Fast Computation of a Constrained Energy DFE

Ricardo Merched
Dept. of Electrical and Computer Engineering
Universidade Federal do Rio de Janeiro, Brazil
Email: merched@lps.ufrj.br

***Abstract*— Constrained formulations of decision feedback equalizer (DFE) schemes arise whenever its intrinsic error propagation phenomenon must be reduced. In this context, the solution to the constrained energy DFE is shown to be closely related to the so-called Kalman gain vector usually observed in recursive least-squares algorithms, except that here the desired shift structure that allows for a fast algorithm no longer exists. Still, in this paper we show how to properly correct for this discrepancy and provide a fast recursion for computing the constrained DFE coefficients.**

I. INTRODUCTION

It is well known that minimum mean squared (MMSE) decision feedback equalizer (DFE) schemes exhibits an inherent error propagation phenomenon, due to the assumption of correct decisions in its formulation. One way to diminish the effect of wrong decisions is to pose appropriate constraints on the feedback filter, either via energy or magnitude limiting criteria [1],[2]. As a result, the DFE performance in the former case can be shown to achieve a controlled tradeoff between error propagation and noise enhancement effects, so that further requirements on error control coding can be relaxed.

Now, in channel based equalization schemes, a fast algorithm for computing the equalizer coefficients is crucial. For this purpose, we have recently introduced a new approach for computing the optimum DFE coefficients which bypasses several structural and complexity difficulties inherent to prior arts (see [3] and the references therein). As a fallout, the solution to the feedforwad coefficients in this case can be interpreted as the so-called *Kalman gain*, which in turn can be efficiently computed via well known fast transversal RLS techniques [5]. This is a consequence of formulating the DFE cost function as a linear estimation problem, as opposed to a constrained cost. In this case, the feedback filter can be obtained simply by resorting to fast convolution techniques.

In the constrained energy DFE considered in this paper, the optimal feedforward filter can be similarly shown to depend on the Kalman gain, except that here the desired shift structure leading to an imediate fast algorithm implementation (as in the unconstrained DFE case) no longer exists. Still, we shall show how to compensate for this discrepancy and provide an efficient algorithm to compute the optimum DFE filters in a SISO scenario. This is accomplished in 3 steps:

1) First, we identify which variables among the Kalman gain recursions are intimately affected when shift structure suddenly breaks off;
2) Secondly, we show how to properly correct these variables so that the corresponding Kalman gain vector can be efficiently propagated;
3) Finally, we provide additional recursions in order to obtain the feedforwad filter from the Kalman gain vector. The feedback filter can be easily obtained via fast and stable convolution techniques.

Applications of the proposed method include fast varying channel estimate based structures in communication systems, as well as magnetic recording channels.

II. FINITE-LENGTH DFE FORMULATION

The constrained energy DFE cost function is a regularized formulation of the conventional DFE problem. That is, by collecting the tap coefficients of both $T(z)$ and $B(z)$ of Fig. 1 into vectors, say, $\{t, b\}$, we aim to minimize[1]

$$\xi = \mathsf{E}|x(n-\delta) - \hat{x}(n-\delta)|^2 \;+\; b^* A b\,, \qquad (1)$$

where $\hat{x}(n-\delta)$ is the delayed input signal estimate prior to the decision, and $A = \text{diag}\{\ell_0, \ell_1, \ldots, \ell_{L-1}\}$, represents the energy constraint.

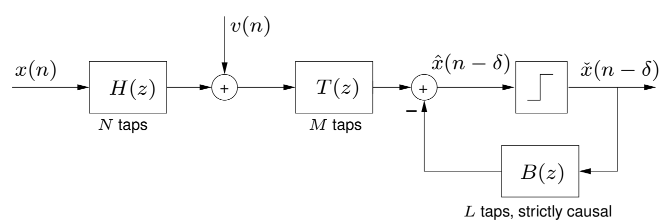

Fig. 1. Discrete symbol-spaced scalar DFE model.

