

Recurrent Neural Network Based Narrowband Channel Prediction

Wei Liu, Lie-Liang Yang and Lajos Hanzo

School of ECS, University of Southampton, SO17 1BJ, UK.

Tel: +44-23-8059 3125, Fax: +44-23-8059 4508

Email: wl03r, lly, lh@ecs.soton.ac.uk; http://www-mobile.ecs.soton.ac.uk

Abstract—In this contribution, the application of fully connected recurrent neural networks (FCRNNs) is investigated in the context of narrowband channel prediction. Three different algorithms, namely the real time recurrent learning (RTRL), the global extended Kalman filter (GEKF) and the decoupled extended Kalman filter (DEKF) are used for training the recurrent neural network (RNN) based channel predictor. Our simulation results show that the GEKF and DEKF training schemes have the potential of converging faster than the RTRL training scheme as well as attaining a better MSE performance.

I. INTRODUCTION

Adaptive modulation is capable of substantially improving the achievable system performance, provided that the relevant channel state information (CSI) can be accurately predicted [1], [2]. In high-rate broad-band wireless systems the carrier frequency has to be high, which results in a high Doppler frequency. For systems using CSI feedback, the outdated information estimated based on the past data may not be sufficiently accurate. Hence an improved accuracy may be achieved with the aid of CSI prediction [3].

In the context of channel prediction [4], [5], amongst others, subspace based algorithms have been investigated. In [4], [5] the complex-valued flat fading process was modeled in the baseband as the summation of a number of sinusoids. Then the ROOT-MUSIC [4] and the modified ESPRIT [5] algorithms were invoked for estimating the frequencies of sinusoids, followed by determining their amplitudes. The complex-valued future fading channel can then be linearly predicted.

Another alternative is to model the channel by a tapped delay line [2], [6], which can be accurately predicted by a linear predictor [3]. However, a specific drawback of the linear predictor is that the channel's correlation coefficients must be estimated from the channel-impaired received data, which degrades the attainable performance of the linear predictor.

As a design alternative, neural networks have also been proposed for the task of channel prediction [7], [8], [9], since they can be trained to learn from the past statistics, which can be exploited for predicting the future. In [7], the multilayer perceptron (MLP) neural network was invoked for predicting the future channel using the ITU-T channel model. By contrast, in [8], [9] a channel predictor based on a hybrid neural network was proposed, which employed Jake's channel model [10] and outperformed the linear Heinonen-Neuvo (H-N) predictor [8].

RNNs [11] constitute a special class of neural networks, which have the capability of generating feedback information from the outputs. Hence RNNs may be viewed as being analogous to infinite impulse response (IIR) filters and have hence found numerous applications in the field of signal processing [11], [12].

In this contribution we investigate the application of RNNs in terms of narrowband complex-valued channel prediction. Specifically, in our

scheme the so-called fully connected RNN (FCRNN) [11], [12] is invoked and three different training algorithms, namely the real time recurrent learning (RTRL) [13], [14], the global extended Kalman filter (GEKF) and the decoupled extended Kalman filter (DEKF) [15], [16] algorithms are investigated.

The outline of the paper is as follows. Section II describes the RNN-based channel predictor invoked for predicting the narrowband channel, while Section III briefly characterises the three different training algorithms used, namely the RTRL, GEKF and DEKF employed in FCRNNs. The simulation results are discussed in Section IV, while our conclusions are offered in Section V.

II. PROBLEM FORMULATION

For a narrowband fading channel, the sampled received signal $r(k)$ is given by

$$r(k) = c(k)b(k) + n(k), \quad (1)$$

where $c(k)$ is obtained by sampling the complex-valued fading channel $c(t)$ at the time instant of $t = kT_b$ and T_b is the data symbol duration, $b(k)$ is the k th transmitted symbol value, while $n(k)$ is a complex-valued discrete AWGN process having a variance of $N_0/2$ per dimension.

