PARTIAL-UPDATE ADAPTIVE DECISION-FEEDBACK EQUALIZATION

Reza Arablouei[*], *Kutluyil Doğançay*[†*], *and Sylvie Perreau*[*]

[*]Institute for Telecommunications Research, University of South Australia
[†]School of Electrical and Information Engineering, University of South Australia
Mawson Lakes 5095, Australia
phone: + (61) 8 830 23316, fax: + (61) 8 830 23873, email: arary003@mymail.unisa.edu.au

ABSTRACT

In this paper, we consider partial updating of decision-feedback equalizers (DFEs). Application of data-dependent partial-update techniques to DFE is complicated by the fact that the feedback filter regressor has entries with discrete amplitudes, making magnitude-based sorting impossible. We present a novel selective-partial-update algorithm for DFEs that treats feedforward and feedback filters separately and utilizes past decision-directed errors for feedback filter coefficient selection. In addition, a new partial-update scheme is proposed that switches between selective partial updates and periodic partial updates to make effective use of training periods in telecommunication applications. Simulation results are presented to corroborate the effectiveness of the new partial-update DFEs in dealing with realistic frequency-selective and time-varying channels at reduced complexity.

1. INTRODUCTION

Ever-growing transmission rates of communication systems have led to an increased interest in decision-feedback equalizers (DFEs). The main advantage of DFE over linear equalizers is its capability to cancel inter-symbol interference (ISI) with reduced noise enhancement, resulting in considerably lower symbol error rate (SER) than that of a linear equalizer [1]. The underlying principle of DFE is to eliminate future ISI contributions of detected symbols [2].

In practice, the receiver is not initially aware of the dynamics of the channel and the channel can vary in time. A common practical way to enable tracking time variations of the channel by an equalizer is to implement it as an adaptive filter [3]. Adaptive DFE is well known for its relatively simple structure. However, in scenarios where the channel is highly dispersive or multiple antennas are utilized, DFE's complexity may become prohibitive. Fractionally-spaced equalization can make it even more expensive. Given that the equalizer/detector block is the computationally most demanding part of a communication system, it is important to have a reliable and low-complexity equalization/detection method with a sensible tradeoff between performance and complexity. Partial updating offers an attractive solution here. Although partial-update techniques have proven their capability in several linear adaptive filtering applications e.g. audio/network echo

This work was supported in part by a *Commonwealth Scientific and Industrial Research Organisation (CSIRO)* scholarship.

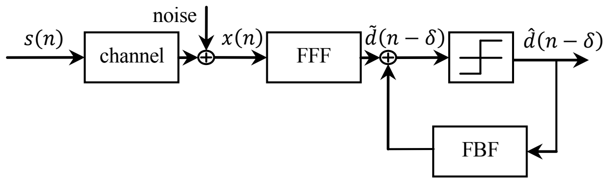

Fig. 1, Block diagram of a decision-feedback equalizer.

cancellation, their application to DFE has not yet been explored. As distinct from *linear* finite impulse response (FIR) adaptive filters, for which the existing partial-update techniques have been originally derived, DFE is a *nonlinear* adaptive filter. DFE comprises two FIR adaptive filters, viz. feedforward and feedback filters. These filters have input signals with different statistics and covariance matrices, implying that for partial updating, the feedforward and feedback filters must be treated separately rather than as one coefficient vector. Hence, some caveats should be taken into account when designing a partial-coefficient-update technique for DFE, especially for data-dependent partial updating which uses input regressor for coefficient selection.

The paper is organized as follows. In Section 2, we review the NLMS algorithm for DFE. In Section 3 we derive a generic partial-update NLMS-DFE using an instantaneous approximation of Newton's method. We develop a new selective-partial-update NLMS-DFE based on the principle of minimum disturbance in Section 4. We introduce a new combined selective-periodic partial-update NLMS-DFE in Section 5 with the objective of making effective use of limited training data. Computational complexity of all the algorithms considered is analysed in Section 6. We provide simulation results in Section 7 and draw conclusions in Section 8.