Thus for a unit power i.i.d. input sequence, and noise autocorrelation matrix R_v, the minimization of (1) (assuming correct decisions) leads to the following solution:

$$\boxed{t_{\text{opt}} = (R_v + H^* W H)^{-1} h_\delta^*} \qquad (2)$$

$$\boxed{b_{\text{opt}} = (I + A)^{-1} \bar{H} t_{\text{opt}}} \qquad (3)$$

For example, $\delta = M-1$ (usually set for various practical channel and noise scenarios with fairly long feedforward filters), yields the $(N+M-1)\times M$ matrix

[1]We denote * as complex conjugate transposition.

$$H = \begin{bmatrix} h(0) & 0 & \cdots & 0 \\ h(1) & h(0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h(N-1) & h(N-2) & \ddots & h(0) \\ 0 & h(N-1) & \ddots & h(1) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h(N-1) \end{bmatrix},$$

a weighting matrix W, and

$$\bar{H} = \begin{bmatrix} 0 & h(N-1) & \ddots & h(1) \\ 0 & 0 & \ddots & h(2) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h(N-1) \end{bmatrix}, \quad h_\delta^* = \begin{bmatrix} h^*(N-1) \\ h^*(N-2) \\ \vdots \\ h^*(0) \end{bmatrix}$$

III. Fast Computation of the Feedforward Filter

Let us define the coefficient matrix

$$P_{N+M-1} \triangleq (R_v + H^* W H)^{-1} .$$

where $W \triangleq \text{diag}\{\mu_0, \mu_1, \ldots, \mu_N\}$ is a general weighting matrix, with $\mu_i = (1 - 1/\ell_i)$. The optimal solution for the feedforward coefficients is then given by

$$t_{\text{opt}} \triangleq t_{N+M-1} = P_{N+M-1} h_\delta^* . \tag{4}$$

This quantity is closely related to the definition of the Kalman gain vector used to update the optimal solution in a certain regularized weighted RLS problem. More specifically, given an $(i+1)\times M$ data matrix H_i and its corresponding coefficient matrix $P_{M,i}$, the (normalized) Kalman gain vector defined as $k_{M,i} = P_{M,i-1} h_{M,i}^*$ can be time-updated according to the following recursions (see, e.g., [6])[2]:

$$\begin{aligned} \gamma_M^{-1}(i) &= \mu_i^{-1} + h_{M,i} P_{M,i-1} h_{M,i}^* , & (5) \\ k_{M,i} &= P_{M,i-1} h_{M,i}^* , & (6) \\ P_{M,i} &= P_{M,i-1} - k_{M,i} \gamma_M(i) k_{M,i}^* , & (7) \end{aligned}$$

with the initial condition $P_{M,-1} = R_v^{-1}$. Then, at the i-th iteration, we have

$$t_{M,i} = P_{M,i} h_\delta^* .$$

Moreover, by associating $P_{M,i}$ and $k_{M,i}$ with a forward estimation problem, its corresponding optimal solution, viz., $w_{M,i}^f$, is recursively obtained as follows:

$$w_{M,i}^f = w_{M,i-1}^f + k_{M,i} f_M(i)$$

where $f_M(i)$ is the corresponding *a posteriori* forward residual. The minimum cost of this problem is updated as

$$\xi_M^f(i) = \xi_M^f(i-1) + \gamma_M(i) |\alpha_M(i)|^2$$

where $\alpha_M(i)$ is the *a priori* error related to $f_M(i)$ as

$$f_M(i) = \gamma_M(i) \alpha_M(i) / \mu_i .$$

Note that analogous relations hold similarly for the backward estimation problem, with the quantities $\{w_{M,i}^f, f_M(i), \alpha_M(i)\}$ replaced by $\{w_{M,i}^b, b_M(i), \beta_M(i)\}$. Now, multiplying Eq. (7) from the right by h_δ^*, the optimal feedforward filter given by (4) can be efficiently computed as

$$\boxed{t_{M,i} = t_{M,i-1} - r_M^*(i) \gamma_M(i) k_{M,i}} \tag{8}$$

with initial condition $t_{M,\delta} = \gamma_M(\delta) k_{M,\delta}$, and where we have defined the quantity

$$\boxed{r_M(i) \triangleq h_{M,\delta} k_{M,i} = h_{M,\delta} P_{M,i-1} h_{M,i}^*} \tag{9}$$

Therefore, our main goal in the sequel is to obtain efficient recursive relations for the variables $\{k_{M,i}, \gamma_M(i)\}$ and $r_M(i)$ by relying on order updates and order downdates.

