Variable Tap-Length Mixed-Tone RLS-based Per-Tone Equalisation with Adaptive Implementation

Suchada Sitjongsataporn[1], Non-member

ABSTRACT

In this paper, a methodology of mixed-tone recursive least squares algorithm development based on an orthogonal projection approach and a new variable tap-length mechanism is presented for per-tone equalisation (PTEQ) in discrete multitone systems. A mixed-tone cost function described as the sum of weight estimated errors is minimised to achieve the solutions for different per-tone equalisers simultaneously. We describe about the inverse square-root recursive least squares algorithm based upon the QR-decomposition which preserves the Hermitian symmetry of the inverse autocorrelation matrix by means of the product of square-root matrix and its Hermitian transpose. Such symmetrical property lends the benefit to the parallel implementation. In order to reduce the computational complexity, a new variable tap-length algorithm based on the sense of mean square mixted-tone errors is introduced to search a proper choice of tap-length of PTEQ. Simulation results show the improvement of achievable bit rate and signal to noise ratio performance as compared to the PTEQ exploiting conventional recursive least squares algorithm.

Keywords: Discrete Multitone (DMT), Adaptive Equalisation, Per-tone Equalisation (PTEQ), Mixed-tone Cost Function, Mixed-tone Recursive Least Squares (MT-RLS) Algorithm, Inverse Square-root RLS (iQR-RLS) Algorithm, a Variable Tap-length Algorithm

1. INTRODUCTION

Discrete multitone (DMT) is a digital implementation technique widely used for high speed wired multicarrier transmission such as asymmetric digital subscriber lines (ADSLs) [1]. It is a well known issue that, in DMT theory, there is no overlapping between tones (subcarriers) due to orthogonality derived from the discrete Fourier transformation among them. In practice, frequency selective fading channel generally destroys such orthogonal structure leading to information interfering from adjacent tones as commonly known as *intercarrier interference*.

In such case, information supposedly belonging to a particular subcarrier (or tone) generally smear into adjacent tones and leave some residual energy in them. In order to eliminate intercarrier interference (ICI) and intersymbol interference (ISI), the cyclic prefix (CP) is inserted among DMT-symbols to arrange subchannels separately.

Conventional equalisation of DMT-based systems consists of an adaptive (real) time-domain equaliser (TEQ) which shortens the convolutional result of TEQ and channel impulse response (CIR), so that ISI can be effectively handled by CP and ICI can also be mitigated. A (complex) one-tap frequency-domain equaliser (FEQ) is applied subsequently to compensate for amplitude and phase of distortion [1], [2]. However, TEQs are not designed to achieve the maximum bit rate performance [3]. The so-called per-tone equalisation which is a frequency-domain equalisation scheme for each tone has been introduced in [4].

A per-tone equalisation scheme using a technique based on transferring the (real) TEQ-operations to the frequency-domain is done per tone after the fast Fourier transform (FFT) demodulation. This enables us to accomplish the signal to noise ratio (SNR) optimisation per tone, because the equalisation of each tone is independent of other tones as suggested in [4]. This PTEQ performance has been presented to be better than any TEQ-based receiver, while its complexity during data transmission can be kept at the same level.

Based on the recursive least squares (RLS) algorithm, adaptive RLS-based PTEQ algorithm requires second order information as autocorrelation matrix of the sliding discrete Fourier transform (DFT) of the received signal. In [5], it is shown that a significant part of RLS-based computations for storing and updating can be shared among different tones leading to sufficiently low initialisation complexity.

Nevertheless, the PTEQs concentrate on the detection on their own tones and do not explicitly reject the interference of transmitted information supposedly belonging to the others. It is possible for the equalisation in the tone of interest, however, to retrieve adaptively the knowledge of residual interfering signal energy from adjacent tones to gain SNR

Manuscript received on June 2, 2011 ; revised on October 18, 2012.

[1] The author is with Centre of Electronic Systems Design and Signal Processing (CESdSP) Department of Electronic Engineering, Mahanakorn University of Technology 140 Cheumsamphan Road, Nong Chok, Bangkok 10530 Thailand., E-mail: ssuchada@mut.ac.th

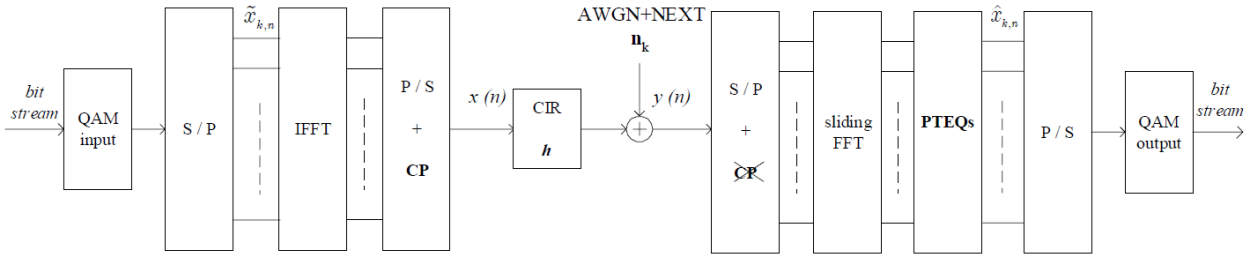

Fig.1: *Block diagram of a discrete multitone system.*

improvement of the tone of interest. This principle leads to the proposal of a novel structure of a mixed-tone PTEQ for DMT systems [6].

Because of the numerical instability of conventional RLS algorithm, the inverse Square-Root RLS (iQR-RLS) algorithm, known as the inverse QR-decomposition RLS algorithm, is used in parallel implementation in forms of the fast algorithm with the method of [7], [8]. In addition, the symmetrical LU-decomposition of inverse covariance matrix also befits to parallel implementation in forms of systolic array [9].

According to the complexity and convergence rate, the tap-length of a linear filter is an important parameter that influences significantly the performance and computational complexity of a minimum mean square error (MMSE) adaptive filter. As proved in [10], the shorter filter has faster convergence than the longer one. It is desirable to derive algorithms that can automatically find the optimum tap-length. In recent years, several variable tap-length mechanisms have been proposed in [11]-[13].

Based on the least mean square (LMS) algorithm, the computational complexity cost and excess mean square error will increase if the tap-length is too long [12]. Therefore, a variable tap-length algorithm needs to search a proper choice of the tap-length. The fractional tap-length (FT) algorithm is designed to consider the optimal filter tap-length based on the MSE output. As analysed in [11], the variable tap-length algorithm using the concept of the FT algorithm is more robust and has lower computational cost when compared with other methods. But there is no scheme that optimises the tap-length based on the mean square mixed-tone errors and the recursive least squares (RLS) algorithm.

We describe an overview of system model and notation in Section 2.and a solution of the PTEQ design criterion as given in Section 3. The derivation of proposed mixed-tone inverse squre-root recursive least squres (MTiQR-RLS) criterion is described in Section 4.and can be designed recursively with the orthogonal projection.

We introduce a new variable tap-length RLS algorithm with reduced computational complexity per update in Section 4. 5 based on the fractional tap-length mechanism for the complex-valued PTEQ framework. The proposed MTiQR-RLS algorithm with a new variable tap-length method presented in Section 5. which is particularly designed to work in conjunction with the complex-valued PTEQ structure. Analysis of proposed variable tap-length algorithm is introduced in Section 6. Simulation results show in Section 7.and Section 8.concludes the paper.

2. SYSTEM MODEL AND NOTATION

We describe briefly the basic structure of the DMT transceiver shown in Fig. 1. The incoming bit stream is mapped using the quadrature amplitude modulation on each tone and then split into N parallel paths. A sequence of $N \times 1$ input vectors $\tilde{\mathbf{x}}_{k,N}$, where $\tilde{\mathbf{x}}_{k,N} = [\tilde{x}_{k,0} \quad \cdots \quad \tilde{x}_{k,N-1}]^T$, is also modulated with the method of inverse discrete Fourier transform (IDFT). Including the CP among symbols, the IDFT outputs are then transformed into serial DMT-symbols and sent through the channel.

