



# EE210A: Adaptation and Learning

## Professor Ali H. Sayed





# LECTURE #20

## LATTICE FILTERS

Sections in order: 41.1-41.3



# FORWARD AND BACKWARD ERRORS

**TABLE 40.1** A listing of the time and order-update relations derived in Secs. 40.2–40.5. All these updates are independent of data structure.

$$\begin{aligned}\xi_M^f(i) &= \lambda \xi_M^f(i-1) + |f_M(i)|^2 / \bar{\gamma}_M(i) \\ \xi_M^b(i) &= \lambda \xi_M^b(i-1) + |b_M(i)|^2 / \gamma_M(i) \\ \bar{\xi}_M^b(i) &= \lambda \bar{\xi}_M^b(i-1) + |\bar{b}_M(i)|^2 / \bar{\gamma}_M(i)\end{aligned}$$

$$\begin{aligned}\xi_{M+1}(i) &= \xi_M(i) - |\rho_M(i)|^2 / \zeta_M^b(i) \\ \xi_{M+1}^b(i) &= \xi_M^b(i) - |\delta_M(i)|^2 / \zeta_M^f(i) \\ \xi_{M+1}^f(i) &= \xi_M^f(i) - |\delta_M(i)|^2 / \zeta_M^b(i)\end{aligned}$$

$$\begin{aligned}\rho_M(i) &= \lambda \rho_M(i-1) + r_M^*(i) b_M(i) / \gamma_M(i) \\ \delta_M(i) &= \lambda \delta_M(i-1) + f_M^*(i) \bar{b}_M(i) / \bar{\gamma}_M(i)\end{aligned}$$

$$\begin{aligned}\kappa_M(i) &= \rho_M^*(i) / \zeta_M^b(i) \\ \kappa_M^b(i) &= \delta_M(i) / \zeta_M^f(i) \\ \kappa_M^f(i) &= \delta_M^*(i) / \zeta_M^b(i)\end{aligned}$$

$$\begin{aligned}r_{M+1}(i) &= r_M(i) - \kappa_M(i) b_M(i) \\ b_{M+1}(i) &= \bar{b}_M(i) - \kappa_M^b(i) f_M(i) \\ f_{M+1}(i) &= f_M(i) - \kappa_M^f(i) \bar{b}_M(i)\end{aligned}$$

$$\begin{aligned}\gamma_{M+1}(i) &= \gamma_M(i) - |b_M(i)|^2 / \zeta_M^b(i) \\ \gamma_{M+1}(i) &= \bar{\gamma}_M(i) - |f_M(i)|^2 / \zeta_M^f(i) \\ \bar{\gamma}_{M+1}(i) &= \bar{\gamma}_M(i) - |\bar{b}_M(i)|^2 / \zeta_M^b(i)\end{aligned}$$