2. NLMS-DFE

The coefficients of DFE shown in Fig. 1 can be updated according to the NLMS algorithm [3] via

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mu}{\epsilon + \|\mathbf{y}(n)\|^2} e^*(n)\mathbf{y}(n) \tag{1}$$

where

$$\mathbf{w}(n) = \begin{bmatrix} \mathbf{f}(n) \\ \mathbf{b}(n) \end{bmatrix},$$

$$\mathbf{y}(n) = \begin{bmatrix} \mathbf{x}(n) \\ \hat{\mathbf{d}}(n-\delta-1) \end{bmatrix},$$

$$\tilde{d}(n-\delta) = \mathbf{w}^*(n)\mathbf{y}(n),$$

and

$$e(n) = \hat{d}(n-\delta) - \tilde{d}(n-\delta).$$

Here, $\mathbf{f}(n)$ and $\mathbf{x}(n) = [x(n), x(n-1), \cdots, x(n-L_f+1)]^T$ are $L_f \times 1$ vectors denoting feedforward filter (FFF) coefficients and received signal regressor vector (FFF input) while $\mathbf{b}(n)$ and $\hat{\mathbf{d}}(n-\delta-1) = [\hat{d}(n-\delta-1), \hat{d}(n-\delta-2), \cdots, \hat{d}(n-\delta-L_b)]^T$ are $L_b \times 1$ vectors representing feedback filter (FBF) coefficients and equalizer output regressor vector (FBF input), respectively. Moreover, $\tilde{d}(n-\delta)$ and $\hat{d}(n-\delta)$ are hard decision device input and output at time index n, whereas L_f, L_b, μ, ϵ, δ, $\|\cdot\|^2$, $(\cdot)^T$, and $(\cdot)^*$ stand for FFF and FBF temporal spans, adaptation step size, regularization parameter, decision delay, Euclidean norm, transposition, and complex-conjugate transposition, respectively. It should be noted that in the training mode, $\hat{d}(n-\delta)$ is replaced by the known-for-the-receiver transmitted symbol, $s(n-\delta)$.

3. PARTIAL-UPDATE NLMS-DFE

To derive a generic partial-update NLMS algorithm for DFE, we apply an instantaneous approximation to Newton's method [3]. Newton's method is a fast-converging iterative estimation method and its regularized version is given by

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(\epsilon\mathbf{I} + \mathbf{R})^{-1}(\mathbf{p} - \mathbf{R}\mathbf{w}(n)) \tag{2}$$

where $\mathbf{R} = E[\mathbf{y}(n)\mathbf{y}^*(n)]$ is the input autocorrelation matrix, $\mathbf{p} = E[\mathbf{y}(n)\hat{d}^*(n-\delta)]$ is the cross-correlation vector between the input of the equalizer filters and the desired equalizer output and $\mathbf{I}$ is an $(L_f + L_b) \times (L_f + L_b)$ identity matrix. Partial updating of the adaptive filter coefficients replaces $\mathbf{R}$ with $\mathbf{R}_M = E[\mathbf{I}_M(n)\mathbf{y}(n)\mathbf{y}^*(n)]$ and $\mathbf{p}$ with $\mathbf{p}_M = E[\mathbf{I}_M(n)\mathbf{y}(n)\hat{d}^*(n-\delta)]$ where $\mathbf{I}_M(n)$ is an $(L_f + L_b) \times (L_f + L_b)$ diagonal matrix with zeros and/or ones as diagonal entries selecting M coefficients out of $L_f + L_b$ for update at each iteration and $E[\cdot]$ is the expectation operator. Consequently, partial-update version of (2) becomes

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(\epsilon\mathbf{I} + \mathbf{R}_M)^{-1}(\mathbf{p}_M - \mathbf{R}_M\mathbf{w}(n)). \tag{3}$$

The instantaneous approximation of (3) is obtained by simply stripping the expectation operator off the correlation matrix and cross-correlation vector:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\left(\epsilon\mathbf{I} + \mathbf{I}_M(n)\mathbf{y}(n)\mathbf{y}^*(n)\right)^{-1} e^*(n)\mathbf{I}_M(n)\mathbf{y}(n). \tag{4}$$

Applying the matrix inversion lemma [4], we get

$$\begin{aligned}
&\left(\epsilon\mathbf{I} + \mathbf{I}_M(n)\mathbf{y}(n)\mathbf{y}^*(n)\right)^{-1}\mathbf{I}_M(n)\mathbf{y}(n) \\
&= \frac{1}{\epsilon}\mathbf{I}_M(n)\mathbf{y}(n)\left(1 - \frac{\mathbf{y}^*(n)\mathbf{I}_M(n)\mathbf{y}(n)}{\epsilon + \mathbf{y}^*(n)\mathbf{I}_M(n)\mathbf{y}(n)}\right) \\
&= \frac{\mathbf{I}_M(n)\mathbf{y}(n)}{\epsilon + \mathbf{y}^*(n)\mathbf{I}_M(n)\mathbf{y}(n)}.
\end{aligned} \tag{5}$$