The data model based on an FIR model of the transmission channel is presented in (1) shown at the bottom of this page, where Δ is the synchronisation delay and l denotes the first considered sample of the k-th received DMT symbol vector $\mathbf{y}$ and transmitted vector $\mathbf{x}$.

$$\underbrace{\begin{bmatrix} y_{k,l+\Delta} \\ \vdots \\ y_{k,N-l+\Delta} \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} \mathbf{0}_{(1)} & \overbrace{\begin{bmatrix} [\,\bar{\mathbf{h}}^T\,] & 0 & \cdots \\ & \ddots & \ddots \\ \cdots & 0 & [\,\bar{\mathbf{h}}^T\,] \end{bmatrix}}^{\mathbf{H}_T} & \mathbf{0}_{(2)} \end{bmatrix}}_{\mathbf{H}} \cdot (\mathbf{I} \otimes \boldsymbol{\mathcal{P}}_\nu \boldsymbol{\mathcal{F}}_N^*) \cdot \underbrace{\begin{bmatrix} \mathbf{x}_{k-1,N} \\ \mathbf{x}_{k,N} \\ \mathbf{x}_{k+1,N} \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} \eta_{k,l+\Delta} \\ \vdots \\ \eta_{k,N-l+\Delta} \end{bmatrix}}_{\boldsymbol{\eta}}, \tag{1}$$

The $\boldsymbol{\eta}$ is a vector with the additive white Gaussian noise (AWGN) and near-end crosstalk (NEXT). The $\bar{\mathbf{h}}$ is channel impulse responce (CIR) vector in reverse order. The $\boldsymbol{\mathcal{F}}_N^* = \boldsymbol{\mathcal{F}}_N^H$ is of $N \times N$ (I)DFT matrix and N is the size of (I)DFT. The $(N+\nu)\times N$ $\boldsymbol{\mathcal{P}}_\nu$ matrix which adds the CP of length ν. The matrices $\mathbf{0}_{(1)}$ and $\mathbf{0}_{(2)}$ are also the zero matrices. The matrix $\mathbf{I}$ is an identity matrix. The parameter N_d is range of active tones, n is the tone index and k is the symbol index.

Some notation will be used throughout this paper as follows: the operator $(\cdot)^H$, $(\cdot)^T$ and $(\cdot)^*$ denote as the Hermitian, transpose and complex conjugate operator, respectively. A tilde over the variable indicates the frequency-domain. The vectors are in bold lowercase and matrices are in bold uppercase.

3. PER-TONE EQUALISATION

This section describes the concept of per-tone equaliser (PTEQ) in DMT-based systems. We refer the readers to [4] for more details. The per-tone equalisation structure is based on transferring the TEQ-operations into the frequency-domain ones after DFT demodulation, which results in a L-tap PTEQ for each tone separately. For each tone n, the TEQ-operation is shown as follows.

$$\tilde{d}_n = \overbrace{\tilde{z}_n}^{\text{1-tap FEQ}} \cdot \text{row}_n \overbrace{(\boldsymbol{\mathcal{F}}_N) \cdot (\mathbf{Y} \cdot \mathbf{w})}^{\text{1 DFT}}, \quad (2)$$

$$= \text{row}_n \underbrace{(\boldsymbol{\mathcal{F}}_N \cdot \mathbf{Y})}_{L \text{ DFTs}} \cdot \underbrace{\mathbf{w} \cdot \tilde{z}_n}_{L-\text{tap FEQ } \mathbf{v}_n}, \quad (3)$$

where $\tilde{d}_n$ is the output after frequency-domain equalisation for tone n. The $\tilde{z}_n$ is the (complex) 1-tap FEQ for tone n. The parameter $\mathbf{w}$ is of (real) L-tap TEQ. The matrix $\mathbf{Y}$ is a Toeplitz matrix of channel output samples.

The PTEQ output $\hat{x}_k$ on each tone n can be specified for $n \in N_d$ as $\hat{x}_k = \hat{\mathbf{p}}_k^H \cdot \tilde{\mathbf{y}}_k$, where $\hat{\mathbf{p}}_k$ is the L-tap complex-valued PTEQ vector at symbol k. The $\tilde{\mathbf{y}}_k$ is the sliding DFT output as

$$\tilde{\mathbf{y}}_k = \underbrace{\left[\begin{array}{c|c|c} \mathbf{I}_{L-1} & \mathbf{0} & \text{-}\mathbf{I}_{L-1} \\ \hline \mathbf{0} & \multicolumn{2}{c}{\mathcal{F}_N(n,:)} \end{array}\right]}_{\mathbf{F}_n} \cdot \mathbf{y}, \quad (4)$$

where $\mathbf{F}_n$ is a $(L-1)\times(N+L-1)$ matrix. By using the sliding DFT, the first block row of matrix $\mathbf{F}_n$ in (4) extracts the difference terms where $\mathcal{F}_N(n,:)$ is the n-th row of $\mathcal{F}_\mathcal{N}$, while the last row corresponds to the usual DFT operation as shown in [4]. The vector $\mathbf{y}$ is of channel output samples as described in (1).

4. MIXED-TONE INVERSE SQUARE-ROOT RECURSIVE LEAST SQUARES PER-TONE EQUALISATION WITH VARIABLE TAP-LENGTH

In this section, we describe shortly the orthogonal projection matrix for the mixed-tone PTEQ in Section 4.1. The mixed-tone cost function which is presented as the sum of mixed-tone weight-estimated errors in Section 4.2.

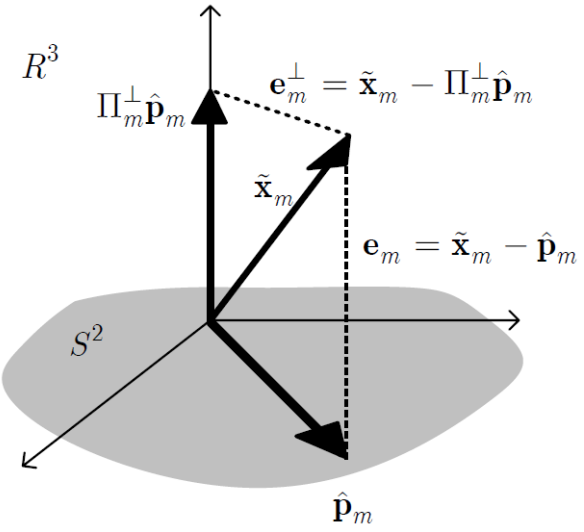

Fig.2: *A $\hat{\mathbf{p}}_m$ and its orthogonal projection as well as $\tilde{\mathbf{x}}_m$ are shown in two-dimentional subspace S^2 [6].*

In Section 4.3 , we derive an adaptive mixed-tone RLS with orthogonal projection algorithm for per-tone equalisation by means of the mixed-tone cost function. Section 4.4 introduces the inverse square-root recursive least squares (iQR-RLS) algorithm based on the mixed-tone cost function. Then, we introduce a new variable tap-length method in the sense of mean square mixed-tone errors presented in Section 4.5.

4.1 The orthogonal projection matrix

The illustration of the $\hat{\mathbf{p}}_m$ and its orthogonal projection as well as $\tilde{\mathbf{x}}_m$ for two-dimensional subspace S^2 is depicted in Fig. 2. The error $\mathbf{e}_m^\perp$ associated with the orthogonal projection of $\hat{\mathbf{p}}_m$, $\Pi_m^\perp \hat{\mathbf{p}}_m$, where $\Pi_m^\perp$ denotes as the orthogonal projection matrix of $\hat{\mathbf{p}}_m$, will be used in the update of the $\hat{\mathbf{p}}_g$ where $g \neq m$.

The orthogonal projection matrix $\Pi_{k,m}^\perp$ which is the matrix difference determined by the vector $\hat{\mathbf{p}}_m$ as [14]

$$\Pi_m^\perp = \mathbf{I} - \hat{\mathbf{p}}_m \left[\hat{\mathbf{p}}_m^H \hat{\mathbf{p}}_m\right]^{-1} \hat{\mathbf{p}}_m^H, \quad (5)$$

where $\mathbf{I}$ denotes as the identity matrix. We note that the orthogonal projection matrix is mentioned by the vector $\hat{\mathbf{p}}_m$ at tone m.

