-	#_and_07_ran (Fig 9)	#_and_15_ran (Fig 5 and Fig 9)	#_and_13_ran	#_and_18_ran
XOR	Orthogonal	-	-	#_xor_10_ortho (Fig 3 and Fig 9)	#_xor_02_ortho	-
	Randon Normal	-	#_xor_10_ran	#_xor_09_ran	-	#_xor_06_ran
OR	Orthogonal	-	-	#_or_01_ortho (Fig 4 and Fig 9)	#_or_18_ortho	#_or_15_ortho
	Randon Normal	#_or_11_ran	-	#_or_01_ran (Fig 9)	#_or_03_ran	#_or_13_ran
Flip Flop	Orthogonal	-	-	#_ff_18_ortho (Fig 9)	#_ff_08_ortho	-
	Randon Normal	-	-	#_ff_05_ran (Fig 7)	#_ff_17_ran	#_ff_10_ran

Table 2 Supplementary. Id of the simulations in the repository corresponding to the sample cases described in the work.

The histogram in Figure 3 shows same information, indicating how most of the simulations have 1 real positive+ a complex pair, or 1 real positive +2 complex pairs or more.

Additional examples that are of interest are included here in this summary. Figure 2 shows the description of the situation corresponding to case of fixed point state for an OR task of the Section 4.3 of the paper.

Regarding the frequency of the oscillations, it can be observed that the frequency increases as the angle of the dominant eigenvalues measured with respect to the abscissa. Some examples are shown in the following comparison. In the first panel, the response to a stimulus with the passivated output of realization "*And_id*#18_*orthogonal*" is observed, then "*And_id*#13_*orthogonal*", and "*And_*#19_*randomn*". Here the frequencies of oscillation are 12 Hz, 26 Hz, and 182 Hz, respectively. (**Fig Sup 1**).

On the other hand, the "*And_id*#18_*randomn*" realization constitutes an example of what happens when negative real eigenvalues are obtained, presented in **Fig Sup 2**.

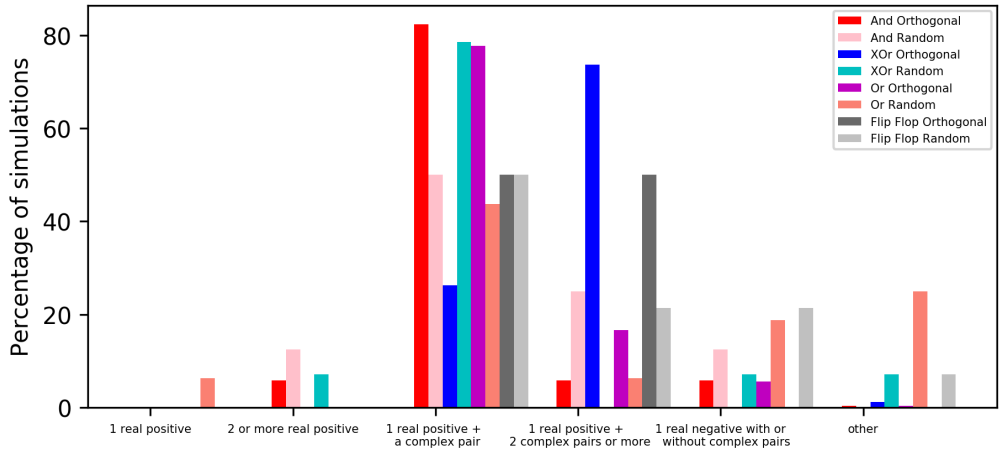

Fig. 1 Supplementary. A visual statistical summary of Supplementary Table 1 the simulations divided by initial condition and task.