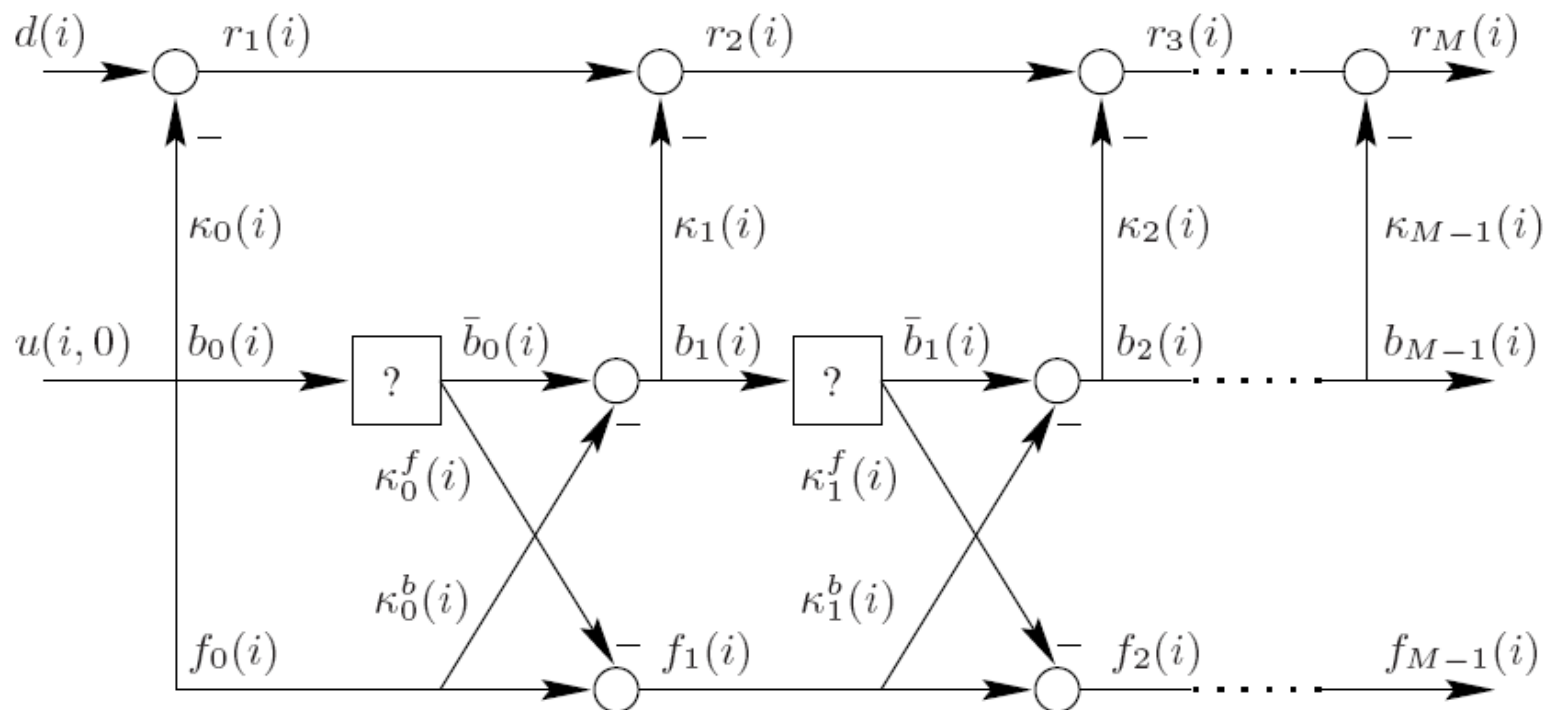


# MOTIVATION

**F**igure 41.1 illustrates how the error variables  $\{r_M(i), b_M(i), f_M(i)\}$  are related in terms of the reflection coefficients  $\{\kappa_M(i), \kappa_M^b(i), \kappa_M^f(i)\}$ , as was described in Chapter 40. It should be noted that the recursions listed in Table 40.1 help characterize *almost* fully the operation of the structure shown in the figure. The only missing piece of information is to know how to update the error sequence  $\{\bar{b}_M(i)\}$ . This fact is indicated schematically in Fig. 41.1 by the boxes with question marks. It is the update of these variables that is determined by data structure, and figuring out their update is the key to achieving an efficient algorithm; by efficient we mean  $O(M)$  operations per iteration for a filter of order  $M$ .



# MOTIVATION



**FIGURE 41.1** Relations among the residuals  $\{r_M(i), f_M(i), b_M(i)\}$ . The boxes with question marks indicate that we still need to develop a relation between  $\{b_M(i), \bar{b}_M(i)\}$ . This relation turns out to be a function of data structure. For example, if the regressors have shift structure, then the question marks will be replaced by pure delays,  $z^{-1}$ .