4.2 The mixed-tone recursive least squares cost function

We introduce the idea of using orthogonal projection of adjacent equalisers to include the information of interfering tones while equalisation of the interefering tones themselves are not affected. Therefore, the

mixed-tone exponentially least squares cost function to be minimised is defined as [6]

$$J_{k,m} = \frac{1}{2}\sum_{m=1}^{M}\sum_{i=1}^{k}\lambda^{k-i}\{|\xi_{i,m}|^2\} , \quad (6)$$

where $\xi_{i,m}$ denotes as the mixed-tone estimate error with the parameter estimate tap-weight $\hat{\mathbf{p}}_{k,m}$ for tone $m \in M$ as

$$\xi_{i,m} = \tilde{x}_{i,m} - \hat{\mathbf{p}}_{k,m}^{H}\tilde{\mathbf{y}}_{i,m} - \sum_{l=1}^{L}\big(\Pi_{k,l}^{\perp}\,\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{i,l} , \quad (7)$$
$$\text{for } m \neq l \ , \ L \leq M-1 \ .$$

and λ is the exponential weighting-factor or forgetting-factor. The orthogonal projection matrix applied to the tap-weight vector $\hat{\mathbf{p}}_{k,m}$ as $\Pi_{k,m}^{\perp}\hat{\mathbf{p}}_{k,m}$ yields the mixed-tone estimated error $\xi_{k,m}$.

With the definition for this cost function, the mth term on the right-hand side of (6) represents as the estimated error of the step k due to the m^{th}-tone equaliser $\hat{\mathbf{p}}_{k,m}$.

4.3 A mixed-tone recursive least squares algorithm

We now demonstrate the derivation of adaptive mixed-tone recursive least squares (MT-RLS) algorithm for per-tone equalisation in DMT-based systems.

The objective of this proposed algorithm is to minimise the mixed-tone cost function $J_{k,m}$ shown in (6) for each tone m, where $m \in M$.

By taking the derivative of (6) with respect to $\hat{\mathbf{p}}_{k,m}^{H}$, we have

$$\frac{\partial\xi_{k,m}}{\partial\hat{\mathbf{p}}_{k,m}^{H}} = -\sum_{i=1}^{k}\lambda^{k-i}\tilde{\mathbf{y}}_{i,m}\big\{\tilde{x}_{i,m}^{*} - \sum_{l=1}^{L}\big(\Pi_{k,l}^{\perp}\,\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{k,l}\big\} . \quad (8)$$

Setting the result in (8) equals to zero. Then, we get

$$-\sum_{i=1}^{k}\lambda^{k-i}\tilde{\mathbf{y}}_{i,m}\tilde{\mathbf{y}}_{i,m}^{H}\,\hat{\mathbf{p}}_{k,m} + \sum_{i=1}^{k}\lambda^{k-i}\,\tilde{\mathbf{y}}_{i,m}\tilde{x}_{i,m}^{*} - \sum_{i=1}^{k}\lambda^{k-i}\big\{\tilde{\mathbf{y}}_{k,l}\sum_{l=1}^{L}\big(\Pi_{k,l}^{\perp}\,\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{k,l}\big\} = 0 . \quad (9)$$

The complexed value of tap-weight estimated $\hat{\mathbf{p}}_{k,m}$ vector in (9) can be expressed by the normal equations as

$$\mathbb{R}_{k,m}\,\hat{\mathbf{p}}_{k,m} = \tilde{\mathbf{z}}_{k,m} , \quad (10)$$

where $\mathbb{R}_{k,m}$ denotes as an autocorrelation matrix of the tap-input vector $\tilde{\mathbf{y}}_{i,m}$ as

$$\mathbb{R}_{k,m} = \sum_{i=1}^{k}\lambda^{k-i}\,\tilde{\mathbf{y}}_{i,m}\tilde{\mathbf{y}}_{i,m}^{H} , \quad (11)$$

where λ is a forgetting-factor parameter and $\tilde{\mathbf{z}}_{k,m}$ is the cross-correlation vector as

$$\tilde{\mathbf{z}}_{k,m} = \sum_{i=1}^{k}\lambda^{k-i}\,\tilde{\mathbf{y}}_{i,m}\big\{\tilde{x}_{i,m}^{*} - \sum_{l=1}^{L}\big(\Pi_{k,l}^{\perp}\,\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{k,l}\big\} . \quad (12)$$

We may rewrite (11) in the recursion form as

$$\mathbb{R}_{k,m} = \lambda\,\mathbb{R}_{k-1,m} + \{\tilde{\mathbf{y}}_{k,m}\,\tilde{\mathbf{y}}_{k,m}^{H}\} , \quad (13)$$

where $\mathbb{R}_{k,m}$ denotes as the autocorrelation matrix of tap-inputs $\tilde{\mathbf{y}}_{k,m}$ for each tone m at symbol k.

In a similar fashion, the cross-correlation vector $\tilde{\mathbf{z}}_{k,m}$ in (12) can be expressed recursively as

$$\tilde{\mathbf{z}}_{k,m} = \lambda\,\tilde{\mathbf{z}}_{k-1,m} + \tilde{\mathbf{y}}_{k,m}\big\{\tilde{x}_{k,m}^{*} - \sum_{l=1}^{L}\big(\Pi_{k,l}^{\perp}\,\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{k,l}\big\} . \quad (14)$$

From (10), the estimate of tap-weight $\hat{\mathbf{p}}_{k,m}$ vector is then defined as

$$\hat{\mathbf{p}}_{k,m} = \mathbb{R}_{k,m}^{-1}\,\tilde{\mathbf{z}}_{k,m} . \quad (15)$$

where $\mathbb{R}_{k,m}^{-1}$ denotes as an inverse of the autocorrelation matrix for tone m at symbol k and $\tilde{\mathbf{z}}_{k,m}$ is given in (14).

Using the matrix inversion lemma[1], the inverse of the autocorrelation matrix $\mathbb{R}_{k,m}^{-1}$ can then be calculated in the following form as

$$\mathbb{R}_{k,m}^{-1} = \lambda^{-1}\mathbb{R}_{k-1,m}^{-1} - \frac{\lambda^{-2}\mathbb{R}_{k-1,m}^{-1}\tilde{\mathbf{y}}_{k,m}\tilde{\mathbf{y}}_{k,m}^{H}\mathbb{R}_{k-1,m}^{-1}}{1+\lambda^{-1}\tilde{\mathbf{y}}_{k,m}^{H}\mathbb{R}_{k-1,m}^{-1}\tilde{\mathbf{y}}_{k,m}} . \quad (16)$$

Substituting (16) in (15), we have

$$\hat{\mathbf{p}}_{k,m} = \lambda^{-1}\mathbb{R}_{k-1,m}^{-1}\tilde{\mathbf{z}}_{k,m} - \lambda^{-1}\tilde{\mathbf{k}}_{k,m}\tilde{\mathbf{y}}_{k,m}^{H}\mathbb{R}_{k-1,m}^{-1}\tilde{\mathbf{z}}_{k,m} , \quad (17)$$

where $\tilde{\mathbf{k}}_{k,m}$ is the gain vector as given as

$$\tilde{\mathbf{k}}_{k,m} = \frac{\lambda^{-1}\mathbb{R}_{k-1,m}^{-1}\tilde{\mathbf{y}}_{k,m}}{1+\lambda^{-1}\tilde{\mathbf{y}}_{k,m}^{H}\mathbb{R}_{k-1,m}^{-1}\tilde{\mathbf{y}}_{k,m}} . \quad (18)$$

The matrix inversion lemma [15] Let A and B be two positive definite M-by-M matrices related by $A = B^{-1} + C\cdot D^{-1}\cdot C^{H}$ where D is a positive definite N-by-M matrix and C is a M-by-N matrix. We may express the inverse of the matrix A by $A^{-1} = B - BC\,(D + C^{H}\,B\,C)^{-1}\,C^{H}\,B$.

