

論文 / 著書情報
Article / Book Information

題目(和文)	適応フィルタにおけるエコーステートネットワークの貯留槽適応と読み出し部の低コストオンライン学習
Title(English)	Reservoir Adaptation and Low Cost Online Readout Training for Echo State Network in Adaptive Filtering
著者(和文)	YUENYONGSUMETH
Author(English)	Sumeth Yuenyong
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第9634号, 授与年月日:2014年9月25日, 学位の種別:課程博士, 審査員:西原 明法,國枝 博昭,山田 功,府川 和彦,杉山 将,田中 聡久
Citation(English)	Degree:., Conferring organization: Tokyo Institute of Technology, Report number:甲第9634号, Conferred date:2014/9/25, Degree Type:Course doctor, Examiner:,,,,,
学位種別(和文)	博士論文
Category(English)	Doctoral Thesis
種別(和文)	論文要旨
Type(English)	Summary

This study deals with the use of a new type of neural network called the Echo States Network (ESN) in nonlinear adaptive filtering. Unlike traditional recurrent neural network (RNN), in ESN only some part of the network is trained and the error surface is unimodal.

However despite this advantage, there are two problems of ESN when used as an adaptive filter: first, most of the structure is randomly initialized and never trained, so it is not optimal for any given task, secondly, the training process encounters extremely high eigenvalue spread. Due to this, online training of ESN requires the use of recursive least squares (RLS) methods instead of the much cheaper least mean squares (LMS) type. Therefore, computational cost becomes an issue since RLS type algorithms have squared complexity.

We address the above two problems. For the first one, we propose an algorithm that adapts the parameters involved in the generating the fixed part such that the lowest possible mean square error (MSE) can be achieved. For the second problem, we propose the use of a different structure for the trained part that leads to much lower computational cost of online training.

There are three parameters involved in generating the reservoir: the input scaling, the connectivity of the reservoir and its spectral radius. Currently, these three parameters must be chosen by manual trial and error. The weights of the reservoir are sampled from a uniform random distribution from -1 to 1, and then scaled with the input scaling and the spectral radius. We call reservoirs generated in this manner "classical reservoirs".

Optimization of reservoir parameter is called "pre-training" since it is conducted before the readout is trained. It is necessary to use derivative-free global optimization algorithm. Such algorithm need to perform a large number of function evaluations, therefore using MSE as the cost function means that the readout needs to be computed repeatedly, which is very expensive computationally. A cost function that does not involve training the readout is needed. Moreover, classical reservoir causes noise in the optimization process due to the fact that different ones are generated from the same parameter set, so we perform pre-training on a deterministic reservoir called CRJ. The proposed cost function is the average of the mutual information (MI) between the states history of *each* neuron in the reservoir and the vector of desired response. The reason behind this cost function is that forming the desired response vector is basically taking the linear combination of the individual neuron state history, therefore, in order for the desired response to be approximated well there should be a high degree of dependency between the states history and the desired response.

In order to approximate the MI's, we use the histogram method which the cheapest but also least accurate. However it is not necessary to predict the individual MI values with high accuracy since it is the average that we are trying to maximize. The estimation errors cancel out when the MI values are averaged. We used 500 samples for MI estimation. The cost function was maximized using the Differential Evolution (DE) algorithm.

We tested pre-training by comparing pre-trained CRJ reservoir against classical reservoirs with the parameters set to: spectral radius equals to 0.75, input scaling equals to 0.5, and connectivity of 10%. These parameters are from the middle of the range considered in Chapter 2. We called this parameter set "standard parameter". We considered the problems of order 10 system identification, time-series prediction, nonlinear-delay mapping, and equalization. On average, pre-training reduce the NMSE by 34%.

The computational time of pre-training when implemented in C++ can be approximated roughly by $t=0.0502N$, where t is time in seconds and N is the reservoir size. For $N=100$, our method takes about 35 second to pre-train in the MATLAB environment, if MSE is used as the cost function, the time needed was over 300 seconds.

Next, it shown experimentally that LMS type algorithm performs poorly. The computational cost can instead be reduced by reducing the number of coefficients that must be adapted. We used a structure called Extreme Learning Machine (ELM) as readouts. When the linear readout is replaced with ELM, the number of coefficient becomes equal to the number of neurons M of ELM.

The reservoir size was fixed at $N=200$. We considered 4 different test problems. On average, $M=30$ is needed for the averaged MSE produced by ELM readouts to be comparable to (no more than 10% in excess of) or even lower than those produced by linear readout. At $N=200$ the time taken for 20 runs by linear readout was 29 seconds, while that of ELM readout was 4.61 seconds. This represents average computation saving of 84.1%.