As we can see in Fig.1, the received data $r(k)$ of Eq.(1) is fed into a detector for the sake of generating the detected data $\hat{b}(k)$, which is the estimate of the transmitted data $b(k)$. Then a decision-directed channel estimator (DDCE) is invoked for generating the narrowband channel estimate $\tilde{c}(k)$, which can be expressed as

$$\begin{aligned} \tilde{c}(k) &= \frac{r(k)}{\hat{b}(k)} \\ &= c(k) \frac{b(k)}{\hat{b}(k)} + \frac{n(k)}{\hat{b}(k)}. \end{aligned} \quad (2)$$

The delayed output of the DDCE constitutes a $(P \times 1)$ -dimensional vector $\tilde{\mathbf{c}}(k)$, which is expressed as

$$\tilde{\mathbf{c}}(k) = [\tilde{c}(k), \dots, \tilde{c}(k - P + 1)]. \quad (3)$$

Then $\tilde{\mathbf{c}}(k)$ of Eq.(3) is fed into the RNN-based channel predictor in order to generate the complex-valued predicted channel sample $\hat{c}(k + 1)$.

The internal structure of the RNN-based channel predictor is shown in Fig.2. The family of FCRNNs constitutes a specific subclass of RNNs, where every single neuron of the output layer seen in Fig.2 is fed back into the input layer and every neuron of the input layer is connected to every neuron of the output layer in the network [11], [12].

In this section we assume that the reader is familiar with the operational principles of FCRNNs, which have been documented for example in [11], [12]. More specifically, Fig.2 shows an FCRNN, which consists of N activation neurons having P external inputs as well as a

The financial support of the EPSRC, UK and that of the EU under the auspices of the Phoenix and Newcom projects is gratefully acknowledged.