Then, substituting (14) in (17), we arrive

$$\begin{aligned}\hat{\mathbf{p}}_{k,m} &= \lambda^{-1}\mathbb{R}^{-1}_{k-1,m}\lambda\,\tilde{\mathbf{z}}_{k-1,m}\\ &+ \lambda^{-1}\mathbb{R}^{-\mathbb{1}}_{\mathbb{k}-\mathbb{1},}\tilde{\mathbf{y}}_{\mathbb{k},}\{^{*}_{\mathbb{k},} - \sum_{=\mathbb{1}}^{\mathbb{L}}\big(^{\perp}_{\mathbb{k},}\hat{\mathbf{p}}_{\mathbb{k},}\big)^{\mathbb{H}}\tilde{\mathbf{y}}_{\mathbb{k},}\}\\ &- \lambda^{-1}\tilde{\mathbf{k}}_{k,m}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{R}^{-1}_{k-1,m}\tilde{\mathbf{y}}_{k,m}\{\tilde{x}^{*}_{k,m} - \sum_{l=1}^{L}\big(\Pi^{\perp}_{k,l}\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{k,l}\}\\ &- \lambda^{-1}\tilde{\mathbf{k}}_{k,m}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{R}^{-1}_{k-1,m}\lambda\,\tilde{\mathbf{z}}_{k-1,m}\hat{\mathbf{p}}_{k-1,m}\ .\end{aligned} \tag{19}$$

We may rewrite (19) as

$$\begin{aligned}\hat{\mathbf{p}}_{k,m} &= \hat{\mathbf{p}}_{k-1,m} - \tilde{\mathbf{k}}_{k,m}\tilde{\mathbf{y}}^{H}_{k,m}\hat{\mathbf{p}}_{k-1,m}\\ &- \lambda^{-1}\tilde{\mathbf{k}}_{k,m}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{R}^{-1}_{k-1,m}\tilde{\mathbf{y}}_{k,m}\{\tilde{x}^{*}_{k,m} - \sum_{l=1}^{L}\big(\Pi^{\perp}_{k,l}\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{k,l}\}\\ &+ \lambda^{-1}\mathbb{R}^{-1}_{k-1,m}\tilde{\mathbf{y}}_{k,m}\{\tilde{x}^{*}_{k,m} - \sum_{l=1}^{L}\big(\Pi^{\perp}_{k,l}\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{k,l}\}\ .\end{aligned} \tag{20}$$

By rearranging (18), we obtain

$$\begin{aligned}\tilde{\mathbf{k}}_{k,m} &= \lambda^{-1}\mathbb{R}^{-1}_{k-1,m}\tilde{\mathbf{y}}_{k,m} - \lambda^{-1}\tilde{\mathbf{k}}_{k,m}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{R}^{-1}_{k-1,m}\tilde{\mathbf{y}}_{k,m}\\ &= \big[\lambda^{-1}\mathbb{R}^{-1}_{k-1,m} - \lambda^{-1}\tilde{\mathbf{k}}_{k,m}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{R}^{-1}_{k-1,m}\big]\,\tilde{\mathbf{y}}_{k,m}\ .\end{aligned} \tag{21}$$

Using (18), we may rewrite the inverse autocorrelation matrix $\mathbb{R}^{-1}_{k,m}$ in (16) as

$$\mathbb{R}^{-1}_{k,m} = \lambda^{-1}\mathbb{R}^{-1}_{k-1,m} - \lambda^{-1}\tilde{\mathbf{k}}_{k,m}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{R}^{-1}_{k-1,m}\ . \tag{22}$$

We note that the expression inside the bracket on the right-handed side of (21) equals $\mathbb{R}^{-1}_{k,m}$ in (22). We may simplify the gain vector $\tilde{\mathbf{k}}_{k,m}$ in (21) to

$$\tilde{\mathbf{k}}_{k,m} = \mathbb{R}^{-1}_{k,m}\,\tilde{\mathbf{y}}_{k,m}\ . \tag{23}$$

Finally, using (22) and (23) into (20), we get the updating the tap-weight estimate PTEQ vector $\hat{\mathbf{p}}_{k,m}$ for $m \in M$ as

$$\hat{\mathbf{p}}_{k,m} = \hat{\mathbf{p}}_{k-1,m} + \mathbb{R}^{-1}_{k,m}\tilde{\mathbf{y}}_{k,m}\,\xi^{*}_{k,m}\ , \tag{24}$$

where the conjugate of *a prior* estimate mixed-tone error $\xi_{k,m}$ at symbol k is given as

$$\xi_{k,m} = \tilde{x}_{k,m} - \hat{\mathbf{p}}^{H}_{k-1,m}\,\tilde{\mathbf{y}}_{k,m} - \sum_{l=1}^{L}\big(\Pi^{\perp}_{k,l}\hat{\mathbf{p}}_{k,l}\big)^{H}\tilde{\mathbf{y}}_{k,l}\ . \tag{25}$$

$$\text{for } m \neq l\ ,\ L \leq M-1$$

4.4 A mixed-tone inverse square-root recursive least squares algorithm

We then describe briefly the inverse square-root RLS (iQR-RLS) algorithm that determines especially on the inverse autocorrelation matrix, in place of operating on the correlation matrix as in the conventional QR-RLS algorithm [16].

Considering the step-by-step among the Kalman and RLS variables, the iQR-RLS algorithm is a fundamentally square-root covariance Kalman algorithm as given in [17].

For the convenience of computation, let

$$\mathbb{Q}_{k,m} = \mathbb{R}^{-1}_{k,m}\ . \tag{26}$$

Following [18], the inverse autocorrelation matrix in (16) may be expressed as

$$\mathbb{Q}_{k,m} = \lambda^{-1}\mathbb{Q}_{k-1,m} - \lambda^{-2}\mathbb{Q}_{k-1,m}\tilde{\mathbf{y}}_{k,m}\gamma^{-1}_{k,m}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{Q}_{k-1,m}\ , \tag{27}$$

where the parameter $\gamma_{k,m}$ is given as

$$\gamma_{k,m} = 1 + \lambda^{-1}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{Q}_{k-1,m}\tilde{\mathbf{y}}_{k,m}. \tag{28}$$

We then introduce the block matrix $\mathbb{M}$, its result consists of the matrix product of right-handed on (27) using the Cholesky factorisation as follows.

$$\mathbb{M} = \begin{bmatrix} 1 & \boldsymbol{\zeta}_{k,m} \\ \mathbf{0} & \lambda^{-\frac{1}{2}}\mathbb{Q}^{\frac{1}{2}}_{k-1,m} \end{bmatrix}\begin{bmatrix} 1 & \mathbf{0}^{T} \\ \boldsymbol{\zeta}^{H}_{k,m} & \lambda^{-\frac{1}{2}}\mathbb{Q}^{\frac{H}{2}}_{k-1,m} \end{bmatrix}, \tag{29}$$

$$\boldsymbol{\zeta}_{k,m} = \lambda^{-\frac{1}{2}}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{Q}^{\frac{1}{2}}_{k-1,m}. \tag{30}$$

We may set the prearray to resulting postarray transformation for inverse QR-RLS algorithm.

$$\begin{bmatrix} 1 & \lambda^{-\frac{1}{2}}\tilde{\mathbf{y}}^{H}_{k,m}\mathbb{Q}^{\frac{1}{2}}_{k-1,m} \\ \mathbf{0} & \lambda^{-\frac{1}{2}}\mathbb{Q}^{\frac{1}{2}}_{k-1,m} \end{bmatrix}\Theta \;=\; \begin{bmatrix} \gamma^{\frac{1}{2}}_{k,m} & \mathbf{0}^{T} \\ \tilde{\mathbf{k}}_{k,m}\gamma^{\frac{1}{2}}_{k,m} & \mathbb{Q}^{\mathbb{1}}_{\mathbb{k},} \end{bmatrix}\ , \tag{31}$$

where Θ is a unitary rotation.

Note that the inverse autocorrelation $\mathbb{Q}^{\frac{1}{2}}_{k,m}$ in (31) is the upper triangular matrix. Accordingly, the inverse autocorrelation $\mathbb{Q}_{k,m}$ may be defined with its factor as

$$\mathbb{Q}_{k,m} = \mathbb{Q}^{\frac{1}{2}}_{k,m}\,\mathbb{Q}^{\frac{H}{2}}_{k,m}, \tag{32}$$

in virtue of the product of square matrix and its Hermitian transpose is always nonnegative matrix as described in [17]. Therefore, the tap-weight estimate PTEQ vector $\hat{\mathbf{p}}_{k,m}$ for $m \in M$ based on the proposed mixed-tone inverse square-root RLS (MTiQR-RLS) algorithm in the recursion form may be computed by

$$\hat{\mathbf{p}}_{k,m} = \hat{\mathbf{p}}_{k-1,m} + \mathbb{Q}_{k,m}\,\tilde{\mathbf{y}}_{k,m}\,\xi^{*}_{k,m}\ , \tag{33}$$

where $\xi_{k,m}$ is given in (25).

