EXACT RLS LAGUERRE-LATTICE ADAPTIVE FILTERING

RICARDO MERCHED AND ALI H. SAYED

Electrical Engineering Department
University of California
Los Angeles, CA 90095

ABSTRACT

This paper solves the problem of designing *exact* RLS lattice (or order-recursive) algorithms for adaptive filters that do not involve tapped-delay-line structures. As a special case, an exact RLS Laguerre lattice filter is obtained.

1. INTRODUCTION

As is well-known, all the derivations that are available so far in the literature for exact RLS order-recursive filters are intrinsically based on the assumption of regression vectors with shift structure (see, e.g., [1]-[3] and the references therein). The resulting filters are therefore not applicable to situations that involve other filter structures, such as Laguerre-based networks, where successive regression vectors are not shifted versions of each other.

We resolve this issue in this paper and develop a framework for RLS adaptive lattice filtering that applies to more general data structures. One consequence of our derivation will be the first *exact* RLS Laguerre-based lattice filter. This is in contrast, for example, to the gradient adaptive laguerre-lattice (GALL) filter developed in [4]. Both the GALL solution and the RLS solution are $\mathcal{O}(M)$ algorithms. One advantage of an RLS-based algorithm is that least-squares methods offer significant improvement in convergence performance. This is in addition to other advantages offered by Laguerre networks such as superior modeling capabilities to FIR networks, at a reduced number of taps and with a guaranteed stable performance (see [4, 5, 6]).

While some of our expressions may look familiar to readers acquainted with the theory of least-squares lattice filters, our presentation actually has three contributions that are essential to the extension to more general data structures:

1. First, all expressions are derived *without* assuming any underlying structure in the regression vectors. The general consensus in the literature so far has been that some (or most) of the relations derived in Sec. 3 are valid only for shift structured data.
2. Second, the derivation shows that it is possible to derive *efficient* order-recursive RLS filters even for cases where the regression vectors do not possess shift structure. This is achieved by pointing out the exact variable whose update is intimately affected by the data structure. The derivation also shows what kind of data structures lead to fast order-recursive filters.
3. Third, all order-recursive relations are derived by explicitly solving *regularized* least-squares problems. In contrast, similar relations have always been derived in the literature without taking into account the need for regularization; this need is usually accounted for by initializing lattice algorithms with certain small initial conditions without proper justification. Our arguments resolve this inconsistency.

This material was based on work supported in part by the National Science Foundation under awards CCR-9732376 and ECS-9820765. The work of R. Merched was also in part supported by a fellowship from CAPES, Brazil.

2. REGULARIZED LEAST-SQUARES

We first provide a brief review of the regularized least-squares problem. Thus given a column vector $y \in \mathbb{C}^{N+1}$ and a data matrix $H \in \mathbb{C}^{(N+1)\times M}$, the exponentially-weighted least-squares problem seeks the column vector $w \in \mathbb{C}^M$ that solves

$$\min_{w} \left[\mu\lambda^{N+1}\|w\|^2 \ + \ (y - Hw)^* W (y - Hw)\right], \qquad (1)$$

where μ is a scalar positive regularization parameter (usually small), and $W = (\lambda^N \oplus \lambda^{N-1} \oplus \cdots \oplus 1)$ is a weighting matrix that is defined in terms of a forgetting factor λ satisfying $0 \ll \lambda < 1$. The symbol $*$ denotes complex conjugate transposition.

The individual entries of y will be denoted by $\{d(i)\}$, and the individual rows of H will be denoted by $\{u_i\}$. Let w_N denote the optimal solution of (1). It is given by

$$w_N = (\mu\lambda^{N+1} I + H^* W H)^{-1} H^* W y \triangleq P_N H^* W y,$$

We further let $\widehat{y}$ denote the vector $\widehat{y} \triangleq H w_N$. We shall refer to $\widehat{y}$ as the *regularized* projection (or simply projection) of the observation vector y onto the range space of H, $\mathcal{R}(H)$. We also define the a posteriori and a priori error vectors, $e_N = y - Hw_N$ and $\epsilon_N = y - Hw_{N-1}$. Let $\xi(N)$ denote the minimum cost of (1). Then $\xi(N) = y^* W e_N$.