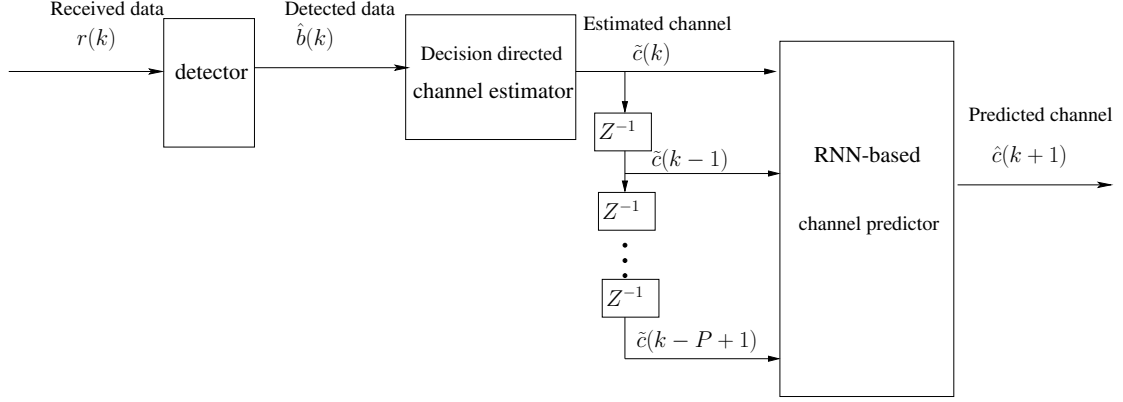


Fig. 1. Schematic of the RNN-based channel predictor

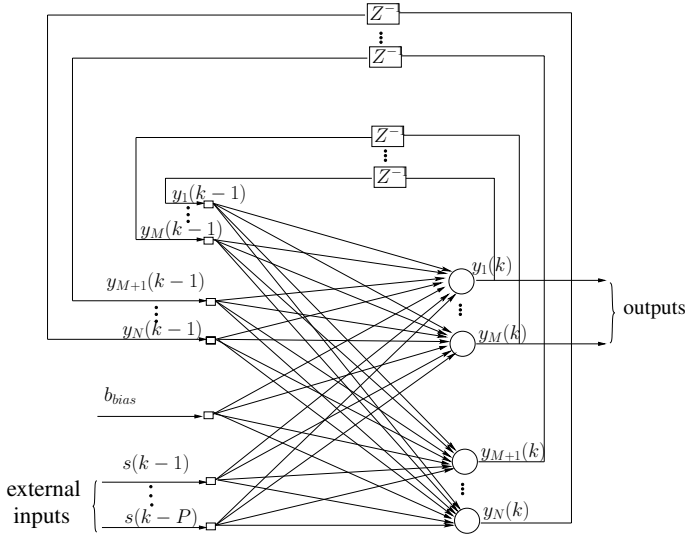


Fig. 2. A fully connected recurrent neural network

fixed bias input b_{bias} . The network has two distinct layers consisting of the external input-feedback layer and a layer of processing elements. Let the $(N \times 1)$ -dimensional vector $\mathbf{y}(k) = [y_1(k), \dots, y_N(k)]^T$ denote the complex-valued output vector of all the neurons at time index k , where $y_n(k)$ is the complex-valued output of the n th neuron, $n = 1, \dots, N$ and the $(P \times 1)$ -dimensional vector $\mathbf{s}(k) = [s(k-1), \dots, s(k-P)]^T$ denotes the complex-valued external input vector at time index k . Furthermore, let the bias be $b_{bias} = (1 + j)$, where $j = \sqrt{-1}$. Then the $(P+1+N) \times 1$ -dimensional combined input vector $\boldsymbol{\rho}(k)$ of the network seen in Fig.2 represents the concatenation of the vectors $\mathbf{s}(k)$, b_{bias} and $\mathbf{y}(k-1)$, which is given by

$$\begin{aligned} \boldsymbol{\rho}(k) &= \begin{bmatrix} \mathbf{s}(k) \\ b_{bias} \\ \mathbf{y}(k-1) \end{bmatrix} \\ &= [s(k-1), \dots, s(k-P), 1 + j, y_1(k-1), \dots, y_N(k-1)]^T \\ &= \boldsymbol{\rho}^{(r)}(k) + j\boldsymbol{\rho}^{(i)}(k), \end{aligned} \quad (4)$$

where the superscripts $(\cdot)^{(r)}$ and $(\cdot)^{(i)}$ denote the real and imaginary parts of the argument, respectively. Assuming that the outputs

of the first M neurons are the outputs of the FCRNN, the $(M \times 1)$ -dimensional output vector of the FCRNN $\mathbf{y}_o(k)$ can be expressed as

$$\mathbf{y}_o(k) = [y_1(k), \dots, y_M(k)]^T, \quad 1 \leq M \leq N. \quad (5)$$

In the context of narrowband channel prediction, we have

$$[s(k-1), \dots, s(k-P)] = [\tilde{c}(k), \dots, \tilde{c}(k-P+1)]. \quad (6)$$

Furthermore, the number of outputs of the FCRNN is $M = 1$. Therefore, in Eq.(5) the output vector $\mathbf{y}_o(k)$ seen in Fig.2 and representing the predicted fading channel is reduced to a scalar $y_o(k)$, which is given by

$$y_o(k) = y_1(k) = \hat{c}(k+1). \quad (7)$$

Let $w_{n,l}$ denote the complex-valued weight of the FCRNN, which connects the n th neuron and the l th input, where we have $1 \leq n \leq N$ and $1 \leq l \leq (P+1+N)$. Then the input of the n th node at time index k is given by

$$\begin{aligned} n_{et_n}(k) &= \sum_{l=1}^{P+1+N} w_{n,l}(k) \rho_l(k) \\ &= n_{et_n}^{(r)}(k) + j n_{et_n}^{(i)}(k), \end{aligned} \quad (8)$$

where $n_{et_n}^{(r)}(k)$ and $n_{et_n}^{(i)}(k)$ are the real and imaginary parts of $n_{et_n}(k)$, respectively, which can be expressed as [13]

$$n_{et_n}^{(r)}(k) = \sum_{l=1}^{P+1+N} [w_{n,l}^{(r)}(k) \rho_l^{(r)}(k) - w_{n,l}^{(i)}(k) \rho_l^{(i)}(k)], \quad (9)$$

$$n_{et_n}^{(i)}(k) = \sum_{l=1}^{P+1+N} [w_{n,l}^{(r)}(k) \rho_l^{(i)}(k) + w_{n,l}^{(i)}(k) \rho_l^{(r)}(k)]. \quad (10)$$