4.5 A variable tap-length algorithm

We introduce a new variable tap-length method for adjusting the tap-length and following the change of the received signal in the sense of mean square mixed-tone error described in Section 4.2 We assume that the tap-length of the adaptive filter at steady-state condition is a fixed value and denoted by L.

The corresponding steady-state adaptive tap-weight vector $\hat{\mathbf{p}}_L$ and input vector $\tilde{\mathbf{y}}_{L,k}$, we define the segmented steady-state mixed-tone error $\tilde{xi}_{G,k,m}$ at symbol k as

$$\begin{aligned}\tilde{\xi}_{G,k,m} = \tilde{x}_{k,m} \quad &- \quad \hat{\mathbf{p}}^H_{G,k,m}\tilde{\mathbf{y}}_{G,k,m} \qquad (34)\\ &- \quad \sum_{l=1}^{L}\left(\Pi^{\perp}_{G,k,l}\,\hat{\mathbf{p}}_{G,k,l}\right)^H\tilde{\mathbf{y}}_{G,i,l}\ ,\\ &\qquad \text{for} \quad m \neq l\ ,\ L \leq M-1\ .\end{aligned}$$

where $1 \leq G \leq L$. The vectors $\hat{\mathbf{p}}_{G,m}$ and $\tilde{\mathbf{y}}_{G,k,m}$ are consisting of the first G-coefficient of the steady-state tap-weight vector $\hat{\mathbf{p}}_{L,m}$ and the input vector $\tilde{\mathbf{y}}_{L,k}$. An integer M is called the suboptimum tap-length. The width of suboptimum is defined as the number of successive suboptimum tap-length.

The mean square of this segmented steady-state mixed-tone error with the tap-length G is defined as

$$\tilde{\zeta}_{G,k} = E\{\ |\tilde{\xi}_{G,k,m}|^2\ \}\ , \qquad (35)$$

where $E\{\cdot\}$ represents the statistical expectation.

The basis of this method is to find the minimum value of L as $\min\{L \mid \tilde{\zeta}_{L-\Delta} - \tilde{\zeta}_L \leq \epsilon\}$, where Δ is a positive integer and ϵ is a small positive value.

We introduce an adaptive pseudo-fractional tap-length algorithm which is controlled using the difference of square of prediction mixed-tone errors with the different tap-length, corresponding to $L_k - \Delta$ and L_k by the exponential average as

$$\mathcal{L}_{f_{k+1}} = \lambda\,\mathcal{L}_{f_k} + \beta\,(|\tilde{\xi}_{L_k-\Delta,k}|^2 - |\tilde{\xi}_{L_k,k}|^2)\ , \qquad (36)$$

where λ is a forgetting-factor which is set close to 1. The parameter β is the step-size for tap-length adaptation.

Then, the tap-length L_k is obtained from

$$L_{k+1} = \begin{cases} \lfloor \mathcal{L}_{f_k} \rfloor, & |L_k - \mathcal{L}_{f_k}| \geq \delta\ ,\\ L_k\ , & \text{otherwise}\end{cases} \qquad (37)$$

where $\lfloor\cdot\rfloor$ is the floor operator and rounds the embraced value to the nearest integer. The parameter δ is a small integer.

It is obvious from (36) that, we may set $\min\{\mathcal{L}_{f_k}\} = \Delta + 1$ to ensure $\mathcal{L}_{f_k} \geq \Delta + 1$. Then, we can verify that $\min\{L_k\} = \min\{\mathcal{L}_{f_k}\} + \delta$, as concluded in [11]. Thus, the fractional tap-length $\mathcal{L}_{f_k}$ and tap-length L_k should be initialised under this conditions: $\mathcal{L}_f(0) \geq \Delta + 1$ and $L(0) \geq \Delta + \delta + 1$ in the simulations.

5. MTIQR-RLS PTEQS WITH VARIABLE TAP-LENGTH

A recursive initialisation based on the QR decomposition-based iQR-RLS algorithm [16] which stores and updates the upper triangular of square-root inverse matrix $\mathbb{U}_{k,n}$, where the inverse autocorrelation $\mathbb{Q}_{k,m}$ given in (27) is redefined using the product of square matrix and its Hermitian transpose in (32) as

$$\mathbb{Q}_{k,m} = \mathbb{U}_{k,m}\,\mathbb{U}^H_{k,m}\ . \qquad (38)$$

We then introduce the proposed mixed-tone inverse square-root recursive least squares per-tone equalisers (MTiQR-RLS PTEQs) with variable tap-length that adjust adaptively for tone $m \in M$. The following pseudocode constitutes this algorithm.

Adaptive Algorithm : MTiQR-RLS with variable tap-length

For $m = 1, \ldots, M$

For $k = 1, \ldots, K$

Initialise the tone independent $L(0), \mathcal{L}(0), \mathbb{U}_0$, $\hat{\mathbf{p}}_0$, $\mathbb{Q}_0$, and ξ_0.

1. Update L_k as

$$\begin{aligned}\mathcal{L}_{f_{k+1}} &= \lambda\,\mathcal{L}_{f_k} + \gamma\,(|\xi_{L_k-\Delta,k,m}|^2 - |\xi_{L_k,k,m}|^2)\\ L_{k+1} &= \begin{cases}\lfloor\mathcal{L}_{f_k}\rfloor, & |L_k - \mathcal{L}_{f_k}| \geq \delta\\ L_k\ , & \text{otherwise}\end{cases}\end{aligned}$$

2. Form the matrix-vector product as

$$\boldsymbol{a} = \lambda^{-\frac{1}{2}}\tilde{\mathbf{y}}^H_{L_k,k,m}\mathbb{U}_{L_k,k-1,m}$$

3. Determine Givens rotation [15] $\mathbf{Q}^a_t$ for $t = 1, \ldots, L_k$ and update $\mathbb{L}_{k,m}$ as

$$\begin{aligned}\begin{bmatrix}\mathbf{0}_{(L_k-1)\times 1}\\ \hline \delta\end{bmatrix} &\Leftarrow \mathbf{Q}_{L_k}\cdot\mathbf{Q}_{L_k-1}\cdots\mathbf{Q}_1\begin{bmatrix}\boldsymbol{a}\\ \hline 1\end{bmatrix}\\ \mathbb{D} &\Leftarrow \lambda^{-\frac{1}{2}}\mathbb{U}_{L_k,k-1,m}\\ \begin{bmatrix}\mathbb{U}_{L_k,k,m}\\ \hline \delta\cdot\tilde{\mathbf{k}}_{k,m}\end{bmatrix} &\Leftarrow \mathbf{Q}_{L_k}\cdot\mathbf{Q}_{L_k-1}\cdots\mathbf{Q}_1\begin{bmatrix}\mathbb{D}\\ \hline \mathbf{0}_{1\times(L_k-1)}\end{bmatrix}\\ \mathbb{Q}_{L_k,k,m} &\Leftarrow \mathbb{U}_{L_k,k,m}\mathbb{U}^H_{L_k,k,m}\end{aligned}$$