A. Fast Recursions for Weighted Problems

First, observe that a fast method for computing $k_{M,i}$, and hence $\gamma_M(i)$, is not possible for general weighted problems. To see this, let $P_{M+1,i-1}$ be the augmented Ricatti variable (via forward prediction) at time $i-1$. Likewise, let $\breve{P}_{M+1,i}$ be the corresponding augmented matrix (via backward prediction) at time i. Hence when $W_{i-1} = W_i$, we have that

$$P_{M+1,i-1} = \breve{P}_{M+1,i}, \tag{10}$$

so that the corresponding Kalman vector can be efficiently time updated via consecutive order update and order downdate steps:

$$\underbrace{k_{M,i-1} \rightarrow k_{M+1,i-1}}_{\text{(1) order update}} = \underbrace{\breve{k}_{M+1,i} \rightarrow k_{M,i}}_{\text{(2) order downdate}} \tag{11}$$

(note that, at time i we work with two vectors, i.e., $k_{M,i-1}$ and $k_{M,i}$, whereas the index i refers to the correponding $i-1$ and i-th data matrix rows). Here, due to the different tap weighting α_i employed, we have $W_{i-1} \neq W_i$, so that, in general, $k_{M+1,i-1} \neq \breve{k}_{M+1,i}$. Now for the constrained problem of this paper, we shall consider constant weights $\mu_i = \mu$, which implies that W_N assumes the following structure:

$$W_N = \begin{bmatrix} I_{\delta+1} & \\ & \mu I_{N+M-\delta-2} \end{bmatrix}$$

This means that up to iteration $i = \delta + 1$, we have, from the usual fast Kalman recursions, that $k_{M+1,i-1} = \breve{k}_{M+1,i}$. Since exactly at time $i = \delta + 1$ the underlying data matrix shift structure is interrupted[3], the question is then how to relate $\{\bar{k}_{M+1,i-1}, \breve{k}_{M+1,i}\}$, and hence $\{\gamma_{M+1}(i-1), \breve{\gamma}_{M+1}(i)\}$, from time $i = \delta + 2$ onwards.

[2]We indicate the order M in addition to the time index i (also refering to the i-th row), since we will be dealing with order recursive relations.

[3]Observe that even though row $h_{\delta+1}$ is weighted by α, it still holds that $k_{M+1,\delta} = \breve{k}_{M+1,\delta+1}$.

B. Correction for $k_{M+1,i}$

Suppose that we have computed $k_{M,i} \triangleq P_{M,i-1}h^*_{M,i}$ at time $i \geq \delta+1$, so that now we wish to compute $k_{M,i+1} \triangleq P_{M,i}h^*_{M,i+1}$ according to steps (1) and (2) in Eq. (11). Moreover, let $P_{M+1,i-1}$ be the order updated Ricatti variable at time $i \geq \delta+1$. In this case, we may write

$$\breve{P}^{-1}_{M+1,i} = P^{-1}_{M+1,i-1} \ + \ (\mu-1)h^*_{M+1,\delta}h_{M+1,\delta}$$

so that inverting this relation we arrive at

$$\breve{P}_{M+1,i} = P_{M+1,i-1} \ - \ \frac{P_{M+1,i-1}h^*_{M+1,\delta}h_{M+1,\delta}P_{M+1,i-1}}{\frac{1}{\mu-1} + h_{M+1,\delta}P_{M+1,i-1}h^*_{M+1,\delta}} \, . \tag{12}$$

