Decouple Adversarial Capacities with Dual-Reservoir Network

Qianli Ma[1,2(✉)], Lifeng Shen[1], Wanqing Zhuang[1], and Jieyu Chen[3]

[1] School of Computer Science and Engineering,
South China University of Technology, Guangzhou 510006, China
qianlima@scut.edu.cn, scuterlifeng@foxmail.com
[2] Guangdong Key Laboratory of Big Data Analysis and Processing,
Guangzhou 510006, China
[3] Linguistic Department, University of California, San Diego, CA 92093, USA
jic387@ucsd.edu

Abstract. Reservoir computing such as Echo State Network (ESN) and Liquid State Machine (LSM) has been successfully applied in dynamical system modeling. However, there is an antagonistic trade-off between the non-linear mapping capacity and the short-term memory capacity in single-reservoir networks, especially when the input signals contain high non-linearity and short-term dependencies. To address this problem, we propose a novel reservoir computing model called Dual-Reservoir Network (DRN), which connects two reservoirs with an unsupervised encoder such as PCA. Specifically, we allow these two adversarial capacities to be decoupled and enhanced in the dual reservoirs respectively. In our experiments, we first verify DRN's feasibility on an extended polynomial system, which allows us to control the nonlinearity and short-term dependencies of data. In addition, we demonstrate the effectiveness of DRN on the synthesis and real-world time series predictions.

Keywords: Reservoir computing · Echo-state network · Short-term memory · Non-linearity mapping · Time series prediction

1 Introduction

Reservoir Computing (RC) [9] is a popular framework of designing and training recurrent neural networks (RNNs) due to its simplicity and effectiveness. A RC network usually consists of three components: an input layer, a dynamic layer called reservoir and an output layer. Weights in the input layer and the reservoir are all fixed randomly, and output weights need to be adapted during training. The reservoir is the core of the whole system as it can provide abundant dynamics with its fixed, random and sparse recurrent connections.

Echo state network (ESN) is a main type of RC networks [7]. Given an ESN with a N-size reservoir, we can define its state equation of ESN as follows:

$$\mathbf{x}(n) = (1-\gamma)\mathbf{x}(n-1)+\gamma f(\mathbf{z}) \tag{1}$$

D. Liu et al. (Eds.): ICONIP 2017, Part V, LNCS 10638, pp. 475–483, 2017.
https://doi.org/10.1007/978-3-319-70139-4_48

where z is named as the working point within activation function and is formulated by

$$\mathbf{z} = \mathbf{W}\mathbf{x}(n-1) + IS \cdot \mathbf{W}_{in}\mathbf{u}(n) \tag{2}$$

where n is the time step, $\mathbf{u}$ is a T-length K-dim input signal, $\mathbf{x}$ is reservoir's echo state. IS denotes the input scaling. γ is a hyperparamer and it denotes the leaky rate of reservoir, and $\mathbf{W}_{in} \in \mathbb{R}^{N\times K}$ is the projection matrix. $\mathbf{W} \in \mathbb{R}^{N\times N}$ is the state transition matrix and is generated by

$$\mathbf{W} = \frac{\rho}{\mathbf{W}_0}\lambda_{max}(\mathbf{W}_0) \tag{3}$$

where $\lambda_{max}(\mathbf{W}_0)$ is the largest eigenvalue of matrix $\mathbf{W}_0$. The elements of $\mathbf{W}_0$ are sampled randomly from [−0.5,0.5]. To ensure the echo state property (necessary stability condition) [6], the spectral radius ρ is suggested to be smaller than one.

From the perspective of dynamic modeling, ESN has two most important capacities: non-linear mapping capacity (NMC) and short-term memory capacity (MC). High NMC means that we can model non-linearity data well, while MC focuses on capturing the short-term dependencies of data. However, as pointed out in previous works [2,12], there is an antagonistic trade-off between NMC and MC in the single-reservoir networks.