4. Update $\xi_{L_k,k,m}$ as

For $l = 1, \ldots, M$ and $l \neq m$

$$\begin{aligned}\Pi^{\perp}_{L_k,k,l} &\Leftarrow \mathbf{I} - \hat{\mathbf{p}}_{L_k,k,l}\,[\hat{\mathbf{p}}^H_{L_k,k,l}\,\hat{\mathbf{p}}_{L_k,k,l}]^{-1}\,\hat{\mathbf{p}}^H_{L_k,k,l}\\ \xi_{L_k,k,m} &\Leftarrow \tilde{x}_{k,m} - \hat{\mathbf{p}}^H_{L_k,k-1,m}\tilde{\mathbf{y}}_{L_k,k,m}\\ &\quad - (\Pi^{\perp}_{L_k,k,l}\hat{\mathbf{p}}_{L_k,k,l})^H\,\tilde{\mathbf{y}}_{L_k,k,l}\\ &\quad - (\Pi^{\perp}_{L_k,k,l-1}\hat{\mathbf{p}}_{L_k,k,l-1})^H\,\tilde{\mathbf{y}}_{L_k,k,l-1}\\ &\quad - \ldots - (\Pi^{\perp}_{L_k,k,M}\hat{\mathbf{p}}_{L_k,k,M})^H\,\tilde{\mathbf{y}}_{L_k,k,M}\end{aligned}$$

end

5. Update $\hat{\mathbf{p}}_{L_k,k,m}$ as

$$\hat{\mathbf{p}}_{L_k,k,m} \Leftarrow \hat{\mathbf{p}}_{L_k,k-1,m} + \mathbb{Q}_{L_k,k,m}\, \tilde{\mathbf{y}}_{L_k,k,m}\, \xi^*_{L_k,k,m}$$

end

end

By applying this algorithm to adaptively L_k-tap PTEQ $\hat{\mathbf{p}}_{L_k,k,m}$, an active tone has a L_k-tap PTEQ which its input is the complex sliding-FFT output for that tone and $L_k - 1$ difference terms. The upper triangular matrix $\mathbb{U}_{k,m}$ is stored and updated that can be used to adapt the inverse autocorrelation matrix $\mathbb{Q}_{L_k,k,m}$ for every symbol.

The update of weight-vector $\hat{\mathbf{p}}_{L_k,k,m}$ with variable tap-length L_k at symbol k and tone m is also performed separately for each tone and every symbol by means of adaptive inverse autocorrelation matrix.

6. ANALYSIS OF VARIABLE TAP-LENGTH ALGORITHM

We analyse the convergence of proposed adaptive pseudo-fractional tap-length algorithm which is introduced in Section 4.5. According to $L_k - \Delta$ and L_k, an pseudo-fractional tap-length parameter $\mathcal{L}_{f_k}$ is adjusted adaptively by means of the difference of square of prediction mixed-tone errors with the different tap-length.

By taking expectations on both sides of (36), we then arrive at

$$E\{\mathcal{L}_{f_{k+1}}\} = \lambda\, E\{\mathcal{L}_{f_k}\} + \beta\, E\big\{(\tilde{\zeta}_{\bar{L}_k-\Delta,k}) - (\tilde{\zeta}_{\bar{L}_k,k})\big\}\ , \tag{39}$$

where $\bar{L}_k = E\{L_k\}$, which is an average tap-length and

$$\tilde{\zeta}_{\bar{L}_k,k} = E\{|\tilde{\xi}_{\bar{L}_k,k}|^2\}\ , \tag{40}$$

where $\tilde{\zeta}_{\bar{L}_k,k}$ is the mean square mixed-tone error with the average tap-length $\bar{L}_k$.

It is observed from (39), if $L(0) = \min\{L_k\}$ and Δ is larger than the width of the suboptimum tap-length, L_k keeps increasing until

$$\tilde{\zeta}_{\bar{L}_k-\Delta,k} - \tilde{\zeta}_{\bar{L}_k,k} \le \epsilon\ . \tag{41}$$

On the other hand, if $L(0)$ is too large, $\bar{L}_k$ keeps decreasing until

$$\tilde{\zeta}_{\bar{L}_k-\Delta,k} - \tilde{\zeta}_{\bar{L}_k,k} \ge \epsilon\ . \tag{42}$$

It is noted that if $\delta = 1$ in (37), $\bar{L}_k$ can converge to the optimum tap-length L_{opt}. Generally, if $\delta \neq 1$, $\bar{L}_k$ will converge to a range of $L_{opt} - \delta$ and $L_{opt} + \delta$ as described in [11].

We study the steady-state performance and assume that the system has arrived at steady-state when $k \to \infty$. Therefore, the steady-state of tap-length will converge to a fixed value as [12]

$$L_{opt} \le L(\infty) \le L_{opt} + \Delta\ . \tag{43}$$

As presented in [19], the initial parameters should be predicted in order to obtain a small fluctuation of the tap-length at the steady-state condition as

$$\lfloor L_{opt} - \delta \rfloor + 1 \le L(\infty) \le \lfloor L_{opt} + \delta \rfloor + 1\ , \tag{44}$$

where $\delta < 1$.

7. SIMULATION RESULTS

In this section, the ADSL downstream simulations with the carrier serving area (CSA) loop #1 and #5 were the representative of simulations with all 8 CSA loops detailed in [20] as follows. The CSA#1 loop is a 7700 ft, 26-gauge loop with 26-gauge bridged tap of length of 600 ft at 5900 ft. The CSA#5 is consisting of 26-gauge bridged tap of length of 1200 ft at 5800 ft and of 24 and 26-gauge loop of length of 150 ft at 5950 ft and 1200 ft at 7150 and of 24 and 26-gauge loop of 300 ft at 7450 ft and 300 ft at 7750 ft.

We implemented transmission simulations for the ADSL-based downstream including additive white Gaussian noise (AWGN) and near-end crosstalk (NEXT) detailed as follows. The used tones for downstream transmission were starting at active tones 38 to 255 and unused tones including tones 8 to 32 for upstream transmission were set to zero. The samples of reference CSA loop were used for the entire test channel, which comprises 512 coefficients of channel impulse response.

Other parameters were as the sampling rate $f_s = 2.208$ MHz and the size of FFT $N = 512$. The length of CP (ν) was identical to 32. The synchronisation delay was of 45. The SNR gap of 9.8dB, the coding gain of 4.2dB, the noise margin of 6 dB, and the input signal power of -40 dBm/Hz were used for all active tones [21]. With the power of AWGN of -140dBm/Hz and NEXT from 24 ADSL disturbers were included in the test channel. The bit allocation calculation requires an estimate of SNR on tone $n \in N_d$, when the noise energy is estimated after per-tone equalisation.

We compare the proposed MTiQR-RLS PTEQ with variable tap-length method and conventional complex RLS [22] PTEQ. The proposed MTiQR-RLS PTEQ and RLS PTEQ were initialised with $\hat{\mathbf{p}}_0 = [1\,0\,0\ldots\,0]^T$ and $\mathbb{Q}_0 = \varsigma \cdot \hat{\mathbf{I}}_{L_k}$, where $\varsigma = 0.095$. The matrix $\hat{\mathbf{I}}_{L_k}$ is the $L_k \times L_k$ identity matrix. Other parameters of RLS PTEQ were $\tilde{\mathbf{k}}_0 = [1\,0\,0\ \ldots\ 0]^T$ and of proposed MTiQR-RLS PTEQ were $\Delta = 4$, $L(0) = \mathcal{L}_f(0) = 4$ and 10, $\delta = 0.025$ and $\beta = 1$.

The forgetting-factor parameter λ of adaptive RLS PTEQ was fixed at 0.99 and the proposed MTiQR-RLS PTEQ was increased from $\lambda = 0.999$ during between the active tones at 38 to 127, to $\lambda = 0.97$ for the remaining updated active tones. The number of tap-length for underbounding α and overbounding ω of proposed MTiQR-RLS were fixed at $\alpha = 4$ and $\omega = 32$. The number of tap-length of

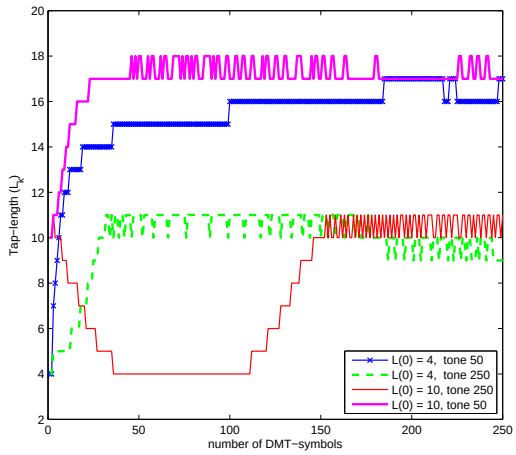

Fig.3: *Trajectories of tap-length adaptation of proposed MTiQR-RLS PTEQ with the initial tap-length parameters* $L(0) = 4$ *and 10 of CSA loop* #1.