Substituting (5) into (4) yields the coefficient update equation for a partial-update NLMS-DFE:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mu}{\epsilon + \|\mathbf{I}_M(n)\mathbf{y}(n)\|^2} e^*(n)\mathbf{I}_M(n)\mathbf{y}(n). \tag{6}$$

Several schemes have been introduced to control the computational complexity of adaptive filters by way of partial coefficient updates. Data-independent approaches are based on the primary idea of selecting coefficient subsets for update in a round-robin fashion (sequential partial updates [5]) or updating all the coefficients in a period greater than basic time index (periodic partial updates [6]). These partial-update adaptive filters may become unstable when the input signal is cyclostationary or periodic [7]. The stochastic partial updates method [8] offers a solution to such stability problems, where subsets of coefficients are randomly selected for update at each iteration.

For sequential or stochastic partial updates, M diagonal elements of $\mathbf{I}_M(n)$ are set to 1 and the rest are set to 0 at each iteration in a sequential or stochastic fashion. For periodic partial updates, $\mathbf{I}_M(n)$ is set to identity matrix once at each update period and is set to a zero matrix at other time instances.

Data-dependent partial-update methods use the regressor vector entries to select equalizer coefficients to be updated. Since the regressor vectors $\mathbf{x}(n)$ and $\hat{\mathbf{d}}(n-\delta-1)$ do not have the same statistics, this may result in $\mathbf{I}_M(n)$ favouring the FFF or FBF with larger input signal variance, leading to undesirable performance degradation as a result of both filters not having equal probability of receiving updates. In order to give a fair chance to all the coefficients (in FFF and FBF) for being updated, we define $M = M_f + M_b$ where M_f and M_b are the number of coefficients to be updated at each time instant from FFF and FBF, respectively. Accordingly, the matrix $\mathbf{I}_M(n)$ is redefined as

$$\mathbf{I}_M(n) = \begin{bmatrix} \mathbf{I}_{M_f}(n) & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{M_b}(n) \end{bmatrix} \tag{7}$$

where $\mathbf{I}_{M_f}(n)$ and $\mathbf{I}_{M_b}(n)$ are $L_f \times L_f$ and $L_b \times L_b$ diagonal coefficient selection matrices to select M_f out of L_f and M_b out of L_b coefficients from FFF and FBF, respectively.

4. SELECTIVE-PARTIAL-UPDATE NLMS-DFE

Data-independent approaches reduce the convergence rate, often proportional to the size of coefficient subsets in sequential or stochastic partial updates and the update frequency in periodic partial updates. On the other hand, data-dependent partial-update techniques offer better convergence performance, where *M*-max updates [9] and selective partial updates [10] are the most prominent ones.

In selective-partial-update NLMS algorithm [10], the coefficients to be updated at each time instant are determined according to the principle of minimum disturbance [3] by solving

Table 1, Required number of arithmetic operations and comparisons at each iteration by different equalizers.

	Multiplications	Additions	Divisions	Comparisons
Full-update NLMS-DFE	$2L+5$	$2L+4$	1	-
SeqPU or StoPU NLMS-DFE	$L+M+5$	$L+M+4$	1	-
PerPU NLMS-DFE	$(1+1/p)L+4+1/p$	$(1+1/p)L+3+1/p$	$1/p$	-
SPU-NLMS-DFE	$L+M+5$	$L+M+4$	1	$2(\log_2 L_f+\log_2 L_b)+4$
SPPU-NLMS-DFE	$\left(1+\frac{4}{5p}\right)L+\frac{M}{5}+\frac{21}{5}+\frac{4}{5p}$	$\left(1+\frac{4}{5p}\right)L+\frac{M}{5}+\frac{16}{5}+\frac{4}{5p}$	$\frac{1}{5}+\frac{4}{5p}$	$\frac{2}{5}(\log_2 L_f+\log_2 L_b)+\frac{4}{5}$

$$\min_{\mathbf{I}_M(n)} \|\mathbf{w}(n+1)-\mathbf{w}(n)\|^2 \quad (8)$$

which, using (6), can be rewritten as

$$\min_{\mathbf{I}_M(n)} \left\| \frac{\mu}{\epsilon+\|\mathbf{I}_M(n)\mathbf{y}(n)\|^2} e^*(n)\mathbf{I}_M(n)\mathbf{y}(n) \right\|^2. \quad (9)$$