Thus multiplying both sides of (12) by $\breve{h}_{M+1,i+1} = h_{M+1,i}$, we get

$$\boxed{\breve{k}_{M+1,i+1} = k_{M+1,i} \ - \ r_{M+1}(i)\gamma^h_{M+1}(i)t_{M+1,i-1}} \tag{13}$$

where we have defined the following variables:

$$\begin{aligned} t_{M+1,i-1} &\triangleq P_{M+1,i-1}h^*_{M+1,\delta} && (14) \\ r_{M+1}(i) &\triangleq h_{M+1,\delta}P_{M+1,i-1}h^*_{M+1,i} && (15) \\ &= h_{M+1,\delta}k_{M+1,i} = t^*_{M+1,i-1}h^*_{M+1,i} && (16) \\ \gamma^h_{M+1}(i) &\triangleq \frac{1}{\frac{1}{\mu-1} + h_{M+1,\delta}P_{M+1,i-1}h^*_{M+1,\delta}} && (17) \end{aligned}$$

A quick look onto Eq. (13) implies that 3 additional vector recursions would be needed in order to obtain $\breve{k}_{M+1,i+1}$. That is, one time-update for $t_{M+1,i-1}$ [similar to (8)], its product with $r_{M+1}(i)\gamma^h_{M+1}(i)$, and the inner product in (15). However, this can still be acomplished by using only 2 additional vector recursions, via an order update for $t_{M,i-1}$. To see this, first consider the order update for the Kalman gain:

$$k_{M+1,i} = \begin{bmatrix} 0 \\ k_{M,i} \end{bmatrix} + \frac{\alpha^*_M(i)}{\xi^f_M(i-1)} \begin{bmatrix} 1 \\ -w^f_{M,i-1} \end{bmatrix} \tag{18}$$

Likewise, consider the order update for $P_{M+1,i-1}$:

$$P_{M+1,i-1} = \begin{bmatrix} 0 & 0 \\ 0 & P_{M,i-1} \end{bmatrix} + \frac{1}{\xi^f_M(i-1)} \begin{bmatrix} 1 \\ -w^f_{M,i-1} \end{bmatrix} [\cdot]^* \tag{19}$$

(the notation $[\cdot]$ signifies "repeat the previous term"). Substituting (19) into Eq. (14), we get

$$t_{M+1,i-1} = \begin{bmatrix} 0 \\ t_{M,i-1} \end{bmatrix} + \frac{\alpha^{h*}_M(i-1)}{\xi^f_M(i-1)} \begin{bmatrix} 1 \\ -w^f_{M,i-1} \end{bmatrix} \tag{20}$$

where we have defined

$$\alpha^h_M(i-1) \triangleq h_{M+1}(\delta+1) - h_{M,\delta}w^f_{M,i-1} \tag{21}$$

For compactness of notation, we denote the quantity

$$s_{M+1}(i) \triangleq r_{M+1}(i)\gamma^h_{M+1}(i) \, , \tag{22}$$

so that substituting Eqs. (18) and (20) into (13), we obtain

$$\begin{aligned} \breve{k}_{M+1,i+1} \ = \ & \begin{bmatrix} 0 \\ k_{M,i} - s_{M+1}(i)t_{M,i-1} \end{bmatrix} \\ & + \frac{\alpha^*_M(i) - s_{M+1}(i)\alpha^{h*}_M(i-1)}{\xi^f_M(i-1)} \begin{bmatrix} 1 \\ -w^f_{M,i-1} \end{bmatrix} \end{aligned} \tag{23}$$

Now, further definition of the following auxiliary quantities

$$\begin{aligned} k'_{M,i} &\triangleq k_{M,i} - r_{M+1}(i)\gamma^h_{M+1}(i)t_{M,i-1} && (24) \\ \tau_M(i) &\triangleq \alpha^*_M(i) - r_{M+1}(i)\gamma^h_{M+1}(i)\alpha^{h*}(i-1) \end{aligned}$$

allows us to write (23) more compactly as

$$\boxed{\breve{k}_{M+1,i+1} = \begin{bmatrix} 0 \\ k'_{M,i} \end{bmatrix} + \frac{\tau_M(i)}{\zeta^f_M(i-1)} \begin{bmatrix} 1 \\ -w^f_{M,i-1} \end{bmatrix}}$$