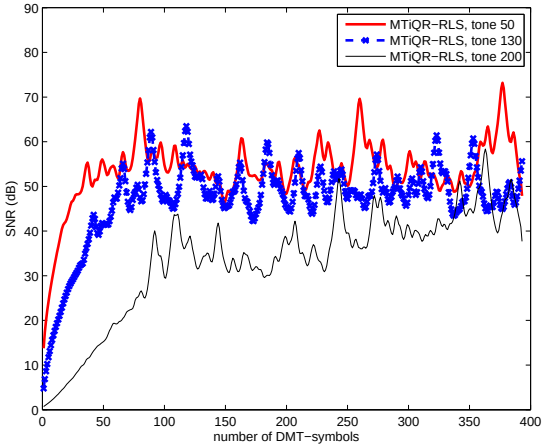

Fig.5: *Learning curves of SNR performance of proposed MTiQR-RLS PTEQ using* $L(0) = \mathcal{L}_f(0) = 4$ *with the different tones of CSA loop* #1.

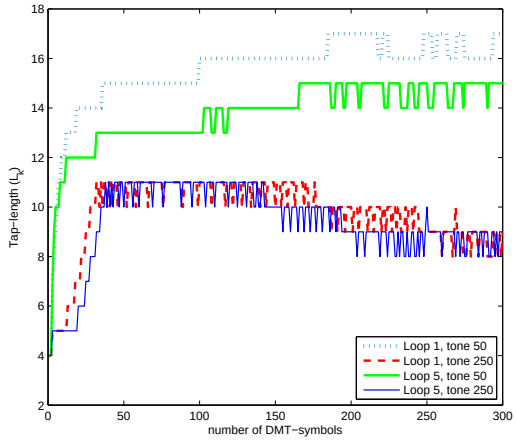

Fig.4: *Trajectories of tap-length parameters of proposed MTiQR-RLS PTEQ using* $L(0) = 4$ *at tone 50 and 250 of CSA loop* #1 *and* #5.

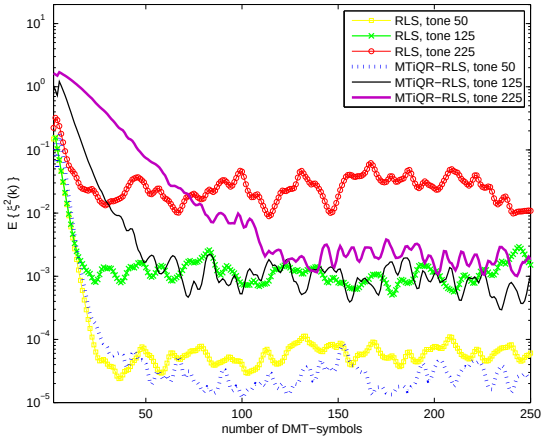

Fig.6: *Sum of squared mixed-tone errors curves of proposed MTiQR-RLS and RLS PTEQs using* $L(0) = \mathcal{L}_f(0) = 4$ *with the different tones of CSA loop* #1.

RLS PTEQ was set to 32. Fig. 3 shows the learning curves of trajectories of variable tap-length L_k at two different initialisation of $L(0)$ of proposed MTiQR-RLS PTEQ at active tone 50 and 250 for the samples of CSA loop #1. It is noted that L_k converges automatically to around the optimum tap-length with the samples of two different values of initial variable tap-length $L(0) = 4$ and 10 and of initial parameter $\Delta = 4$, respectively. We note that the steady-state tap-length $L(\infty)$ fluctuates slightly around $\lfloor L_{opt} - \delta \rfloor + 1 \leq L(\infty) \leq \lfloor L_{opt} + \delta \rfloor + 1$, where $\delta < 1$, corresponding to Section 6. Fig. 4 illustrates the trajectories of tap-length L_k of proposed MTiQR-RLS PTEQ for the samples of active tones at 50 and 250 with the fixed initial tap-length at $L(0) = 4$ of CSA loop #1 and #5. It is shown that the tap-length of the each tone can adjust adaptively to its own equilibrium.

Fig. 5 shows the learning curves of signal to noise ratio (SNR) performance of proposed MTiQR-RLS PTEQ with the different active tones at 50, 130 and 200 for the samples of CSA loop #1. These results indicate that the proposed MTiQR-RLS PTEQ converge to steady-state at approximate 150 symbols of all sampled tones. Fig. 6 reveals the sum of squared mixed-tone errors curves of the proposed MTiQR-RLS and RLS algorithms for the sample of CSA loop #1 of active tones as 50, 125 and 225. It is noticed that the proposed MTiQR-RLS algorithm can converge to steady-state conditions as slower rate than RLS algorithm with the lower errors shown at tone 225. Fig. 7 depicts the learning curves of trajectories of variable tap-length L_k at two different initialisation of $L(0)$ of proposed MTiQR-RLS PTEQ for the

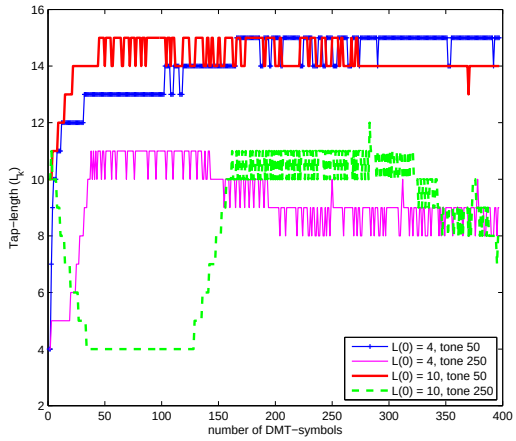

Fig.7: *Trajectories of tap-length adaptation of proposed MTiQR-RLS PTEQ with the initial tap-length* $L(0) = 4$ *and 10 of CSA loop* #5.

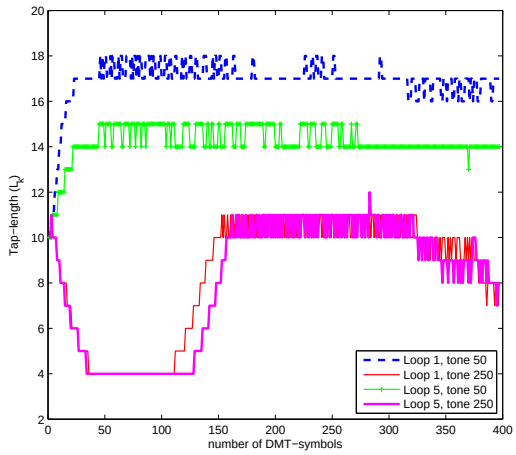

Fig.8: *Trajectories of tap-length of proposed MTiQR-RLS PTEQ using* $L(0) = 10$ *at tone 50 and 250 of CSA loop* #1 *and* #5.

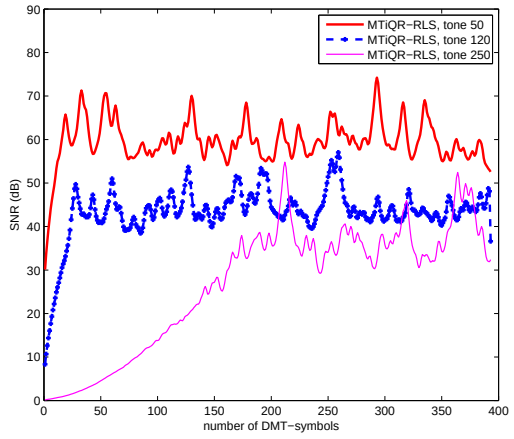

Fig.9: *Learning curves of SNR performance of proposed MTiQR-RLS PTEQ using* $L(0) = \mathcal{L}_f(0) = 4$ *with the different tones of CSA loop* #5.