## 41.1 SIGNIFICANCE OF DATA STRUCTURE

To illustrate how the evaluation of  $\bar{b}_M(i)$  is dependent on data structure, we now focus on the case of regressors with shift structure, i.e., we assume that the entries of  $u_{M,i}$  are delayed versions of an input sequence  $\{u(\cdot)\}$  so that

$$\begin{aligned} u_{M,i} &= \begin{bmatrix} u(i) & u(i-1) & \dots & u(i-M+2) & u(i-M+1) \end{bmatrix} \\ u_{M,i+1} &= \begin{bmatrix} u(i+1) & u(i) & u(i-1) & \dots & u(i-M+2) \end{bmatrix} \end{aligned}$$

Comparing the expressions for both  $u_{M,i}$  and  $u_{M,i+1}$  we see that  $u_{M,i+1}$  is obtained from  $u_{M,i}$  by shifting the entries of the latter by one position to the right and introducing a new entry,  $u(i+1)$ , at the left. In this way, the data matrix  $H_{M,i}$  will also exhibit structure, e.g., for  $M = 4$ , it will have the form (compare with (40.18)):



$$H_{4,i} = \begin{bmatrix} u(0) & 0 & 0 & 0 \\ u(1) & u(0) & 0 & 0 \\ u(2) & u(1) & u(0) & 0 \\ u(3) & u(2) & u(1) & u(0) \\ \vdots & \vdots & \vdots & \vdots \\ u(i) & u(i-1) & u(i-2) & u(i-3) \end{bmatrix}$$

where we are assuming  $u(j) = 0$  for  $j < 0$ . Observe that every column of  $H_{M,i}$  is a shifted version of the previous column, i.e., every column is obtained from the previous column by shifting its entries downwards by one position and by adding a zero entry. This means that any two successive columns of  $H_{M,i}$ , say  $\{x_{j,i}, x_{j+1,i}\}$ , are related by the lower triangular shift matrix  $Z$ , i.e.,

$$x_{j+1,i} = Zx_{j,i} \quad (41.1)$$

where  $Z$  is the  $(i+1) \times (i+1)$  lower triangular matrix with zeros everywhere except for unit entries on the first sub-diagonal, e.g., for  $i = 3$ ,



$$Z = \begin{bmatrix} 0 & & & \\ 1 & 0 & & \\ & 1 & 0 & \\ & & 1 & 0 \end{bmatrix}$$

Now, as in (40.31) and (40.45), we partition  $H_{M+1,i}$  as

$$H_{M+1,i} = \begin{bmatrix} x_{0,i} & \bar{H}_{M,i} \end{bmatrix} = \begin{bmatrix} H_{M,i} & x_{M,i} \end{bmatrix}$$

to find that, in view of (41.1), the following relation holds between  $\{H_{M,i}, \bar{H}_{M,i}\}$ :

$$\boxed{\bar{H}_{M,i} = ZH_{M,i}} \quad (41.2)$$

In addition, the following relation also holds between  $\{H_{M,i}, H_{M,i-1}\}$ :

$$ZH_{M,i} = \begin{bmatrix} 0 \\ H_{M,i-1} \end{bmatrix} \quad (41.3)$$



With these relations we can now relate the residual vectors  $\bar{b}_{M,i}$  and  $b_{M,i}$ . Thus recall their definitions:

$$\bar{b}_{M,i} = x_{M+1,i} - \bar{H}_{M,i} w_{M,i}^{\bar{b}} \quad b_{M,i} = x_{M,i} - H_{M,i} w_{M,i}^b \quad (41.4)$$

where

$$\begin{cases} w_{M,i}^{\bar{b}} = \bar{P}_{M,i} \bar{H}_{M,i}^* \Lambda_i x_{M+1,i} & \bar{P}_{M,i} = (\lambda^i \Pi_M + \bar{H}_{M,i}^* \Lambda_i \bar{H}_{M,i})^{-1} \\ w_{M,i}^b = P_{M,i} H_{M,i}^* \Lambda_i x_{M,i} & P_{M,i} = (\lambda^{i+1} \Pi_M + H_{M,i}^* \Lambda_i H_{M,i})^{-1} \end{cases} \quad (41.5)$$