Excluding the trivial solution of zero inputs, (9) is simplified to

$$\max_{\mathbf{I}_M(n)} \|\mathbf{I}_M(n)\mathbf{y}(n)\|^2 \quad (10)$$

which means coefficients corresponding to the elements of the input regressor vector with the largest magnitudes should be selected for update at each time index [7], [10]. Using this criterion, the coefficient selection matrices become

$$\mathbf{I}_{M_f}(n) = \begin{bmatrix} i_1(n) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & i_{L_f}(n) \end{bmatrix} \quad (11)$$

$$i_k(n) = \begin{cases} 1 & \text{if } |x(n-k+1)| \in \left\{M_f \text{ maxima of } |x(n-l+1)|_{1\le l\le L_f}\right\} \\ 0 & \text{otherwise} \end{cases}$$

and

$$\mathbf{I}_{M_b}(n) = \begin{bmatrix} j_1(n) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & j_{L_b}(n) \end{bmatrix} \quad (12)$$

$$j_k(n) = \begin{cases} 1 & \text{if } |\hat{d}(n-\delta-k)| \in \left\{M_b \text{ maxima of } |\hat{d}(n-\delta-l)|_{1\le l\le L_b}\right\}. \\ 0 & \text{otherwise} \end{cases}$$

However, the selection criterion of (12) suggests ranking the elements of $\hat{\mathbf{d}}(n-\delta-1)$ according to their magnitudes. Clearly, the magnitudes of these elements, which are hard-decided symbols drawn from a finite-size constellation set, do not carry any information about the significance of updating their corresponding taps in the FBF. Specifically, if the employed modulation is M-ary PSK (e.g., BPSK or QPSK), all decided symbols are of the same magnitude. If one has to make a choice as to which M_b out of L_b coefficients to be updated in the FBF, the significance of the individual coefficients needs to be made explicit. The hard decisions $\hat{d}(n-\delta-i)_{i=1,\ldots,L_b}$ do not have this information. However, for each $\hat{d}(n-\delta-i)_{1\le i\le L_b}$ the equalizer output error $e(n-i)_{1\le i\le L_b}$ is readily available. The selective-partial-update method requires selection of coefficients with the largest contribution to error reduction, which is also shared by the *M*-max technique for LMS and NLMS [9]. This amounts to identifying coefficients with the largest regressor entries. Therefore for the FBF, the identification of coefficients must be based on the equalization errors that the FBF tap inputs have produced. In other words, past decisions with higher detection errors have larger share of the residual ISI of the FFF output. Therefore, the FBF taps corresponding to these symbols are more in need of correction and updating them provides faster convergence for partial updating. This approach leads to the following coefficient selection criterion for the FBF, which guarantees that the best M_b coefficients are updated in the selective-partial-update sense:

$$\max_{\mathbf{I}_{M_b}(n)} \left\|\mathbf{I}_{M_b}(n)\mathbf{e}(n-1)\right\|^2 \quad (13)$$

where $\mathbf{e}(n-1) = [e(n-1), e(n-2), \cdots, e(n-L_b)]^T$. Hence, the selective-partial-update technique is customized for decision-feedback equalization by replacing the selection criterion of (12) with

$$j_k(n) = \begin{cases} 1 & \text{if } |e(n-k)| \in \left\{M_b \text{ maxima of } |e(n-l)|_{1\le l\le L_b}\right\}. \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

5. COMBINED SELECTIVE-PERIODIC PARTIAL-UPDATE NLMS-DFE

Periodic partial updating [7] is basically a downsampled version of the full-update filter exhibiting the same convergence as full-update, only slowed down by the update period. Thus, it should achieve the same steady-state mean-square error (MSE) as the full-update filter if the same step-size is used in both. In addition, it is the cheapest partial-update technique since, unlike the other techniques, it does not require calculation of the adaptation error and normalization of the step size at each iteration and performs these operations only once in each update period. On the other hand, the method of selective partial updates is the fastest converging partial-update technique. In order to exploit advantages of both these techniques, we propose a new combined partial-update NLMS-DFE, which employs selective partial updates of Section 4 during the training mode to ensure fast convergence to the steady-state MSE and switches to periodic partial updating at the end of training using the decision-directed mode. We call this new scheme *selective-periodic partial updates* (SPPU). It imposes less computational complexity in average compared to SPU and makes best use of available training data in partial-update context.