The last entries of e_N and ϵ_N are the a posteriori and a priori estimation errors at time N and they are given by $e(N) = d(N) - u_N w_N$ and $\epsilon(N) = d(N) - u_N w_{N-1}$. They are both related by a conversion factor, $e(N) = \gamma(N)\epsilon(N)$, where $\gamma(N) = 1 - u_N P_N u_N^*$.

3. ORDER-RECURSIVE RELATIONS

We now derive several order-recursive relations. Before proceeding, we should remark that since in the remainder of this paper we deal primarily with order-recursive least-squares problems, it becomes important to explicitly indicate the size of all quantities involved (in addition to a time index). For example, we shall write $w_{M,N}$ instead of w_N to indicate that it is a vector of order M that is computed by using data up to time N. We shall also write $H_{M,N}$ instead of H, y_N instead of y, W_N instead of W, and $P_{M,N}$ instead of P_N. In a similar vein, we shall write $\widehat{y}_{M,N}$, $e_{M,N}$, $\epsilon_{M,N}$, $e_M(N)$, $\epsilon_M(N)$, $\gamma_M(N)$, $\xi_M(N)$.

3.1. Order-Updating

Assume for simplicity of presentation that $M = 3$. Consider the (regularized) projection of y_N onto $\mathcal{R}(H_{3,N})$, viz., $\widehat{y}_{3,N} = H_{3,N}P_{3,N}H_{3,N}^*W_N y_N$. Now suppose that one more column is appended to $H_{3,N}$,

$$H_{3,N} = \left[\begin{array}{cc} H_{3,N} & x_{3,N} \end{array}\right], \tag{2}$$

where $x_{3,N} = \text{col}\{u(0,3), \ldots, u(N,3)\}$. The projection of y_N onto $\mathcal{R}(H_{4,N})$ is now $\widehat{y}_{4,N} = H_{4,N}P_{4,N}H_{4,N}^*W_N y_N$. We can relate both projections of y_N by noting that

$$P_{4,N} = \tag{3}$$

$$\left[\begin{array}{cc} P_{3,N} & 0 \\ 0 & 0 \end{array}\right] + \frac{1}{\mu\lambda^{N+1} + \xi_3^b(N)} \left[\begin{array}{c} -w_{3,N}^b \\ 1 \end{array}\right] \left[\begin{array}{cc} -w_{3,N}^{b*} & 1 \end{array}\right]$$

where $w_{3,N}^b$ is the solution to the least-squares problem:

$$\min_{w_3^b} \left[\mu\lambda^{N+1}\|w_3^b\|^2 + (x_{3,N} - H_{3,N}w_3^b)^* W_N (x_{3,N} - H_{3,N}w_3^b)\right]$$

and $\xi_3^b(N)$ is the corresponding minimum cost. This problem projects $x_{3,N}$ onto $\mathcal{R}(H_{3,N})$. Let $b_{3,N} = x_{3,N} - H_{3,N}w_{3,N}^b$ denote the resulting (backward) estimation error vector. Substituting (3) into the expression for $\widehat{y}_{4,N}$ and subtracting y_N from both sides of the resulting equation we get

$$\boxed{e_{4,N} = e_{3,N} - \kappa_3(N) b_{3,N}} \tag{4}$$

where we defined the scalar coefficient

$$\kappa_3(N) \triangleq \frac{b_{3,N}^* W_N y_N}{\mu\lambda^{N+1} + \xi_3^b(N)} \triangleq \frac{\rho_3^*(N)}{\mu\lambda^{N+1} + \xi_3^b(N)}\,. \tag{5}$$

The recursion (4) for $e_{4,N}$ depends on $b_{3,N}$. We are thus motivated to study the propagation of $b_{3,N}$ more closely.