The output of the n th activation neuron can be expressed as [13], [14]

$$\begin{aligned} y_n(k) &= \Phi(n_{et_n}(k)) \\ &= f(n_{et_n}^{(r)}(k)) + j f(n_{et_n}^{(i)}(k)) \\ &= \frac{1}{1 + e^{-n_{et_n}^{(r)}(k)}} + j \frac{1}{1 + e^{-n_{et_n}^{(i)}(k)}}, \end{aligned} \quad (11)$$

where Φ is a complex-valued nonlinear activation function, while $f(\cdot)$ is the real-valued input and real-valued output function, which was chosen to be the logistic sigmoid function expressed as [11]

$$f(x) = \frac{1}{1 + e^{-x}}, \quad (12)$$

where x is real-valued.

Let $d_m(k)$, $1 \leq m \leq M$ be the corresponding desired output of the m th neuron. Then the $(M \times 1)$ -dimensional desired output vector of the FCRNN $\mathbf{d}_o(k)$ can be expressed as

$$\mathbf{d}_o(k) = [d_1(k), \dots, d_M(k)]^T, \quad 1 \leq M \leq N. \quad (13)$$

In the context of narrowband channel prediction, the desired output vector $\mathbf{d}_o(k)$ is reduced to a scalar and physically corresponds to the actual fading channel sample $c(k+1)$, which is given by

$$d_o(k) = d_1(k) = c(k+1). \quad (14)$$

Consequently, the error $e(k)$ seen in Fig.3, represents the discrepancy between the actual fading channel sample $c(k+1)$ and the predicted fading channel $\hat{c}(k+1)$, which is given by

$$e(k) = c(k+1) - \hat{c}(k+1). \quad (15)$$

Then the cost function (CF) of

$$E(k) = \frac{1}{2} |e(k)|^2 \quad (16)$$

is invoked by a training algorithm to generate the updated RNN weights, as shown in Fig.3 until a satisfactorily low mean square error value (MSE) is obtained.

III. TRAINING ALGORITHMS FOR FCRNNs

The RTRL algorithm is the most widely used technique of training RNNs, which was first proposed in [18] for real-valued cases, where all the inputs, outputs, weights and activation functions are assumed to be real-valued. However, in many practical applications the inputs and outputs of a dynamic system are best described as complex-valued signals [17]. In such cases, the real-valued RTRL algorithm has to be extended to the complex-valued RTRL (CRTRL) [13], [14].

Although the RTRL algorithm is popular owing to its reasonable complexity, it is based on the gradient method using first-order derivatives. Hence, it may exhibit an inferior convergence speed in comparison to the more sophisticated learning techniques using second-order derivatives [19].

The extended Kalman filter (EKF) [19] forms the basis of a second-order neural network training method. The essence of the recursive EKF procedure is that an approximate covariance matrix is generated, which encapsulates second-order information about the training problem considered and the elements of the matrix evolve during the training process. Since Singhal and Wu introduced the EKF training algorithm in [20] in the context of static forward neural networks (FNNs), the EKF has constituted the basis of computationally efficient neural network based training techniques that facilitate the application of FNNs and RNNs in diverse problems such as pattern classification [15], [16], control [21], [22], channel equalization [23], [24], [25], [26], etc.