Table 2, Number of required FLOPs at each iteration and provided savings by different equalizers when $L_f = L_b$, $L = 30$, $p = 2, 4$ and $M = L/p$.

	FLOPs		Saving (%)	
	$p = 2$	$p = 4$	$p = 2$	$p = 4$
Full-update NLMS-DFE	524	524	–	–
SeqPU or StoPU NLMS-DFE	404	344	22.9	34.4
PerPU NLMS-DFE	397	333	24.2	36.4
SPU-NLMS-DFE	443	383	15.4	26.9
SPPU-NLMS-DFE	406	343	22.5	34.5

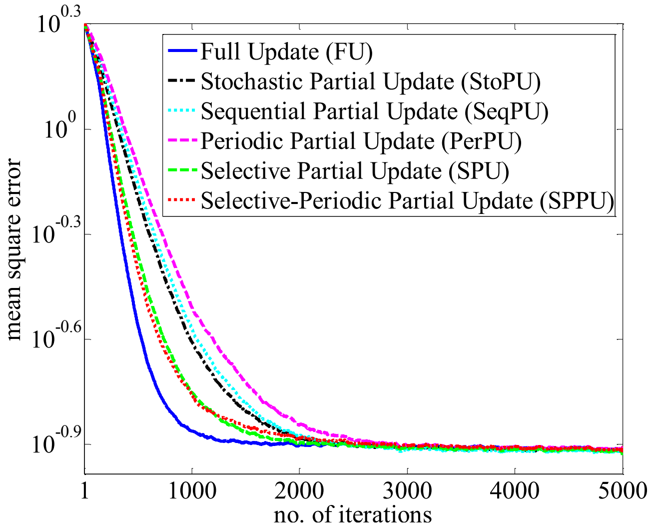

Fig. 2, MSE curves of full-update and different partial-update NLMS-DFEs for a static channel.

6. COMPLEXITY COMPARISON

The number of required arithmetic operations and comparisons at each iteration for equalizers employing full-update and different partial-update NLMS algorithms is presented in Table 1 where $L = L_f + L_b$ and p stands for update period in periodic partial-update and SSPU algorithms. Assuming six floating-point operations (FLOPs) for each multiplication or division and two floating-point operations for each addition, subtraction or comparison [11], the total number of required FLOPs at each iteration by different algorithms and percentage of their saved FLOPs compared to the full-update algorithm are also provided in Table 2 when $L_f = L_b$, $L = 30$, $p = 2, 4$ and $M = L/p$. In both tables for SPPU-NLMS-DFE, training takes up one-fifth of the transmission rate.

7. SIMULATIONS

In order to compare performance of the proposed equalizers with the full-update and data-independent partial-update NLMS-DFEs, simulation results are provided in this section. Two different channels were utilized for this purpose, a static channel with a fractionally-spaced impulse response and a time-varying channel with a baud-spaced impulse response. In all the simulations, 5000 QPSK symbols were transmitted in each run while the first 1000 symbols were used for training. Results were obtained by averaging over 1000 independent Monte Carlo runs. The adaptation step sizes

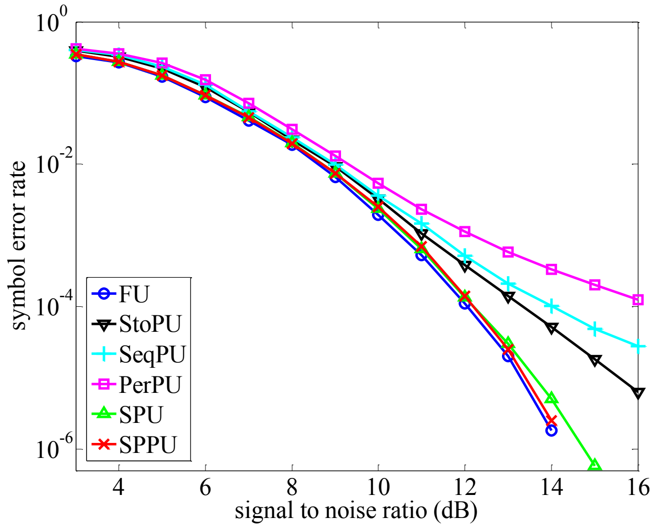

Fig. 3, SER-SNR curves of full-update and different partial-update NLMS-DFEs for a static channel.