3.2. Backward Estimation Problem

We start by partitioning $H_{3,N}$ as $H_{3,N} = \left[\begin{array}{cc} x_{0,N} & \bar{H}_{2,N} \end{array}\right]$. Using arguments similar to those that led to (4), it is straightforward to verify that

$$\boxed{b_{3,N} = \bar{b}_{2,N} - \kappa_2^b(N) f_{2,N}}$$

where the scalar coefficient $\kappa_2^b(N)$ is defined as

$$\kappa_2^b(N) \triangleq \frac{f_{2,N}^* W_N x_{3,N}}{\mu\lambda^{N+1} + \xi_2^f(N)} \triangleq \frac{\delta_2(N)}{\mu\lambda^{N+1} + \xi_2^f(N)}, \tag{6}$$

and $f_{2,N}$ is the residual error that results from solving

$$\min_{w_2^f} \left[\mu\lambda^{N+1}\|w_2^f\|^2 + (x_{0,N} - \bar{H}_{2,N}w_2^f)^* W_N (x_{0,N} - \bar{H}_{2,N}w_2^f)\right]$$

whose minimum cost we denote by $\xi_2^f(N)$. This problem projects $x_{0,N}$ onto $\mathcal{R}(\bar{H}_{2,N})$. Likewise, $\bar{b}_{2,N}$ is the residual error that results from solving

$$\min_{w_2^{\bar{b}}} \left[\mu\lambda^{N+1}\|w_2^{\bar{b}}\|^2 + (x_{3,N} - \bar{H}_{2,N}w_2^{\bar{b}})^* W_N (x_{3,N} - \bar{H}_{2,N}w_2^{\bar{b}})\right]$$

whose minimum cost we denote by $\xi_2^{\bar{b}}(N)$. This problem projects $x_{3,N}$ onto $\mathcal{R}(\bar{H}_{2,N})$.

3.3. Forward Estimation Problem

By similar arguments, $f_{2,N}$ can be updated as follows:

$$\boxed{f_{3,N} = f_{2,N} - \kappa_2^f(N)\bar{b}_{2,N}}$$

where $\kappa_2^f(N)$ is defined as

$$\kappa_2^f(N) \triangleq \frac{\bar{b}_{2,N}^* W_N x_{0,N}}{\mu\lambda^{N+1} + \xi_2^{\bar{b}}(N)} = \frac{\delta_2^*(N)}{\mu\lambda^{N+1} + \xi_2^{\bar{b}}(N)}. \tag{7}$$

Note that we used $\delta_2^*(N)$ in the numerator of $\kappa_2^f(N)$ and $\delta_2(N)$ in the numerator of $\kappa_2^b(N)$ in (6), since it can be easily verified that $[f_{2,N}^* W_N x_{3,N}]^* = \bar{b}_{2,N}^* W_N x_{0,N}$.

Summarizing, we have so far derived the following order-update relations for the last entries of the error vectors $\{e_{M,N}, b_{M,N}, f_{M,N}\}$ (written here for a generic order M):

$$\left\{\begin{array}{lcl} e_{M+1}(N) & = & e_M(N) - \kappa_M(N) b_M(N) \\ b_{M+1}(N) & = & \bar{b}_M(N) - \kappa_M^b(N) f_M(N) \\ f_{M+1}(N) & = & f_M(N) - \kappa_M^f(N)\bar{b}_M(N) \end{array}\right.$$

We still need to derive a relation for $\bar{b}_{M,N}$. We postpone this discussion to Sec. 4.1 due to its intrinsic dependence on data structure.

We now show how to update the quantities $\delta_M(N)$, $\rho_M(N)$, $\xi_M^b(N)$, $\xi_M^{\bar{b}}(N)$, and $\xi_M^f(N)$, which are needed in the evaluation of the (reflection) coefficients $\kappa_M(N)$, $\kappa_M^b(N)$, and $\kappa_M^f(N)$. To do so, we first derive below a general update result.