In [12], Verstraeten investigated the interplay between these two capacities in reservoir over a simulation task, which allows an accurate control over NMC and MC within data. Their results showed that the overall performance of reservoir system is mostly dominated by memory requirements of a given task. In [2], Butcher analyzed this problem from the effects of the working point $\mathbf{z}$ in Eq. (2). As seen in (2) and (3), $\mathbf{z}$ is affected by the input scaling IS and the spectral radius ρ. As illustrated in Fig. 1, Butcher argued that given a small input signal, an IS smaller than one and ρ close to but smaller than one, the working point will generally be in the linear region of the activation function, which will create a linear reservoir with the highest MC, and the amount of non-linearity will be its minimum. When we enlarge IS and ρ (values over one), the memory capacity

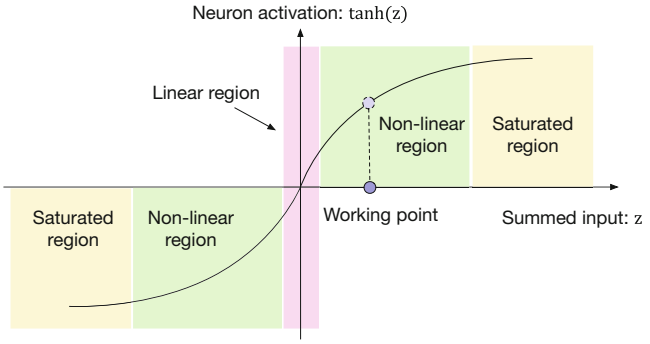


Fig. 1. Illustration of the working point and different regions of the tangent activation function.

of a reservoir will be weakened and its non-linearity will be enhanced as a result of moving working points into non-linear regions. From Butcher's analysis, this trade-off between NMC and MC is mainly due to the unicity of the working point location (linear region or nonlinear one). To resolve this conflict, Butcher introduced two static nonlinear Extreme Learning Machines (ELMs) [5] into an ESN and named the model as R^2SP [2,3]. Another similar work is Gallicchio's φ-ESN [4], which also added an ELM on the top of ESN and enhanced the nonlinearity of the whole system.

In this paper, we propose Dual-Reservoir Network (DRN) to break this antagonistic trade-off between the non-linear mapping capacity and the short term memory capacity. Our main idea is to construct two reservoirs to adjust these two capacities respectively, which is why it is termed as "DUAL". Inspired by the work of φ-ESN, we connect these two reservoirs in a pipeline. However, based on the idea of representation learning in deep learning, we also introduce an unsupervised encoder PCA between the two reservoirs. In this way, we can encoder the state information (with high nonlinearity or short-term dependencies) into an intermediary encoder, and then pass the information from the former reservoir to the latter one. The scales of two capacities in DRN are mainly determined by the hyperparameters IS and ρ and we adopt the genetic algorithm (GA) to optimize them. Finally, several simulations and real-world time series prediction experiments are used to analyze and demonstrate the effectiveness of our proposed DRN.

2 Dual-Reservoir Network

A simple illustration of our proposed Dual-Reservoir Network (DRN) is shown in Fig. 2.

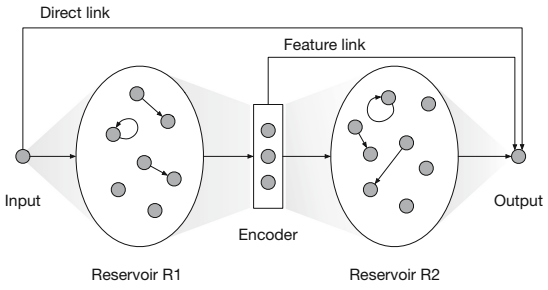


Fig. 2. A simple architecture illustration of the proposed Dual-Rerservoirs Network.

Compared with the general ESN, the defining characteristic of our DRN is in that the hidden layer is replaced by two dual-reservoir (they are denoted by R1 and R2 respectively) and an intermediary unsupervised encoder, where the

encoder is selected as the Principal-Component Analysis (PCA) [13]. The reasons are two folds: (1) PCA is one of the most popular dimension reduction tools and has been used in many data-analysis applications. (2) The reservoir produces a relative high-dimensional state space and in this space, the hidden states in reservoir can be linearly separable with high probability. That also explains why we can use simple regressor to train ESN [7].