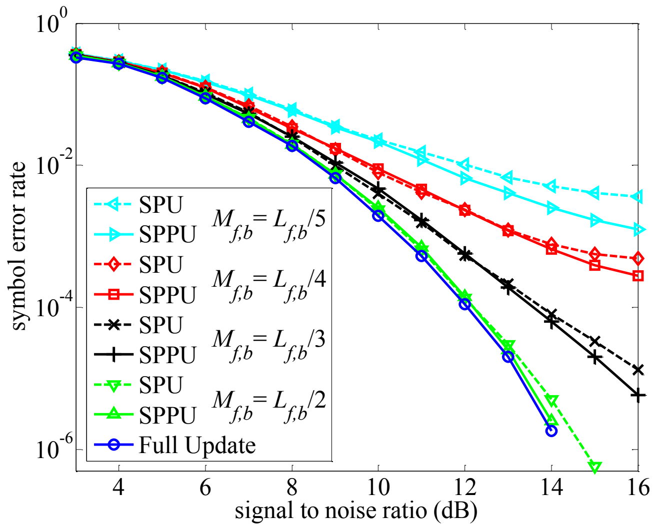

Fig. 4, SER-SNR curves of the new algorithms with different numbers of updated coefficients compared to the full-update NLMS-DFE.

were adjusted such that all the algorithms attained the same steady-state MSE.

7.1 Static channel

First part of the simulations was carried out using a terrestrial microwave channel impulse response (SPIB channel no. 3 from [12]), which is fractionally spaced ($T/2$) and comprises 300 taps. The corresponding equalizers have $L_f = 300$ (fractionally-space) and $L_b = 60$ coefficients and a decision delay of $\delta = 25$. Except in Fig. 4, partial-update equalizers update half of the coefficients at each time instant ($M_f = 150$ and $M_b = 30$). For periodic partial updates, it means half in average ($p = 2$).

Fig. 2 shows MSE curves of the full-update and different partial-update NLMS-DFEs at channel output signal to noise ratio (SNR) of 12 dB. It is clear that SPU and SPPU algorithms yield faster initial convergence than the other partial-update techniques. Fig. 3 depicts SER of different algorithms for different SNRs. It is observed that SPU and SPPU algorithms result in SER performances very close to the one of the full-update algorithm while they only update half of the filter coefficients at each iteration. SER-SNR curves of the proposed partial-update equalizers and the full-update NLMS-DFE are compared in Fig. 4 when different fractions of the coefficients are updated at each iteration. It clearly shows that how performance can be traded for complexity by using the new partial-update equalizers.

7.2 Time-varying channel

In this part, the 3GPP typical urban channel model [13] was considered. This channel comprises 20 taps and varies in time according to Jakes model [14] with a normalized Doppler frequency of 12.5×10^{-6}. The equalizers have $L_f = 40$ and $L_b = 20$ coefficients and a decision delay of $\delta = 40$. For partial updating, half of the coefficients are updated at each iteration ($M_f = 20$, $M_b = 10$, and $p = 2$).
Fig. 5 compares MSE curves of the full-update and different partial-update equalizers at SNR of 15 dB. SER performance of the equalizers for different SNRs is compared in Fig. 6. It is seen in Figs. 5 and 6 that the proposed equalizers achieve superior performance over other partial-update equalizers.

8. CONCLUSION

Employing partial coefficient updates in DFE was examined and a new selective-partial-update algorithm for NLMS-DFE was developed. The developed algorithm treats the feedforward and feedback adaptive filters separately and resolves the sorting ambiguity for the feedback filter by taking into account the errors associated with the past decisions. A new combined partial-update algorithm for NLMS-DFE was also proposed to take advantage of both selective-partial-update and periodic-partial-update techniques. Computer simulations demonstrate the capability of the proposed equalizers to achieve a good tradeoff between performance and complexity by means of partial updating. It is shown that in practical equalization scenarios assuming both static and dynamic channels, the proposed partial-update techniques offer appreciable complexity savings at the expense of slight performance degradation.