3.4. A General Time-Update Result

Consider a generic data matrix of the form $\left[\begin{array}{ccc} x & \bar{H} & z \end{array}\right]$ where x and z are column vectors, and $\bar{H}$ is a matrix of appropriate dimensions. Define the weighted inner product $\Delta = x^* W \tilde{z}$, where $\tilde{z}$ is the residual vector from a regularized projection of z onto $\mathcal{R}(\bar{H})$, namely $\tilde{z} = z - \bar{H}w_z$, where w_z is obtained by solving

$$\min_{w} \left[\mu\lambda^{N+1}\|w\|^2 + (z - \bar{H}w)^* W (z - \bar{H}w)\right] \tag{8}$$

where, as before, $W = \text{diag}\{\lambda^N, \ldots, \lambda, 1\}$.

Now assume that one more row is appended to the data matrix, say

$$\left[\begin{array}{ccc} x & \bar{H} & z \\ \alpha & h & \beta \end{array}\right] \triangleq \left[\begin{array}{ccc} x_1 & \bar{H}_1 & z_1 \end{array}\right],$$

and introduce the corresponding factor $\Delta_1 = x_1^* W_1 \tilde{z}_1$, where $W_1 = (\lambda W \oplus 1)$. We would like to relate Δ_1 and Δ.

Let w_{z_1} denote the solution of a problem similar to (8) with $\{z, \bar{H}, W, \lambda^{N+1}\}$ replaced by $\{z_1, \bar{H}_1, W_1, \lambda^{N+2}\}$. Likewise, let

w_{x_1} denote the solution of (8) with $\{z, \bar{H}, W, \lambda^{N+1}\}$ replaced by $\{x_1, \bar{H}_1, W_1, \lambda^{N+2}\}$. Now define the a posteriori errors $\tilde{\alpha} = \alpha - hw_{x_1}$ and $\tilde{\beta} = \beta - hw_{z_1}$, as well as the conversion factor

$$\gamma = 1 - h\left[\mu\lambda^{N+2}I + \bar{H}_1^* W_1 \bar{H}_1\right]^{-1} h^* \triangleq 1 - h\bar{P}_1 h^*.$$

From the definition of Δ_1 we can show that

$$\boxed{\Delta_1 = \lambda\Delta + \frac{\tilde{\alpha}^*\tilde{\beta}}{\gamma}} \qquad (9)$$

3.5. Time-Update Relations

We can now use the general result (9) to derive the following time-updates:

$$\begin{aligned}
\delta_M(N) &= \lambda\delta_M(N-1) + f_M^*(N)\bar{b}_M(N)/\bar{\gamma}_M(N) && (10)\\
\rho_M(N) &= \lambda\rho_M(N-1) + e_M^*(N)\bar{b}_M(N)/\gamma_M(N) && (11)\\
\xi_M^{\bar{b}}(N) &= \lambda\xi_M^{\bar{b}}(N-1) + |\bar{b}_M(N)|^2/\bar{\gamma}_M(N) && (12)\\
\xi_M^{f}(N) &= \lambda\xi_M^{f}(N-1) + |f_M(N)|^2/\bar{\gamma}_M(N) && (13)\\
\xi_M^{b}(N) &= \lambda\xi_M^{b}(N-1) + |b_M(N)|^2/\gamma_M(N) && (14)
\end{aligned}$$

where the conversion factors $\{\bar{\gamma}_M(N), \gamma_M(N)\}$ satisfy the order-updates:

$$\begin{aligned}
\gamma_{M+1}(N) &= \gamma_M(N) - \frac{|b_M(N)|^2}{\mu\lambda^{N+1} + \xi_M^b(N)} && (15)\\
\bar{\gamma}_{M+1}(N) &= \bar{\gamma}_M(N) - \frac{|\bar{b}_M(N)|^2}{\mu\lambda^{N+1} + \xi_M^{\bar{b}}(N)} && (16)
\end{aligned}$$

So far we have derived almost all the necessary recursions for the development of an adaptive lattice filter. All the results hold for *arbitrary* data structures. The only update missing is the one for the error sequence $\{\bar{b}_M(N)\}$. It is the update of these variables that is directly affected by data structure and it is the key to achieving a fast algorithm (by fast we mean $\mathcal{O}(M)$ operations per iteration for a filter of order M). For example, in the case of prewindowed input data with shift-structure it is easy to conclude that $\bar{b}_M(N) = b_M(N-1)$, $\xi_M^{\bar{b}}(N) = \xi_M^b(N-1)$, and $\bar{\gamma}_M(N) = \gamma_M(N-1)$. These equalities eliminate the need for recursions (12) and (16) and the general lattice recursions collapse to the well-known tapped-delay-line lattice network.