Substituting (41.2) and (41.3) into the expression for  $\bar{P}_{M,i}$  we obtain

$$\bar{P}_{M,i} = \left( \lambda^i \Pi_M + \begin{bmatrix} 0 & H_{M,i-1}^* \end{bmatrix} \Lambda_i \begin{bmatrix} 0 \\ H_{M,i-1} \end{bmatrix} \right)^{-1} = (\lambda^i \Pi_M + H_{M,i-1}^* \Lambda_{i-1} H_{M,i-1})^{-1}$$

That is,

$$\bar{P}_{M,i} = P_{M,i-1}$$



# DATA STRUCTURE

Substituting this result, along with (41.1), into expression (41.5) for  $w_{M,i}^{\bar{b}}$  we obtain

$$\begin{aligned}w_{M,i}^{\bar{b}} &= P_{M,i-1} H_{M,i}^* Z^* W_i Z x_{M,i} \\&= P_{M,i-1} H_{M,i-1}^* W_{i-1} x_{M,i-1} \\&= w_{M,i-1}^b\end{aligned}$$

so that

$$\begin{aligned}\bar{b}_{M,i} &= x_{M+1,i} - \bar{H}_{M,i} w_{M,i}^{\bar{b}} \\&= Z x_{M,i} - Z H_{M,i} w_{M,i-1}^b \\&= \begin{bmatrix} 0 \\ x_{M,i-1} \end{bmatrix} - \begin{bmatrix} 0 \\ H_{M,i-1} \end{bmatrix} w_{M,i-1}^b \\&= \begin{bmatrix} 0 \\ b_{M,i-1} \end{bmatrix}\end{aligned}$$

and, consequently, by equating the last entries of both sides,

$$\boxed{\bar{b}_M(i) = b_M(i-1)} \quad (41.6)$$



In other words, we find that in the shift-structured case, the residual errors  $\{b_M(i)\}$  are time delayed versions of  $\{b_M(i)\}$ . In a similar vein, it can be verified that

$$\xi_M^{\bar{b}}(i) = \xi_M^b(i-1), \quad \bar{\gamma}_M(i) = \gamma_M(i-1) \quad (41.7)$$

## ***Interpretation of the Estimation Errors***

In the shift-structured case, we can provide additional insights into the meaning of the *a posteriori* estimation errors  $\{f_M(i), b_M(i), r_M(i)\}$ . Indeed, in this scenario, we have from the definitions of these errors,

$$\begin{aligned} f_M(i) &= u(i) - u_{M,i-1} w_{M,i}^f \\ b_M(i) &= u(i-M) - u_{M,i} w_{M,i}^b \\ r_M(i) &= d(i) - u_{M,i} w_{M,i} \end{aligned}$$

where

$$u_{M,i} = \begin{bmatrix} u(i) & u(i-1) & \dots & u(i-M+1) \end{bmatrix}$$



# DATA STRUCTURE

In this way,  $f_M(i)$  can be interpreted as the forward prediction error in estimating  $u(i)$  from the  $M$  past values, while  $b_M(i)$  can be interpreted as the backward prediction error in estimating  $u(i - M)$  from the  $M$  future values.



# A-POSTERIORI LATTICE FILTER

**Algorithm 41.1 (A posteriori lattice filter)** Let  $0 \ll \lambda \leq 1$  be a forgetting factor and define  $\Pi_M = \eta^{-1} \text{diag}\{\lambda^{-2}, \lambda^{-3}, \dots, \lambda^{-(M+1)}\}$ . Consider a reference sequence  $\{d(j)\}$  and a regressor sequence  $\{u_{M,j}\}$  with shift structure and of dimension  $1 \times M$ , say  $u_{M,j} = [u(j) \ u(j-1) \ \dots \ u(j-M+1)]$ . For each  $i \geq 0$ , the  $M$ -th order *a posteriori* estimation error,  $r_M(i) = d(i) - u_{M,i} w_M$ , that results from the solution of the regularized least-squares problem:

$$\min_{w_M} \left[ \lambda^{i+1} w_M^* \Pi_M w_M + \sum_{j=0}^i \lambda^{i-j} |d(j) - u_{M,j} w_M|^2 \right]$$

can be computed as follows:

1. Initialization. From  $m = 0$  to  $m = M - 1$  set:

$$\begin{aligned} \delta_m(-1) &= \rho_m(-1) = 0, \quad \gamma_m(-1) = 1, \quad b_m(-1) = 0 \\ \zeta_m^f(-1) &= \eta^{-1} \lambda^{-2}, \quad \zeta_m^b(-1) = \eta^{-1} \lambda^{-m-2} \end{aligned}$$

2. For  $i \geq 0$ , repeat:

- Set  $\gamma_0(i) = 1$ ,  $b_0(i) = f_0(i) = u(i)$ , and  $r_0(i) = d(i)$
- From  $m = 0$  to  $m = M - 1$ , repeat:

$$\begin{aligned} \zeta_m^f(i) &= \lambda \zeta_m^f(i-1) + |f_m(i)|^2 / \gamma_m(i-1) \\ \zeta_m^b(i) &= \lambda \zeta_m^b(i-1) + |b_m(i)|^2 / \gamma_m(i) \\ \delta_m(i) &= \lambda \delta_m(i-1) + f_m^*(i) b_m(i-1) / \gamma_m(i-1) \\ \rho_m(i) &= \lambda \rho_m(i-1) + r_m^*(i) b_m(i) / \gamma_m(i) \\ \gamma_{m+1}(i) &= \gamma_m(i) - |b_m(i)|^2 / \zeta_m^b(i) \\ \kappa_m^b(i) &= \delta_m(i) / \zeta_m^f(i) \\ \kappa_m^f(i) &= \delta_m^*(i) / \zeta_m^b(i-1) \\ \kappa_m(i) &= \rho_m^*(i) / \zeta_m^b(i) \\ b_{m+1}(i) &= b_m(i-1) - \kappa_m^b(i) f_m(i) \\ f_{m+1}(i) &= f_m(i) - \kappa_m^f(i) b_m(i-1) \\ r_{m+1}(i) &= r_m(i) - \kappa_m(i) b_m(i) \end{aligned}$$