REFERENCES

[1] R. A. Casas, P. B. Schniter, J. Balakrishnan, C. R. Johnson, Jr., and Cornell Univ. BERG, *DFE Tutorial*, July 1998.
[2] C. A. Belfiore and J. H. Park, "Decision feedback equalization," *Proc. IEEE*, vol. 67, no. 8, pp. 1143–1156, Aug. 1979.
[3] A. H. Sayed, *Adaptive Filters*, Hoboken, NJ: Wiley, 2008.
[4] G. H. Golub and C. F. Van Loan, *Matrix Computations*, third ed., Baltimore, MD: Johns Hopkins Univ. Press, 1996.
[5] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Trans. on Circuits and Systems II*, vol. 44, no. 3, pp. 209–216, Mar. 1997.
[6] J. R. Treichler, C. R. Johnson, Jr., and M. G. Larimore, *Theory and Design of Adaptive Filters*, Wiley, NY, 1987.
[7] K. Doğançay, *Partial-Update Adaptive Signal Processing: Design, Analysis and Implementation,* Academic Press, Oxford, UK, 2008.
[8] M. Godavarti, and A. O. Hero III, "Partial update LMS algorithms," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2382–2399, July 2005.
[9] T. Aboulnasr, and K. Mayyas, "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1421–1424, May 1999.
[10] K. Doğançay and O. Tanrıkulu, "Adaptive filtering algorithms with selective partial updates," *IEEE Trans. Circuits Syst. II*, vol. 48, no. 8, pp. 762–769, Aug. 2001.
[11] J. Benesty, Y. Huang, and J. Chen, "A fast recursive algorithm for optimum sequential signal detection in a BLAST system," *IEEE Trans. Signal Process.*, vol. 51, no. 7, pp. 1722–1730, Jul. 2003.
[12] Rice University Signal Processing Information Base (SPIB), http://spib.rice.edu/spib/microwave.html.
[13] 3GPP TR 25.943 V6.0.0, 3rd Generation Partnership Project; Technical Specification Group Radio Access Networks; Deployment aspects (Release 6), Dec. 2004.
[14] W. C. Jakes, *Microwave Mobile Communications*, New York: Wiley, 1974.

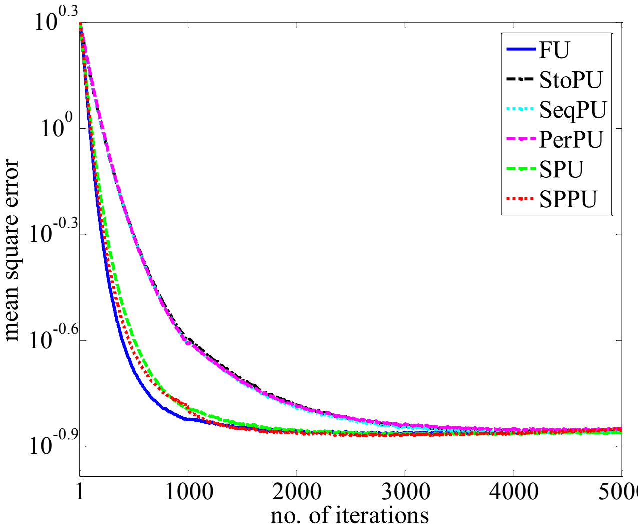

Fig. 5, MSE curves of full-update and different partial-update NLMS-DFEs for a time-varying channel.

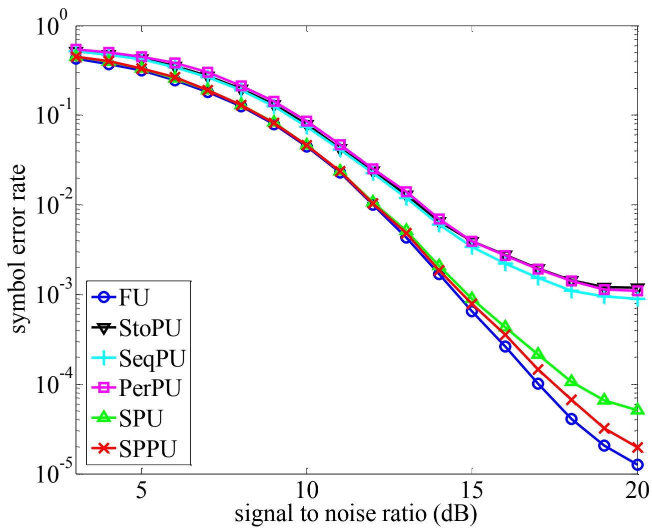

Fig. 6, SER-SNR curves of full-update and different partial-update NLMS-DFEs for a time-varying channel.