Also, note from recursions (22) and (23), that we still need to obtain fast recursions for the scalar variables $\{r_{M+1}(i), \gamma^h_{M+1}(i), \alpha^h_M(i)\}$. We do so as follows.

(*i*) *Recursion for $r_{M+1}(i)$*. Assuming that we have already computed $r_M(i)$ via the inner product in (9), we can obtain $r_{M+1}(i)$ by substituting (18) into (16) to get

$$\boxed{r_{M+1}(i) = r_M(i) + \frac{\alpha^*_M(i)\alpha^h_M(i-1)}{\xi^f_M(i-1)}}$$

(*ii*) *Recursion for $\gamma^h_{M+1}(i)$*. Denote $\gamma^{-h}_{M+1}(i) \triangleq \left[\gamma^h_{M+1}(i)\right]^{-1}$. Then, replacing M by $M+1$ into the time update of (7), and substituting the resulting recursion into (17), we obtain

$$\boxed{\gamma^{-h}_{M+1}(i) = \gamma^{-h}_{M+1}(i-1) - \gamma_{M+1}(i-1)|r_{M+1}(i-1)|^2} \tag{25}$$

The initial condition of this variable (which can be set at the transition time $\delta+1$) is obtained from its definition in (17), which gives $\gamma^{-h}_{M+1}(\delta+1) = 1 - \gamma_{M+1}(\delta) + (\mu-1)^{-1}$.

(*iii*) *Recursion for $\alpha^h_M(i)$*. Consider the time update of the forward optimal weight vector $w^f_{M,i-1}$, i.e.,

$$w^f_{M,i} = w^f_{M,i-1} + k_{M,i}f_M(i)$$

Multiplying this relation from the left by $h_{M,\delta}$ and substituting the result into (21) yields

$$\boxed{\alpha^h_M(i) = \alpha^h_M(i-1) - r_M(i)f_M(i) \, , \quad \alpha^h_M(\delta) = f_M(\delta)}$$

C. Correction for $\gamma_{M+1}(i)$

Finally, note that at time $i = \delta+1$ it holds that

$$\breve{\gamma}_{M+1}(\delta+1) = \gamma^{-1}_{M+1}(\delta) - 1 + 1/\mu \, .$$

It is part of the initialization of all variables described so far, and accomodates the transition in structure (to now scaled rows) inherent to this quantity.

Now, a correction recursion for $\gamma_{M+1}(i)$, for $i \geq \delta+2$, can be obtained from Eq. (13), by multiplying it from the left by $h_{M+1,i}$. This gives

$$\boxed{\breve{\gamma}^{-1}_{M+1}(i) \ = \ \gamma^{-1}_{M+1}(i-1) \ - \ \gamma^h_{M+1}(i-1)|r_{M+1}(i-1)|^2}$$

Also, combining the above relation with (25) we obtain the following alternative recursion for $\breve{\gamma}^{-1}_{M+1}(i)$:

$$\breve{\gamma}^{-1}_{M+1}(i) = \frac{\gamma^{-h}_{M+1}(i)}{\gamma^{-h}_{M+1}(i-1)\gamma_{M+1}(i-1)}$$

IV. Initial Conditions

It has been shown in [5] that in order for the fast transversal filter (FTF) recursions to hold in the case of a general structure for R_v, one must select certain initial quantities $\{p, q, \epsilon, \eta\}$ such that the following condition is satisfied (see [5]):

$$\begin{bmatrix} R_v & p \\ p^* & \epsilon \end{bmatrix} = \begin{bmatrix} \eta & q^* \\ q & R_v \end{bmatrix}, \text{ for some } p, q. \tag{26}$$