# A-POSTERIORI LATTICE FILTER

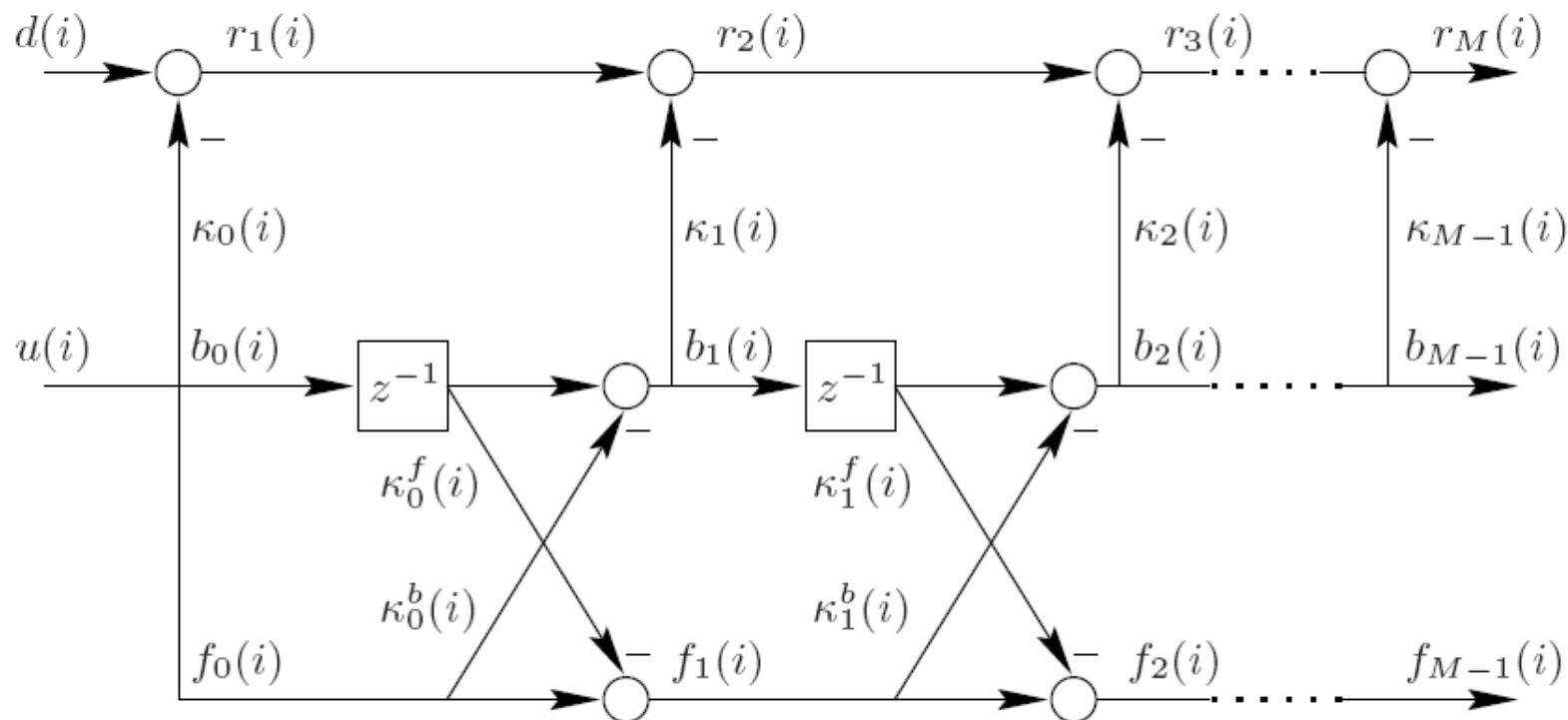


FIGURE 41.2 The *a posteriori*-based lattice filter.



# A-PRIORI LATTICE FILTER

## 41.3 A PRIORI-BASED LATTICE FILTER

Algorithm 41.1 relies on order-updates for the *a posteriori* estimation errors denoted by  $\{r_M(i), f_M(i), b_M(i)\}$ . A similar algorithm can be obtained by relying instead on *a priori* errors, which are defined as follows.

Refer again to Sec. 40.5 and to the data matrix  $H_{M+1,i}$  with its equivalent partitionings:

$$\begin{aligned} H_{M+1,i} = \begin{bmatrix} x_{0,i} & x_{1,i} & \dots & x_{M,i} \end{bmatrix} &= \begin{bmatrix} H_{M,i} & x_{M,i} \end{bmatrix} \\ &= \begin{bmatrix} x_{0,i} & \bar{H}_{M,i} \end{bmatrix} \\ &= \begin{bmatrix} x_{0,i} & \bar{H}_{M-1,i} & x_{M,i} \end{bmatrix} \end{aligned}$$

We define the *a priori* residual vectors as follows:

$$\begin{cases} e_{M,i} &= a \text{ priori residual from projecting } y_i \text{ onto } H_{M,i} \\ \beta_{M,i} &= a \text{ priori residual from projecting } x_{M,i} \text{ onto } H_{M,i} \\ \alpha_{M,i} &= a \text{ priori residual from projecting } x_{0,i} \text{ onto } \bar{H}_{M,i} \\ \bar{\beta}_{M,i} &= a \text{ priori residual from projecting } x_{M+1,i} \text{ onto } \bar{H}_{M,i} \end{cases}$$