In this contribution, the RTRL [13], [14], the GEKF and DEKF [15], [16] training algorithms are investigated and compared.

| Training algorithm | RTRL | GEKF/DEKF |
|------------------------------------|----------|-----------|
| Number of activations N | 1 | 2 |
| Number of external inputs P | 9 | 9 |
| Number of training data samples | 5000 | 5000 |
| Maximum Doppler frequency f_{dm} | 120 Hz | 120 Hz |
| Data rate f_b | 4kbits/s | 4kbits/s |
| Sampling rate f_s | 4kHz | 4kHz |
| SNR | 10dB | 10dB |
| Learning rate | 0.01 | |

TABLE I
PARAMETERS USED BY THE RTRL, GEKF AND DEKF TRAINING SCHEMES

IV. SIMULATION RESULTS

In the context of narrowband channel prediction, the range of the real-valued logistic sigmoid function of Eq.(12) spans the interval of $(0, 1)$, hence the desired actual channel sample $c(k+1)$ has to be adjusted so that it falls within this range. The adjusted desired actual channel sample $c_s(k+1)$ can be obtained by appropriately shifting and scaling the original desired actual channel $c(k+1)$ sample with the aid of the following equation

$$\begin{aligned} c_s(k+1) &= \frac{c(k+1) + \alpha_s}{\iota_s} \\ &= \frac{(c^{(r)}(k+1) + \alpha_s^{(r)}) + j(c^{(i)}(k+1) + \alpha_s^{(i)})}{\iota_s} \\ &= c_s^{(r)}(k+1) + j c_s^{(i)}(k+1) \quad \iota_s \neq 0, \end{aligned} \quad (17)$$

where α_s is the complex-valued shift coefficient and ι_s is the real-valued scaling coefficient. Furthermore, $\alpha_s^{(r)}$ and $\alpha_s^{(i)}$ are the real and imaginary parts of α_s in Eq.(17), respectively. Specifically, in our simulations, we had $\alpha_s = 5 + j5$ and $\iota_s = 10$.

In this paper, the MSE metric is used for quantifying the attainable performance of the various fading channel predictors. More specifically, the resultant MSE is given by

$$\begin{aligned} \text{MSE} &= \frac{1}{K} \sum_{k=1}^K |c(k+1) - (\hat{c}_s(k+1)\iota_s - \alpha_s)|^2 \\ &= \frac{2}{K} \iota_s^2 \sum_{k=1}^K E(k), \end{aligned} \quad (18)$$

where K is the total number of the channel samples, which were predicted after the training has been completed. Furthermore, in our simulations, BPSK modulation was employed and $E(|c(k)|^2) = 1$ was assumed.

Additionally, the parameters used by the RTRL, GEKF and DEKF training schemes are listed in Table I.

In Fig.4 to Fig.5, the cost function value of Eq.(16) versus the number of training data samples is portrayed for the RTRL and DEKF training schemes, respectively. By comparing Fig.4 to Fig.5 we can observe that the cost function value of Eq.(16) recorded for the DEKF training scheme converges faster to its steady-state value than the RTRL training scheme. Furthermore, the GEKF training scheme results in a similar convergence trend to that of the DEKF training scheme.

In Fig.6 the prediction error between the actual channel sample $c(k+1)$ and the predicted channel sample $\hat{c}(k+1)$ of the three training schemes is characterized with the aid of their Probability Density