Now, what if two successive columns of the input data matrix $H_{M,N}$ are not shifted versions of each other? Would it still be possible to derive a fast lattice algorithm? Interesting enough, the answer is positive for generalized data structures. We demonstrate this fact in the next section by considering an important example.

4. RLS LAGUERRE ADAPTIVE FILTERING

We assume $\lambda = 1$ in this section. Thus consider the Laguerre-based model of Figure 1 where

$$L_0(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}} \quad \text{and} \quad L(z) = \frac{z^{-1}-a}{1-az^{-1}}, \;\; 0 < |a| < 1, \qquad (17)$$

with prewindowed input data (i.e., $s(i) = 0$ for $i \leq 0$ and zero initial conditions).

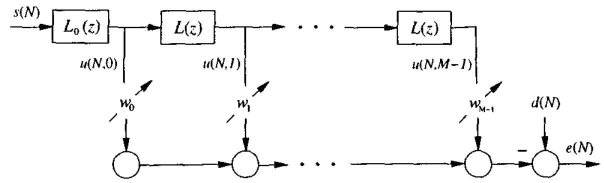

Figure 1: *A transversal Laguerre structure for adaptive filtering.*

Using the difference equations that define $\{L_0(z), L(z)\}$, it is possible to relate two successive columns of the data matrix $H_{M,N}$ as $x_{i+1,N} = \Phi_N x_{i,N}$, where Φ_N is an $(N+1) \times (N+1)$ lower triangular Toeplitz matrix whose first column is $\{-a, 1-a^2, a(1-a^2), \ldots, a^{N-1}(1-a^2)\}$. Of course, it also holds that $\bar{H}_{M,N} = \Phi_N H_{M,N}$.

4.1. Exploiting Data Structure

In order to relate the errors $\{b_M(N), \bar{b}_M(N)\}$, we start by noting that the product $\Phi^*\Phi$ is a rank-one modification of the identity matrix, namely it satisfies $\Phi_N^*\Phi_N = I - c_N c_N^*$, where $c_N = \sqrt{1-a^2}\begin{bmatrix} a^N & a^{N-1} & \ldots & a & 1 \end{bmatrix}^T$. Substituting this expression for $\Phi_N^*\Phi_N$ into the definition of $w_{M,N}^{\bar{b}}$, and after some derivation that we omit for lack of space, it can be verified that the following relation holds:

$$\boxed{\bar{b}_M(N) = \phi_N b_{M,N} + \frac{c_N^* b_{M,N}}{1 - c_N^*\hat{c}_{M,N}}\phi_N \hat{c}_{M,N}} \qquad (18)$$

where we defined the vector $\hat{c}_{M,N} \triangleq H_{M,N}P_{M,N}H_{M,N}^* c_N$, and where ϕ_N is the last row of Φ_N. The vector $\hat{c}_{M,N}$ has the interpretation of being the regularized projection of c_N onto $\mathcal{R}(H_{M,N})$. The above relation for $\bar{b}_M(N)$ involves four growing-length inner products on the right-hand side:

$$\{\phi_N b_{M,N}, \;\; c_N^* b_{M,N}, \;\; c_N^*\hat{c}_{M,N}, \;\; \phi_N\hat{c}_{M,N}\}.$$