2.1 Training Methods

For the convenience of distinguishing parameters and states in the dual reservoirs, we use the superscript (1) to denote reservoir R1, and use superscript (2) to denote reservoir R2. Given T-length K-dim input signals $\{\mathbf{u}(n) \in \mathbb{R}^K\}$, n denotes the time step and $n = 1, 2, \ldots, T$, and let the size of the reservoir R1 and R2 be N_1 and N_2 respectively, the size of encoder be M and the dimension of the output be L, we can formulate the updating equation of the reservoir R1:

$$\mathbf{x}^{(1)}(n) = (1-\gamma^{(1)})\mathbf{x}^{(1)}(n-1)+\gamma^{(1)} f^{(1)}(\mathbf{W}^{(1)}\mathbf{x}^{(1)}(n-1)+IS^{(1)} \cdot \mathbf{W}_{in}^{(1)}\mathbf{u}(n)) \quad (4)$$

where $\mathbf{x}^{(1)} \in \mathbb{R}^{N_1}$ is the echo states of R1. $f^{(1)}$ denotes the activation function and usually is *tanh* function. γ denotes the leaky rate. $\mathbf{W}^{(1)} \in \mathbb{R}^{N_1 \times N_1}$ denotes the transmission matrix, and $\mathbf{W}_{in} \in \mathbb{R}^{N_1 \times K}$ is the projection matrix. At the initial step, $\mathbf{x}^{(R1)}(0)$ is initialized as zeros.

After that, we obtain the echo states of R1 with specific $IS^{(1)}$ and $\rho^{(1)}$, which implies that R1's states have presented some non-linearity and short-term dependencies related with $IS^{(1)}$ and $\rho^{(1)}$. And then, encoding R1's states $\{\mathbf{x}^{(1)}(\mathrm{n})\}$ with PCA, we will get abstract representations $\{\mathbf{h}(\mathrm{n})\}$ from R1's hidden states.

Drive the $\{\mathbf{h}(\mathrm{n})\}$ into the reservoir $R2$, update its echo states and obtain $\{\mathbf{x}^{(2)}(\mathrm{n})\}$ by

$$\mathbf{x}^{(2)}(n) = (1-\gamma^{(2)})\mathbf{x}^{(2)}(n-1)+\gamma^{(2)} f^{(2)}(\mathbf{W}^{(2)}\mathbf{x}^{(2)}(n-1)+IS^{(2)} \cdot \mathbf{W}_{in}^{(2)}\mathbf{h}(n)) \quad (5)$$

where the meanings of the notions are similar to the Eq. (4), but with different inputs $\{\mathbf{h}(\mathrm{n})\}$.

Finally, we introduce direct and feature links from encoders to outputs shown in Fig. 2. Weights of direct link, feature links and output layer all will be collected into a matrix $\mathbf{M}$ and be adapted by regression technique, where $\mathbf{M} \in \mathbb{R}^{N_{collected} \times T}$, and $N_{collected} = K + M + N_2$. Teacher signals are collected into a matrix $\mathbf{T} \in \mathbb{R}^{L \times T}$, and then we have the optimal $\mathbf{W}^{\star}$ formulated by

$$\mathbf{W}^{\star} = \mathbf{T}\mathbf{M}^T(\mathbf{M}\mathbf{M}^T + \beta\mathbf{I})^{-1} \quad (6)$$

which is the well-known ridge regression solution and the β is the *Thikhonov* regularization term.

3 Experiments and Results

In this section, we conduct several experiments to analyze and evaluate the proposed Dual-Reservoir network (DRN). The experiments can be divided into

two parts: (1) firstly, we test the decouple performance of DRN over an extended polynomial dataset with different nonlinearity and short-term dependency levels [2], which allows an accurate control of nonlinearity and short-term dependencies within data. (2) and then, we demonstrate the performance on the tasks of synthesis and real-world time series prediction.

The baseline models we selected here include the leaky ESN [8], R^2SP [2] and φ-ESN [4]. Apart from these baselines, we also introduce two types of encoders in DRN: the elm-based auto-encoder [10] and the random projection (RP) [1]. Hyperparameters of all above reservoir baselines are optimized by the genetic algorithm (GA) and these hyperparameters include IS, ρ and the leaky rate γ. The performance indicator is the normalized root mean squared error (NRMSE), which can be given by

$$NRMSE = \frac{\sqrt{\frac{1}{T}\sum_{t=1}^{T}\left[y(t) - \hat{y}(t)\right]^2}}{Var(\hat{y}(t))} \tag{7}$$

where T is the length of signals, $y(n)$ the outputs at time step n, and $y^{target}(n)$ the corresponding target signals.