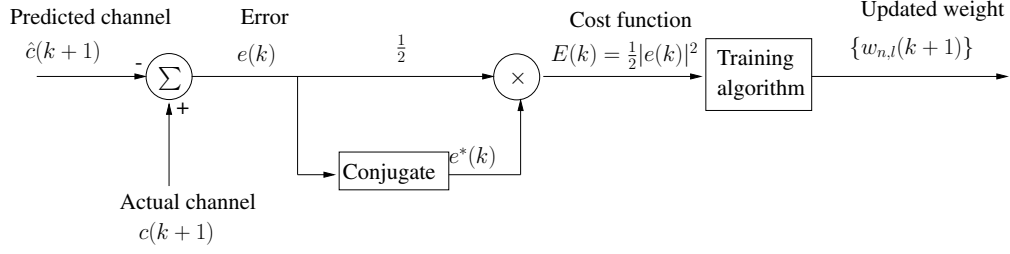


Fig. 3. Schematic of the RNN weight update

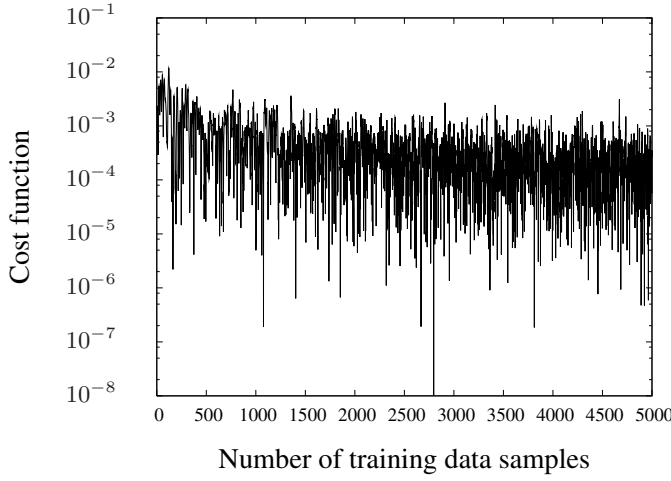


Fig. 4. Cost function value of Eq.(16) versus the number of training data samples for the RTRL training scheme. The remaining parameters were summarized in Table I.

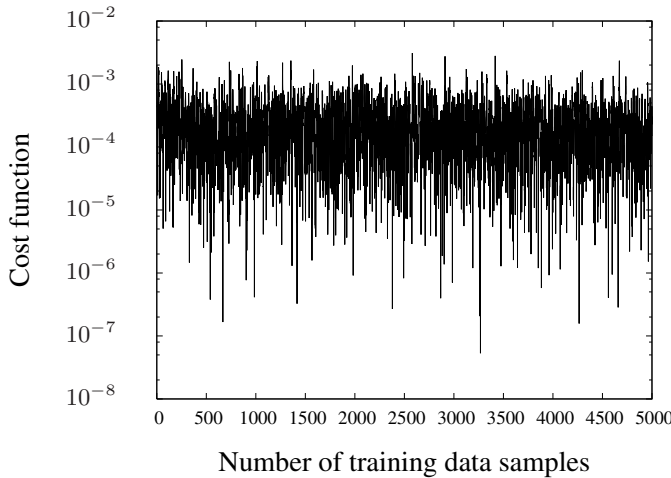


Fig. 5. Cost function value of Eq.(16) versus the number of training data samples for the DEKF training scheme. The remaining parameters were summarized in Table I.

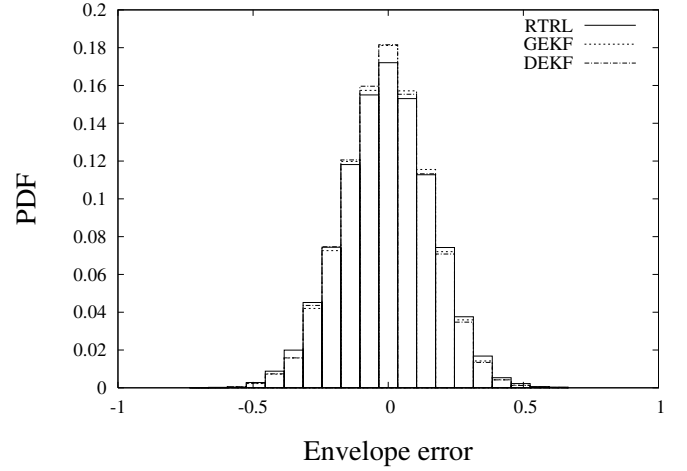


Fig. 6. PDF of envelope of actual channel sample $c(k+1)$ and the predicted channel sample $\hat{c}(k+1)$ by the RTRL, GEKF and DEKF training schemes. The remaining parameters were summarized in Table I.