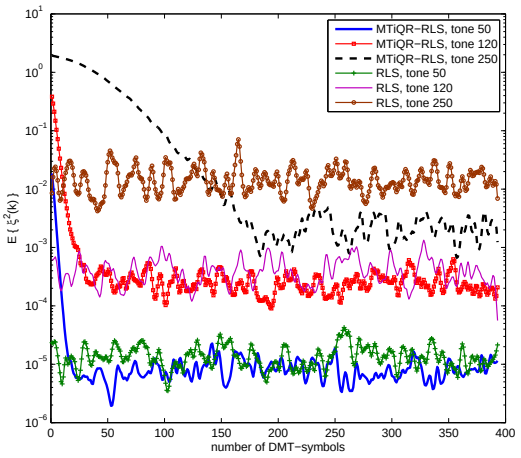

Fig.10: *Sum of squared mixed-tone errors curves of proposed MTiQR-RLS and RLS PTEQs using* $L(0) = \mathcal{L}_f(0) = 4$ *with the different tones of CSA loop* #5.

samples of CSA loop #5. Fig. 8 shows the trajectories of tap-length L_k of proposed MTiQR-RLS PTEQ for the samples of active tones at 50 and 250 with the fixed initial tap-length at $L(0) = 10$ of CSA loop #1 and #5.

Fig. 9 illustrates the learning curves of signal to noise ratio (SNR) performance of proposed MTiQR-RLS PTEQ with the different active tones at 50, 120 and 250 for the samples of CSA loop #5. Fig. 10 shows the learning curves of sum of squared mixed-tone errors curves of the proposed MTiQR-RLS algorithm as compared with RLS algorithm for the sample of CSA loop #5 of tones as 50, 120 and 250. We notice that the RLS algorithm converge rapidly to steady-state condition with higher error rate, while the proposed MTiQR-RLS PTEQ can achieve lower error rate than RLS algorithm shown at tone 225.

8. CONCLUSION

In this paper, we present the proposed MTiQR-RLS PTEQ design with variable tap-length algorithm based on the mixed-tone cost funtion. With the proposed variable tap-length mechanism, the low complexity solutions for per-tone equalisation based on RLS algorithm can be achieved. The proposed MTiQR-RLS algorithm and proposed variable tap-length method are derived by means of minimising the mixed-tone cost function as the sum of weight-estimated mixed-tone errors with the adaptive tap-length at each iteration. The trajectories of tap-length parameters are also shown to converge to equilibrium with different initial parameters. The performance analysis of proposed variable tap-length algorithm at the steady-state is introduced that can predict the optimum tap-length in order to obtain

a small fluctuation of tap-length. Simulation results with several ADSL-based parameters show that the proposed MTiQR-RLS PTEQ is able to improve the signal to noise ratio performance as compared with RLS-PTEQ for the samples of CSA loops.

References

[1] R.Baldemair, and P.Frenger, "A Time-domain Equalizer Minimizing Intersymbol and Intercarrier Interference in DMT Systems", in *Proc. of IEEE Global Commun. Conf. (GLOBECOM'01)*, pp.381-385, Nov. 2001.

[2] S.Sitjongsataporn, and P.Yuvapoositanon, "An Adaptive Step-size Order Statistic Time Domain Equaliser for Discrete Multitone Systems", in *Proc. of IEEE Int. Symp. on Circuits and Syst. (ISCAS'07)*, New Orleans, USA, pp.1333-1336, May 2007.

[3] T.Pollet, M.Peeters, M.Moonen, and L.Vandendorpe, "Equalization for DMT-Based Broadband Modems", *IEEE Commun. Magazine*, pp. 106-113, May 2000.

[4] K.V.Acker, G.Leus, M.Moonen, O.van de Wiel, and T.Pollet, "Per Tone Equalization for DMT-based Systems", *IEEE Trans. on Commun.*, vol. 49, no. 1, pp. 109-119, Jan. 2001.

[5] K.V.Acker, G.Leus, M.Moonen, and T.Pollet, "RLS-Based Initialization for Per-Tone Equalizers in DMT Receivers", *IEEE Trans. on Commun.*, vol. 51, no. 6, pp. 885-889, Jun. 2003.

[6] S.Sitjongsataporn and P.Yuvapoositanon, "A Mixed-Tone RLS Algorithm with Orthogonal Projection for Per-Tone DMT Equalisation", in *Proc. of IEEE Int. Midwest Symp. on Circuits and Syst. (MWSCAS'08)*, Knoxville, Tennesse, USA,pp. 942-945, Aug. 2008.

[7] F.Yu and M.Bouchard, "Recursive Least-Squares Algorithms with Good Numerical Stability for Multichannel Active Noise Control", in *Proc. of IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP'01)*, vol. 5, pp. 3221-3224, May 2001.

[8] A.S.Morris, and S.Khemaissia, "A Fast New Algorithm for A Robot Neurocontroller using Inverse QR Decomposition", *UKACC Int. Conf. on Control (CONTROL'98)*, vol. 1, pp. 751-756, Sep. 1998.

[9] M.Moonen and J.G.McWhirter, "Systolic Array for Recursive Least Squares by Inverse Updating", *Electronics Letters*, vol. 29, no. 13, pp. 1217-1218, June 1993.

[10] Z.Pritzker and A.Feuer, "Variable Length Stochastic Gradient Algorithm", *IEEE Trans. on Signal Process.*, vol. 39, no. 4, pp. 997-1001, Apr. 1991.

[11] Y.Gong and C.F.N.Cowan, "An LMS Style Variable Tap-Length Algorithm for Structure Adaptation", *IEEE Trans. on Signal Process.*, vol. 53, no. 7, pp. 2400-2407, July 2005.

[12] Y.Zhang, N.Li, J.A.Chambers and A.H.Sayed, "Steady-State Performance Analysis of a Variable Tap-Length LMS Algorithm", *IEEE Trans. on Signal Process.*, vol. 56, no. 2, pp. 839-845, Feb. 2008.

[13] N.Li, Y.Zhang, Y.Zhao and Y.Hao, "An Improved Variable Tap-Length LMS Algorithm", *Signal Process.*, vol. 89, pp. 908-912, 2009.

[14] G.Strang, *Linear Algebra and its Applications*, Harcourt Brace Jovanovich, 1988.

[15] S.Haykin, *Adaptive Filter Theory*, Prentice Hall, 1996.

[16] S.T.Alexander and A.L.Ghirnikar, "A Method for Recursive Least Squares Filtering Based Upon an Inverse QR Decomposition", *IEEE Trans. on Signal Process.*, vol. 41, no. 1, Jan. 1993.

[17] A.H.Sayed and T.Kailath, "A State-Space Approach to Adaptive RLS filtering", *IEEE Signal Process. Magazine*, vol. 11, pp. 18-60, 1994.

[18] S.Sitjongsataporn and P.Yuvapoositanon, "Adaptive Forgetting-factor Gauss-Newton inverse QR-RLS Per-Tone Equalisaiton for Discrete Multitone Systems", in *Proc. of Int. Conf. on Elect. Eng./Electron., Comput., Telecommun. and Inform. Technology (ECTI-CON'08)*, Krabi, Thailand, pp. 561-564, May 2008.

[19] S.Sitjongsataporn and P.Yuvapoositanon, "Variable Tap-Length RLS-based Per-Tone Equalisation for DMT-based systems", in *Proc. of* 33^{rd} *Elect. Eng. Conf. (EECON-33)*, Chiang Mai, Thailand, pp. 1333-1336, Dec. 2010.

[20] N.Al-Dhahir and J.M.Cioffi, "Optimum Finite-Length Equalization for Multicarrier Transceivers", *IEEE Trans. on Commun.*, vol. 44, pp. 56-64, Jan. 1996.

[21] International Telecommunications Union (ITU). Recommendation G.996.1, *Test Procedures for Asymmetric Digital Subscriber Line (ADSL) Transceivers*, Feb. 2001.

[22] P.S.R.Diniz, *Adaptive Filtering Algorithms and Practical Implementation*, Springer, Boston, MA, 2008.

Suchada Sitjongsataporn received the B.Eng. and D.Eng. degrees of Electrical Engineering from Mahanakorn University of Technology, Bangkok, Thailand in 2002 and 2009, respectively. She has worked as lecturer at department of Electronic Engineering, Mahanakorn University of Technology, since 2002. Currently, she is an Assistant Professor of Electronic Engineering in Mahanakorn University of Technology. Her research interests are in the area of adaptive algorithm, adaptive equalisation and adaptive signal processing for wireline and wireless communications.