As for our implementation details, the size of all the reservoirs of all the baselines is fixed to be 300, and the size of encoders in DRN is given by the cross-validation. The machine setup is the Matlab platform on an Intel Core i5-2410M, 2.30-GHz CPU 8-GB RAM.

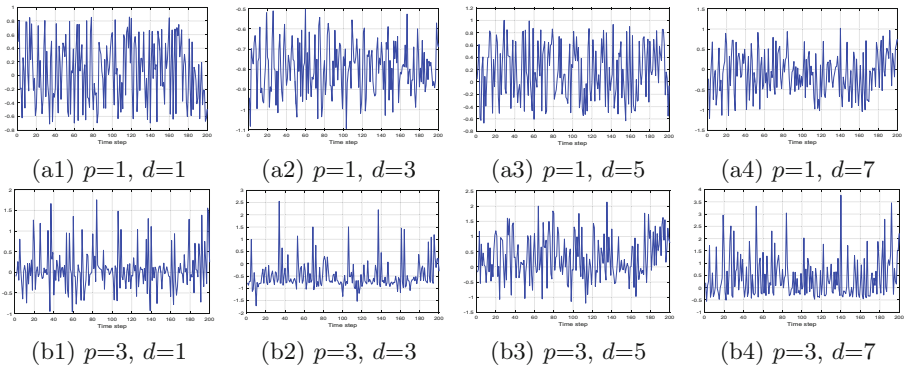
3.1 Extended Polynomial Tasks

This dataset allows an accurate control over the nonlinearity and short-term dependency levels within data [2]. It can be viewed as a time series generated by a specific polynomial system, which can be formulated by

$$y(t) = \sum_{i=0}^{p}\sum_{j=0}^{p-i} c_{ij}u^i(t)u^j(t-d) \qquad s.t.\ i+j \le p \tag{8}$$

where $\{y(t)\}$ denotes the target outputs. $\{u(t)\}$ is the input sequence drawn from an uniform distribution between -1 and +1, c_{ij} is a random number drawn from the same range as $\{u(t)\}$. The terms p and d are two important parameters in this system. The p denotes the order of the polynomial system and the d is the delay. p and d both can be adjusted to produce time series with different characteristics. Larger p means the more non-linear mapping capacity requirements, while larger d requires more short-term memory capacities. These changes are visualized in Fig. 3. In this experiment, we generate 3000 ordered points, and 64% is used for training, 16% for testing and the rest for testing. We do the one-step-ahead prediction over this dataset.

As shown in Table 1, we reported the performance (NRMSE) of our DRN and other baselines over this dataset with different orders p and delays d. From these results, we found that our DRN achieves the best performance among all



(a1) p=1, d=1 (a2) p=1, d=3 (a3) p=1, d=5 (a4) p=1, d=7

(b1) p=3, d=1 (b2) p=3, d=3 (b3) p=3, d=5 (b4) p=3, d=7

Fig. 3. Visualizations of the extended polynomial series with different p and d.

Table 1. Performance (NRMSE) for the extended polynomial tasks. Data with larger p performs more non-linear, and with larger d requires more memory capacity.