Now, because the first two and the last two are related to each other, Eq. (18) can be further simplified to

$$\boxed{\bar{b}_M(N) = -\tfrac{1}{a}b_M(N) + \kappa_M^{\bar{b}}(N)\tilde{c}_M(N)}$$

where we defined

$$\kappa_M^{\bar{b}}(N) = \frac{1}{a}\frac{c_N^* b_{M,N}}{(1 - c_N^*\hat{c}_{M,N})} \triangleq \frac{\tau_M(N)}{a\psi_M(N)}.$$

Hence, all we really need to know is how to update the quantity $\tilde{c}_M(N)$ and the inner products $\tau_M(N)$ and $\psi_M(N)$. Due to space limitations, we simply mention that the following recursions can be established:

$$\begin{aligned}
\tilde{c}_{M+1}(N) &= \tilde{c}_M(N) - \kappa_M^c(N)b_M(N)\\
\psi_{M+1}(N) &= \psi_M(N) - \frac{|\tau_M(N)|^2}{\mu + \xi_M^b(N)}\\
\tau_M(N+1) &= a\tau_M(N) + \frac{\tilde{c}_M^*(N)b_M(N)}{\gamma_M(N)}
\end{aligned}$$

where we defined

$$\kappa_M^c(N) \triangleq \frac{\tau_M^*(N)}{\mu + \xi_M^b(N)}.$$

The recursion for $\tau_M(N)$ can be derived by extending the general result for the time-update of Δ in Sec. 3.4.

Figure 2 illustrates the structure of the RLS-Laguerre Lattice algorithm, which is listed in Table 1. We may note that we have redefined certain variables in order to save addition operations. For example, we defined $\bar{\xi}^b_M(N) = \mu + \xi^b_M(N)$. Then $\bar{\xi}^b_M(N)$ satisfies a similar recursion to that of $\xi^b_M(N)$ and it should be initialized with the value μ at time -1. Likewise, we introduced $\{\bar{\xi}^f_M(N), \bar{\xi}^{\bar{b}}_M(N)\}$.

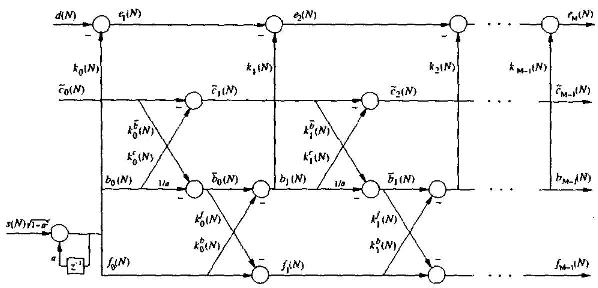

Figure 2: RLS-*Laguerre lattice network.*

5. CONCLUSIONS

Comparing Fig. 2 with the conventional lattice structure, we see that the new lattice filter is still fundamentally simple; the major modification is in the substitution of the delay blocks by a second lattice filter that runs in parallel. This in effect corresponds to replacing the delay blocks by simple time-variant lattice sections. We may remark that several simulations have been carried out to validate the algorithm. Lack of space forbids including these results here. The approach of this paper can be extended to other filter networks, other than the Laguerre structure, especially when differences of the form $W - \Phi^* W \Phi$ have low rank. In addition, normalized versions, array versions, and lattice schemes with feedback can also be developed. These extensions will be pursued elsewhere (see also [7]).

6. REFERENCES

[1] S. Haykin, *Adaptive Filter Theory*, 3rd edition, Prentice Hall, NJ, 1996.

[2] J. G. Proakis, C. M. Rader, F. Ling, and C. L. Nikias, *Advanced Digital Signal Processing*, Macmillan Publishing Co., NY, 1992.

[3] P. Strobach, *Linear Prediction Theory*, Springer-Verlag, Berlin Heidelberg, 1990.

[4] Z. Fejzo and H. Lev-Ari, "Adaptive Laguerre-lattice filters," *IEEE Trans. on Signal Processing*, vol. 45, no. 12, pp. 3006-3016, Dec. 1997.

[5] B. Wahlberg, "System identification using Laguerre models," *IEEE Trans. Automat. Control.*, vol. 36, pp. 551-562, May 1991.