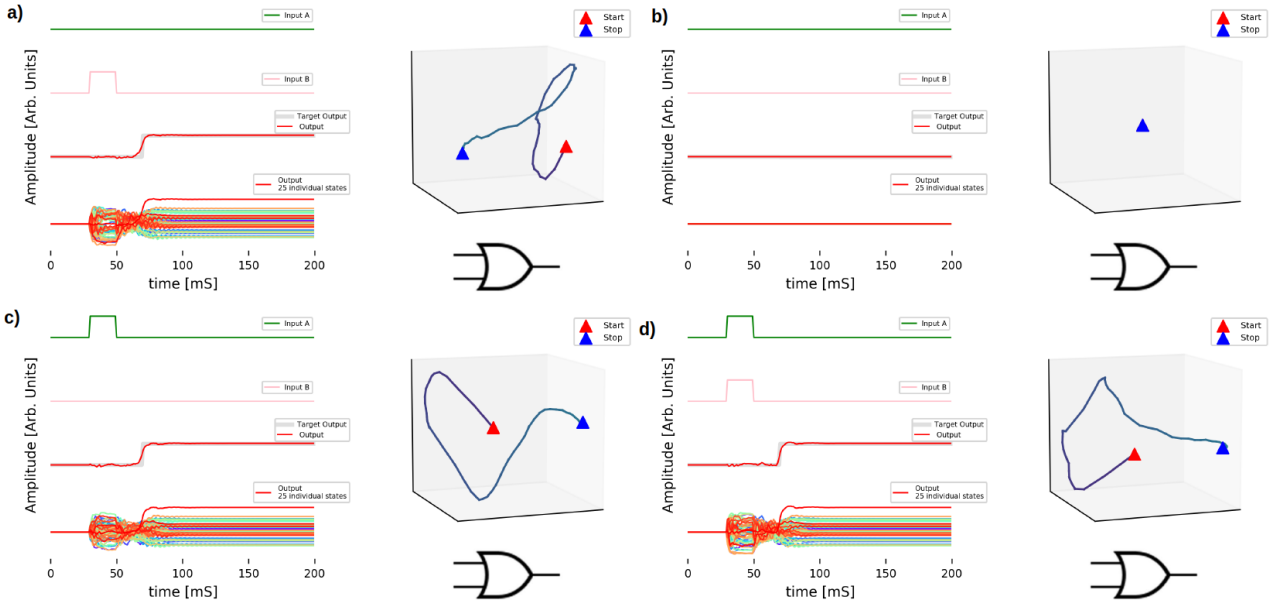

Fig. 2 Excited output states a),c) and d) (fixed-point state) for either input stimulus for the OR function. The case of zero output (b) corresponds only to zero stimuli at the inputs. The network shown corresponds to the simulation with the label: *OR #id*01*_Ortho*.

References

Jarne and Laje(2019). Jarne, C., Laje, R., 2019. Graceful degradation of recurrent neural networks as a function of network size, memory length, and connectivity damage. URL:

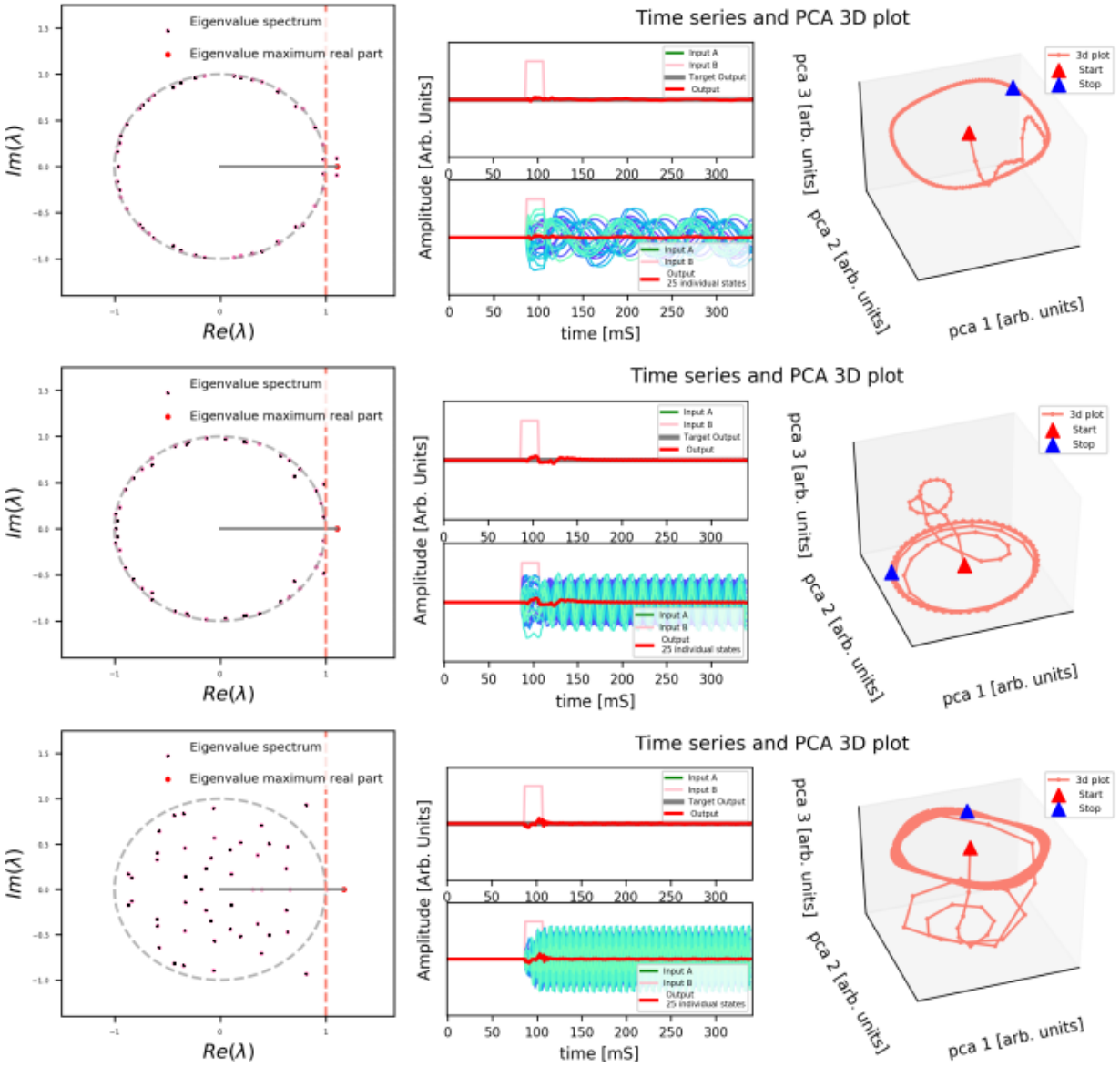

Fig. 3 Supplementary. Trained network for the AND task corresponding to the realization with label *AND# id*05*_orthogonal*, *AND# id*13*_orthogonal* and, *AND# id*19*_random_normal*. Each of the panels in the Figure shows the different positions of eigenvalues outside the unitary circle. The input-output state, and the correspondent dynamics according to the case with dimensionality reduction are shown.

https://arxiv.org/abs/1906.01094, arXiv:1906.01094.

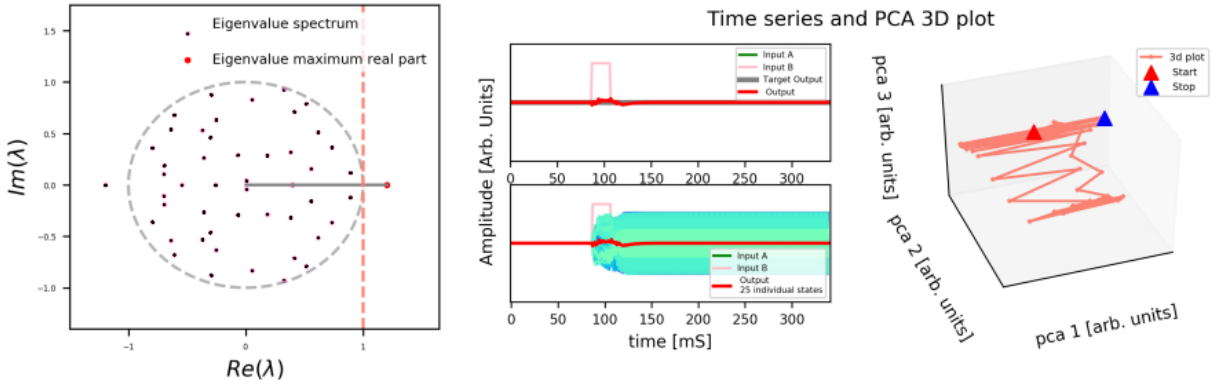

Fig. 4 Supplementary. Trained network for the AND task (corresponding to the realization with label $AND\#\ id18\_randomnormal$). The eigenvalue distribution, the input-output state, and the behavior of the system.