NRMSE	MODELS	d=1	d=3	d=5	d=7	d=9
p=1	ESN	1.07E-04	1.45E-03	2.62E-03	5.12E-02	2.10E-03
	Fai-ESN	6.90E-05	1.09E-03	1.41E-02	4.72E-02	8.32E-01
	R2SP	1.02E-04	6.10E-04	1.83E-03	4.54E-04	3.07E-03
	DRN	4.40E-05	2.20E-04	3.10E-04	2.85E-04	1.20E-03
p=3	ESN	5.97E-01	6.07E-01	4.76E-01	4.06E-01	7.58E-01
	Fai-ESN	8.38E-02	5.50E-01	2.93E-01	4.10E-01	7.56E-01
	R2SP	4.92E-01	6.03E-01	4.14E-01	4.21E-01	7.66E-01
	DRN	5.47E-02	5.43E-01	2.48E-01	4.03E-01	6.57E-01
p=5	ESN	6.82E-01	7.02E-01	5.66E-01	9.67E-01	9.89E-01
	Fai-ESN	1.56E-01	3.70E-01	5.16E-01	9.29E-01	9.33E-01
	R2SP	4.22E-01	6.04E-01	5.83E-01	9.62E-01	9.57E-01
	DRN	1.40E-01	3.61E-01	3.76E-01	9.22E-01	9.43E-01
p=7	ESN	4.12E-01	5.62E-01	9.37E-01	1.01E+00	7.09E-01
	Fai-ESN	1.50E-01	5.57E-01	7.53E-01	1.02E+00	8.80E-01
	R2SP	3.32E-01	4.98E-01	8.59E-01	8.84E-01	7.02E-01
	DRN	1.22E-01	4.44E-01	7.41E-01	9.12E-01	6.89E-01

the methods. In particular, we observed that all models tend to be worse when the nonlinearity is higher (increasing p), but our DRN still perform with a high prediction accuracy.

3.2 Time Series Predictions

We select three datasets to evaluate the practical performance of DRN:
(1) **Mackey-Glass System**: This is a classical time series for evaluating the performance of dynamical system identification methods. In discrete time, the Mackey-Glass delay differential equation can be formulated by

$$y(t+1) = y(t) + \delta \cdot (a\frac{y(t-\tau/\delta)}{1+y(t-\tau/\delta)^n} - by(t)) \tag{9}$$

where the parameters δ, a, b, n usually are set to 0.1, 0.2, −0.1, 10. When $\tau > 16.8$, the system becomes chaotic, and in most previous work, τ is set to 17. Thus, we also let τ be 17 in this task. The model details can be found in the literature [7]. In this task, we adopt the 84 time steps ahead prediction to test our methods. In details, we simulate a 10000-length MGS time series, and split these 10000 points into three parts with length $T_{train} = 6400$, $T_{validate} = 1600$ and $T_{test} = 2000$. To avoid the effects of initial states, we discard a certain number of initial steps, $T_{washout} = 100$ for each reservoir.

(2) **NARMA System**: This is a difficult task to test the performance of recurrent networks. To model this dynamical system, high nonlinearity and strong memory capacity are required. The updated equation of 10th-order NARMA can be defined as follows:

$$y(t+1) = 0.3y(t) + 0.05y(t)\sum_{i=0}^{9} y(t-i) + 1.5u(t-9)u(t) + 0.1 \tag{10}$$

where $u(t)$ is a random value drawn from an uniform distribution between [0, 0.5] for each time step t, and the output signal $y(t)$ is initialized by zeros for the first ten steps. The total length we used is 4000 and $T_{train} = 2560$, $T_{validate} = 640$ and $T_{test} = 800$, respectively. The washout length is set to be 30 for each reservoir. In this task, we conduct the one-step-ahead prediction.

(3) **Daily Minimum Temperatures** in Melbourne, Australia is a real world time series dataset, recorded from January 1, 1981 to December 31, 1990 [11]. There are a total of 3650 sample points. We let $T_{train} = 2336$, $T_{validate} = 584$ and $T_{test} = 730$, respectively. The washout length for each reservoir is set to 30. Since this real world time series presents strong nonlinearity, we smooth it with a 5-step sliding window.

Table 2. Average prediction results (NRMSE) with standard deviation.

MODELS	MGS-84	NARMA	Temperatures
ESN	2.01E-01±2.91E-02	2.45E-01±2.00E-02	1.39E-01±1.02E-03
φ-ESN	3.96E-02±7.49E-03	1.69E-01±1.75E-02	1.41E-01±1.10E-03
R^2SP	1.25E-01±1.96E-02	1.81E-01±2.21E-02	1.37E-01±9.82E-04
DRN-RP	3.73E-02±4.59E-03	1.37E-01±1.89E-02	1.36E-01±1.06E-03
DRN-ELMAE	3.41E-02±7.74E-03	1.53E-01±2.14E-02	1.36E-01±8.00E-04
DRN-PCA	3.67E-02±4.78E-03	1.31E-01±9.50E-03	1.35E-01±4.31E-04

All average prediction results (NRMSE) are summarized in Table 2. As can be seen in Table 2, DRNs outperform other baselines significantly. Within the variants with different encoders, we found that the RP works the worst among

three encoder types but still performs better than other baselines. The ELM-AE performs best over the MGS task, and the PCA also performs well. As for the NARMA and temperature tasks, the PCA achieves the best result.