[6] P. A. Regalia, *Adaptive IIR Filtering in Signal Processing and Control*, Marcel Dekker, NY, 1995.

[7] R. Merched and A. H. Sayed, "Fast RLS Laguerre adaptive filtering," *Proc. Allerton Conference*, IL, Sep. 1999.

Initialization:

For $M = 0$ to $M-1$ set:

μ is a small positive number.

$$\delta_M(-1) = \rho_M(-1) = \tau_M(-1) = 0$$
$$\bar{\xi}^f_M(-1) = \bar{\xi}^b_M(-1) = \bar{\xi}^{\bar{b}}_M(-1) = \mu$$

For $N \geq 0$, repeat:

$$u(N) = a u(N-1) + \sqrt{1-a^2}\, s(N)$$

$$\gamma_0(N) = 1 \qquad e_0(N) = d(N)$$
$$\bar{\gamma}_0(N) = 1 \qquad f_0(N) = u(N) \qquad \bar{c}_0(N) = \sqrt{1-a^2}$$
$$\psi_0(N) = 1 \qquad b_0(N) = u(N)$$

For $M = 0$ to $M-1$, repeat:

$$\tau_M(N) = a\tau_M(N-1) + \frac{\bar{c}^*_M(N) b_M(N)}{\gamma_M(N)}$$
$$\kappa^{\bar{b}}_M(N) = \frac{\tau_M(N)}{a\psi_M(N)}$$
$$\bar{b}_M(N) = -\frac{1}{a} b_M(N) + \kappa^{\bar{b}}_M(N)\bar{c}_M(N)$$

$$\bar{\xi}^f_M(N) = \bar{\xi}^f_M(N-1) + \frac{|f_M(N)|^2}{\bar{\gamma}_M(N)}$$
$$\bar{\xi}^b_M(N) = \bar{\xi}^b_M(N-1) + \frac{|b_M(N)|^2}{\gamma_M(N)}$$
$$\bar{\xi}^{\bar{b}}_M(N) = \bar{\xi}^{\bar{b}}_M(N-1) + \frac{|\bar{b}_M(N)|^2}{\bar{\gamma}_M(N)}$$

$$\delta_M(N) = \delta_M(N-1) + \frac{f^*_M(N)\bar{b}_M(N)}{\bar{\gamma}_M(N)}$$
$$\rho_M(N) = \rho_M(N-1) + \frac{e^*_M(N) b_M(N)}{\gamma_M(N)}$$

$$\gamma_{M+1}(N) = \gamma_M(N) - \frac{|b_M(N)|^2}{\bar{\xi}^b_M(N)}$$
$$\bar{\gamma}_{M+1}(N) = \bar{\gamma}_M(N) - \frac{|\bar{b}_M(N)|^2}{\bar{\xi}^{\bar{b}}_M(N)}$$
$$\psi_{M+1}(N) = \psi_M(N) - \frac{|\tau_M(N)|^2}{\bar{\xi}^b_M(N)}$$

$$\kappa^c_M(N) = \frac{\tau^*_M(N)}{\bar{\xi}^b_M(N)}$$
$$\kappa^b_M(N) = \frac{\delta_M(N)}{\bar{\xi}^f_M(N)}$$
$$\kappa^f_M(N) = \frac{\delta^*_M(N)}{\bar{\xi}^{\bar{b}}_M(N)}$$
$$\kappa_M(N) = \frac{\rho^*_M(N)}{\bar{\xi}^b_M(N)}$$

$$\bar{c}_{M+1}(N) = \bar{c}_M(N) - \kappa^c_M(N) b_M(N)$$
$$e_{M+1}(N) = e_M(N) - \kappa_M(N) b_M(N)$$
$$b_{M+1}(N) = \bar{b}_M(N) - \kappa^b_M(N) f_M(N)$$
$$f_{M+1}(N) = f_M(N) - \kappa^f_M(N)\bar{b}_M(N)$$

Table 1: The $\mathcal{O}(M)$ RLS-Laguerre lattice filter.