| Training algorithm | RTRL | GEKF | DEKF |
|--------------------|--------|--------|--------|
| MSE | 0.0602 | 0.0556 | 0.0556 |

TABLE II
MSE PERFORMANCE

Duction (PDF), when using the parameters of Table I. We can see from Fig.6 that the PDF of the GEKF and DEKF training schemes is higher than that of the RTRL training scheme, when the envelope error is small and vice versa. In Fig.7 the envelope of the predicted complex-valued channel samples generated by the DEKF training scheme are compared to the actual channel samples.

Finally, in Table II, the MSE performance attained by the three training schemes using the parameters of Table I is characterized. We can see from Table II that the GEKF and DEKF training schemes achieve identical MSE performance, which are better than that of the RTRL training scheme, but depending on the specific combination of system parameters, these trends may change.

V. CONCLUSIONS

In this paper, the application of FCRNNs was investigated in terms of narrowband channel prediction. Three different algorithms, namely the RTRL, the GEKF and the DEKF were used for training the RNN based predictor. Our simulation results demonstrated that the GEKF and DEKF training schemes converge faster than the RTRL training

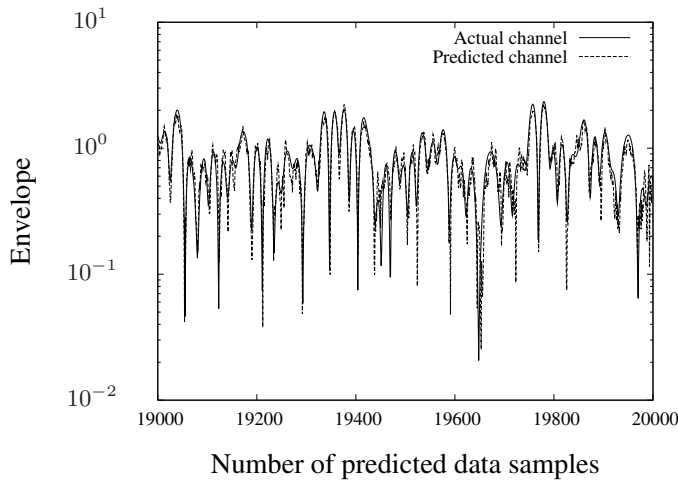


Fig. 7. Magnitude of the complex-valued fading channel envelope predicted by the DEKF training scheme versus the number of predicted data. The remaining parameters were summarized in Table I.

scheme and obtain a better MSE performance. Our future research considers the employment of these techniques in wideband single- and multi-carrier systems.