4 Conclusions and Discussions

In this paper, we focus on the adversarial problems between the non-linear mapping capacity (NMC) and the short-term memory capacity (MC) in ESN. To address this problem, we proposed the dual-reservoir network (DRN), which is to allow the dual-reservoirs to adjust these two capacities respectively. Specifically, these two capacities are affected by the hyperparameters input scaling and spectral radius, which we adopt the genetic algorithm (GA) to optimize. In experiments, we used the extended polynomial dataset to verify the effectiveness of our models under different nonlinearity and memory requirements. We also test on the synthesis and real-world prediction tasks. It is worth noting that our DRN not only works in the adversarial problem of NMC and NC, but also can be developed into a hierarchical reservoir-computing framework.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (Grant Nos. 61502174, 61402181), the Natural Science Foundation of Guangdong Province (Grant Nos. S2012010009961, 2015A030313215), the Science and Technology Planning Project of Guangdong Province (Grant No. 2016A040403046), the Guangzhou Science and Technology Planning Project (Grant Nos. 201704030051, 2014J4100006), the Opening Project of Guangdong Province Key Laboratory of Big Data Analysis and Processing (Grant No. 2017014), and the Fundamental Research Funds for the Central Universities (Grant No. D2153950).

References

1. Bingham, E., Mannila, H.: Random projection in dimensionality reduction: applications to image and text data. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 245–250. ACM (2001)
2. Butcher, J., Verstraeten, D., Schrauwen, B., Day, C., Haycock, P.: Reservoir computing and extreme learning machines for non-linear time-series data analysis. Neural Netw. **38**, 76–89 (2013)
3. Butcher, J., Verstraeten, D., Schrauwen, B., Day, C., Haycock, P.: Extending reservoir computing with random static projections: a hybrid between extreme learning and RC. In: 18th European Symposium on Artificial Neural Networks (ESANN 2010), pp. 303–308. D-Side (2010)
4. Gallicchio, C., Micheli, A.: Architectural and markovian factors of echo state networks. Neural Netw. **24**(5), 440–456 (2011)
5. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: theory and applications. Neurocomputing **70**(13), 489–501 (2006). Neural Networks Selected Papers from the 7th Brazilian Symposium on Neural Networks (SBRN 2004)
6. Jaeger, H.: The echo state approach to analysing and training recurrent neural networks-with an erratum note. German National Research Center for Information Technology GMD Technical report, Bonn, Germany, vol. 148(34), p. 13 (2001)

7. Jaeger, H., Haas, H.: Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication. Science **304**(5667), 78–80 (2004)
8. Jaeger, H., Lukoeviius, M., Popovici, D., Siewert, U.: Optimization and applications of echo state networks with leaky-integrator neurons. Neural Netw. **20**(3), 335–352 (2007). Echo State Networks and Liquid State Machines
9. LukošEvičIus, M., Jaeger, H.: Reservoir computing approaches to recurrent neural network training. Comput. Sci. Rev. **3**(3), 127–149 (2009)
10. Tang, J., Deng, C., Huang, G.B.: Extreme learning machine for multilayer perceptron. IEEE Trans. Neural Netw. Learn. Syst. **27**(4), 809–821 (2016)
11. Time Series Data Library: daily minimum temperatures in Melbourne, Australia, 1981–1990. https://datamarket.com/data/set/2324/daily-minimum-temperatures-in-melbourne-australia-1981-1990
12. Verstraeten, D., Dambre, J., Dutoit, X., Schrauwen, B.: Memory versus nonlinearity in reservoirs. In: The 2010 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2010)
13. Wold, S., Esbensen, K., Geladi, P.: Principal component analysis. Chemometr. Intell. Lab. Syst. **2**(1), 37–52 (1987). Proceedings of the Multivariate Statistical Workshop for Geologists and Geochemists