Hence, in the case of a Toeplitz covariance matrix R_v, whose first row is defined by $\begin{bmatrix} c_0 & c_1 & \cdots & c_{M-1} \end{bmatrix}$, we obtain $\eta = \epsilon = c_0$, and

$$\begin{aligned} p &= \begin{bmatrix} 0 & c_{M-2} & \cdots & c_1 & c_0 \end{bmatrix}^T \\ q &= \begin{bmatrix} c_1^* & c_2^* & \cdots & c_{M-1}^* & 0 \end{bmatrix}^T \end{aligned}$$

The role of the quantities above is to define the initial conditions $\{\zeta_M^f(-2), \zeta_M^b(-1), w_{M,-2}^f, w_{M,-1}^b\}$ as explained in [5]. This completes the fast recursions for computing the feedforward filter coefficients, which are listed in Table 1[4].

V. Computational Complexity

We see that compared to a fast algorithm that propagates the Kalman vector in $N+M-1$ iterations, the new algorithm requires three additional $\mathcal{O}(M)$ recursions. These recursions are given by Eqs. (8), (9) and (24), so that the complexity per iteration amounts to $\mathcal{O}(8M)$ multiplies plus 13 scalar recursions, resulting in $\mathcal{O}(8(N+M-1)M+13N)$ multiplications.

The complexity of computing the feedback filter depends on the method chosen for fast convolution of the matrix $\bar{H}$ with the optimal computed feedforward filter t_{opt}. Thus for an FFT-based filtering, the complexity is the one of obtaining the FFT of two vectors of K elements each [where K is the smallest power-of-two integer larger than or equal to $(L+M)$], the inverse FFT of another K size vector, and K complex multiplies. Thus the overall complexity for the FBF amounts to $2L + 6L\log_2(2L)$.

VI. Conclusion

We have proposed a fast procedure for computing constrained energy MMSE-DFE coefficients, in which case no efficient algorithm was available. In contrast to a recently proposed method for fast DFE computation in the unconstrained formulation, the new algorithm requires three additional vector recursions for propagating the Kalman vector (here obtained in $N+M-1$ iterations, due to the general weighting matrix), considering the more general case of colored noise. The recursions of this paper can also be generalized to the MIMO case, similarly to the unconstrained extension reported in [4]. In a forthcoming work, we shall consider efficient recursions applied to the magnitude constrained DFE formulation of [2].

[4]Note that here the forgetting factor corresponding to a certain RLS problem is always equal to one. However, our fast recursions deal with the problem of processing a finite set of data samples. In other words, for our purpose, the algorithm must stop when $i = N$. In this case there is no need of much concern with error propagation effects, considering that the accumulated errors during such finite interval can be accounted for by proper increase of wordlength, or perhaps another mechanism to enforce stability.

Initialization

$\zeta_M^f(-2) = \zeta_M^b(-1) = c_0$
$w_{-2}^f = R_v^{-1} q$
$w_{M,-1}^b = R_v^{-1} p$
$k_{M,-1} = 0$
$\gamma_M(0) = 1$

For $i = 0$ to $\delta + 1$, run the usual fast Kalman recursions;

When $i = \delta + 1$, set:

$t_{M,\delta} = \gamma_M(\delta) k_{M,\delta}$
$\alpha_M^h(\delta) = f_M(\delta)$
$\gamma_M^{-1}(\delta+1) := \gamma_M^{-1}(\delta+1) - 1 + 1/\mu$
$\gamma_{M+1}^{-h}(\delta+1) = 1 - \gamma_{M+1}(\delta) + (\mu-1)^{-1}$

For $i \geq \delta + 2$:

$\alpha_M(i-1) = h(i) - h_{M,i-1} w_{M,i-2}^f$
$f_M(i-1) = \gamma_M(i-1)\alpha_M(i-1)$
$r_M(i-1) = h_{M,\delta} k_{M,i-1}$
$r_{M+1}(i-1) = r_M(i-1) + \frac{\alpha_M^*(i-1)\alpha_M^h(i-2)}{\xi_M^f(i-2)}$
$s_{M+1}(i-1) = r_{M+1}(i-1)\gamma_{M+1}^h(i-1)$
$k'_{M,i-1} = k_{M,i-1} - s_{M+1}(i-1) t_{M,i-2}$
$\tau_M(i-1) = \alpha_M^*(i-1) - s_{M+1}(i-1)\alpha^{h*}(i-2)$
$\breve{k}_{M+1,i} = \begin{bmatrix} 0 \\ k'_{M,i-1} \end{bmatrix} + \frac{\tau_M(i-1)}{\zeta_M^f(i-2)} \begin{bmatrix} 1 \\ -w_{M,i-2}^f \end{bmatrix}$
$\zeta_M^f(i-1) = \zeta_M^f(i-2) + \alpha_M^*(i-1) f_M(i-1)$
$w_{M,i-1}^f = w_{M,i-2}^f + k_{M,i-1} f_M(i-1)$
$\gamma_{M+1}(i-1) = \gamma_M(i-1)\frac{\zeta_M^f(i-2)}{\zeta_M^f(i-1)}$
$\gamma_{M+1}^{-h}(i) = \gamma_{M+1}^{-h}(i-1) - \gamma_{M+1}(i-1)|r_{M+1}(i-1)|^2$
$\breve{\gamma}_{M+1}^{-1}(i) = \gamma_{M+1}^{-1}(i-1) - s_{M+1}(i-1) r_{M+1}^*(i-1)$
$\alpha_M^h(i-1) = \alpha_M^h(i-2) - r_M(i-1) f_M(i-1)$
$t_{M,i-1} = t_{M,i-2} - r_M^*(i-1)\gamma_M(i-1) k_{M,i-1}$

$\nu_M(i) =$ (last entry of $\breve{k}_{M+1,i}$)
$k_{M,i} = \breve{k}_{1:M,i} + \nu_M(i) w_{M,i-1}^b$
$\beta_M(i) = \zeta_M^b(i-1)\nu_M^*(i)$
$\gamma_M(i) = [\breve{\gamma}_{M+1}^{-1}(i) - \beta_M(i)\nu_M(i)]^{-1}$
$b_M(i) = \gamma_M(i)\beta_M(i)$
$\zeta_M^b(i) = \zeta_M^b(i-1) + \beta_M^*(i) b_M(i)$
$w_{M,i}^b = w_{M,i-1}^b + k_{M,i} b_M(i)$

Table 1: Fast Computation of the feedfoward filter.

References

[1] R. L. Kosut, C. Wonzoo, C. R. Johnson, and S. P. Boyd, "On achieving reduced error propagation sensitivity in DFE design via convex optimization," *Proc. of the 39th IEEE Conference on Decision and Control*, vol. 5, pp. 4320–4323, Dec. 2001.

[2] F. Zhao, G. Mathew, and B. Farhang-Boroujeny, "Techniques for Minimizing Error Propagation in Decision Feedback Detectors for Recording Channels," *IEEE Trans. on Magnetics*, vol. 37, no. 1, pp. 592–602, Jan. 2001.

[3] R. Merched and N. R. Yousef, "Fast Techniques for Computing Finite-Length MMSE Decision Feedback Equalizers," *Proc. of ICASSP*, vol. 4, pp. 1005–1008, May 2004.

[4] R. Merched and N. R. Yousef, "Fast Techniques for Computing Finite-Length MIMO MMSE Decision Feedback Equalizers," *IEEE Trans. on Signal Processing*, Feb. 2006.

[5] R. Merched and A. H. Sayed, "Extended Fast Fixed-order RLS adaptive filters," *IEEE Trans. on Signal Processing*, vol. 49, no. 12, pp. 3015–3031, Dec. 2001.

[6] A. H. Sayed, *Fundamentals of Adaptive Filtering*, John Wiley & Sons, New Jersey, 2003.