REFERENCES

- [1] L. Hanzo, C. H. Wong and M. S. Yee "Adaptive wireless transceivers: turbo-coded, turbo-equalized and space-time coded TDMA, CDMA, and OFDM Systems," John Wiley & Sons - IEEE Press, 2002.
- [2] L. Hanzo, M. Münster and B. J. Choi and T. Keller "OFDM and MC-CDMA for broadband multi-user communications, WLANs and broadcasting," John Wiley & Sons - IEEE Press, 2003.
- [3] A. Duel-Hallen, S. Hu and H. Hallen, "Long range prediction of fading signals: enabling adaptive transmission for mobile radio channels," *IEEE Signal Processing Magazine*, vol. 17, pp. 62-75, May 2000.
- [4] J. K. Huang and J. H. Winters, "Sinusoidal modeling and prediction of fast fading processes," *Global Telecommunications Conference*, pp. 892 - 897, November 1998.
- [5] J. B. Andersen, J. Jensen, S. H. Jensen and F. Frederiksen, "Prediction of future fading based on past measurements," *Proceedings of IEEE Vehicular Technology Conference*, pp. 151 - 155, September 1999.
- [6] R. Steele and L. Hanzo "Mobile Radio Communications: Second and Third-Generation Cellular and WATM Systems," John Wiley & Sons - IEEE Press, 1999.
- [7] B. Visweswaran and T. Kiran, "Channel prediction based power control in W-CDMA systems," *First International Conference on 3G Mobile Communication Technologies*, pp. 41 - 45, March 2000.
- [8] X. M. Gao, J. M. A. Tanskanen and S. J. Ovaska, "Comparison of linear and neural network-based power prediction schemes for mobile DS/CDMA systems," *Proceedings of IEEE Vehicular Technology Conference*, pp. 61 - 65, April 1996.
- [9] X. M. Gao, X. Z. Gao, J. M. A. Tanskanen and S. J. Ovaska, "Power prediction in mobile communication systems using an optimal neural-network structure," *IEEE Transactions on Neural Networks*, vol. 8, pp. 1446 - 1455, November 1997.
- [10] W. C. Jakes "Microwave mobile communications," Wiley, 1974.
- [11] S. Haykin, "Neural networks: a comprehensive foundation," Prentice Hall, Inc, 1999.
- [12] D. P. Mandic and J. A. Chambers, "Recurrent neural networks for prediction: learning algorithms, architectures and stability," John Wiley & Sons, 2001.
- [13] G. Kechriotis and E. S. Manolakos, "Training fully recurrent neural networks with complex weights," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 41, pp. 235 - 238, March 1994.
- [14] G. Kechriotis, E. Zervas and E. S. Manolakos, "Using recurrent neural networks for adaptive communication channel equalization," *IEEE Transactions on Neural Networks*, vol. 5, pp. 267 - 278, March 1994.
- [15] G. V. Puskorius and L. A. Feldkamp, "Decoupled extended Kalman filter training of feedforward layered networks," *International Joint Conference on Neural Networks*, pp. 771 - 777, July 1991.
- [16] G. V. Puskorius and L. A. Feldkamp, "A signal processing framework based on dynamic neural networks with application to problems in adaptation, filtering, and classification," *Proceedings of the IEEE*, pp. 2259 - 2277, November 1998.
- [17] N. Benvenuto, M. Marchesi, F. Piazza and A. Uncini, "A Comparison between real and complex valued neural networks in communication applications," *Proceedings of the 1991 Internal Conference on Neural Networks*, pp. 1177 - 1180, June 1991.
- [18] R. J. Williams and D. Zipser, "A Learning Algorithm for Continually Running Fully Recurrent Neural Networks," *Neural Computation*, vol. 1, pp. 270-280, March 1989.
- [19] S. Haykin, "Kalman filtering and neural networks," John Wiley & Sons, 2001.
- [20] S. Singhal and L. Wu, "Training multilayer perceptrons with the extended Kalman algorithm," *Advances in Neural Information Processing Systems I*, pp. 133-140, 1989.
- [21] L. A. Feldkamp and G. V. Puskorius, "Training of robust neural controllers," *IEEE Conference on Decision and Control*, pp. 2754 - 2759, December 1994.
- [22] G. V. Puskorius and L. A. Feldkamp, "Neurocontrol of nonlinear dynamical systems with Kalman filter trained recurrent networks," *IEEE Transactions on Neural Networks*, vol. 5, pp. 279 - 297, March 1994.
- [23] J. Choi, A. C. Lima and S. Haykin, "Kalman filter-trained recurrent neural equalizers for time-varying channels," *IEEE Transactions on Communications*, vol. 53, pp. 472-480, March 2005.
- [24] J. Choi, M. Bouchard and T. H. Yeap, "Decision feedback recurrent neural equalization with fast convergence rate," *IEEE Transactions on Neural Networks*, vol. 16, pp. 699 - 708, May 2005.
- [25] P. H. G. Coelho, "A complex EKF-RTRL neural network," *International Joint Conference on Neural Networks*, pp. 120 - 125, July 2001.
- [26] P. H. G. Coelho, "Adaptive channel equalization using EKF-CRTRL neural networks," *International Joint Conference on Neural Networks*, pp. 1195 - 1199, May 2002.