# A-PRIORI LATTICE FILTER

where the projection problems for  $\{e_{M,i}, \beta_{M,i}\}$  employ the regularization matrix  $\lambda^{i+1}\Pi_M$ , while the projection problems for  $\{\alpha_{M,i}, \bar{\beta}_{M,i}\}$  employ the regularization matrix  $\lambda^i\Pi_M$ . The term *a priori* in the above definitions means that *prior* weight estimates are used in the definition of the errors. More specifically (compare with (40.59)),

$$\begin{cases} e_{M,i} &= y_i - H_{M,i}w_{M,i-1} \\ \beta_{M,i} &= x_{M+1,i} - H_{M+1,i}w_{M,i-1}^b \end{cases} \quad \begin{cases} \alpha_{M,i} &= x_{0,i} - \bar{H}_{M,i}w_{M,i-1}^f \\ \bar{\beta}_{M,i} &= x_{M+1,i} - \bar{H}_{M,i}w_{M,i-1}^b \end{cases}$$

where now  $w_{M,i-1}^f$ , for example, is the solution to a regularized least-squares problem of the form

$$\min_{w_M^f} \left[ \lambda^{i-1} w_M^{f*} \Pi_M w_M^f + (x_{0,i-1} - \bar{H}_{M,i-1} w_M^f)^* \Lambda_{i-1} (x_{0,i-1} - \bar{H}_{M,i-1} w_M^f) \right] \quad (41.8)$$

Comparing this cost function with (40.60) we see that  $i$  is replaced by  $i - 1$ . The last entries of the above residual vectors are denoted by  $\{e_M(i), \alpha_M(i), \beta_M(i), \bar{\beta}_M(i)\}$  and they are referred to as the *a priori* estimation errors. They are, of course, related to the corresponding *a posteriori* errors (40.61) via the associated conversion factors:



# A-PRIORI LATTICE FILTERS

$$\begin{aligned}r_M(i) &= e_M(i)\gamma_M(i), & b_M(i) &= \beta_M(i)\gamma_M(i) \\f_M(i) &= \alpha_M(i)\bar{\gamma}_M(i), & \bar{b}_M(i) &= \bar{\beta}_M(i)\bar{\gamma}_M(i)\end{aligned}$$

By following arguments similar to what we did in Secs. 40.2–40.4, and which led to (40.63), it can be verified that these *a priori* errors satisfy the following order-update relations in terms of the same reflection coefficients  $\{\kappa_M(i), \kappa_M^f(i), \kappa_M^b(i)\}$ :

$$\begin{aligned}e_{M+1}(i) &= e_M(i) - \kappa_M(i-1)\beta_M(i) \\ \beta_{M+1}(i) &= \bar{\beta}_M(i) - \kappa_M^b(i-1)\alpha_M(i) \\ \alpha_{M+1}(i) &= \alpha_M(i) - \kappa_M^f(i-1)\bar{\beta}_M(i)\end{aligned}\tag{41.9}$$



# A-PRIORI LATTICE FILTERS

Again, relations (41.9) hold irrespective of data structure. However, as was shown in Sec. 41.1 for the variables  $\{\bar{b}_M(i), b_M(i)\}$ , when the regressors possess shift structure it will also hold that

$$\bar{\beta}_M(i) = \beta_M(i-1)$$

In addition, the *a priori* estimation errors  $\{\alpha_M(i), \beta_M(i), e_M(i)\}$  will admit the following interpretations:

$$\alpha_M(i) = u(i) - u_{M,i-1} w_{M,i-1}^f \quad (41.10)$$

$$\beta_M(i) = u(i-M) - u_{M,i} w_{M,i-1}^b \quad (41.11)$$

$$e_M(i) = d(i) - u_{M,i} w_{M,i-1} \quad (41.12)$$

In other words,  $\alpha_M(i)$  denotes the forward prediction error in estimating  $u(i)$  from the  $M$  past values using the prior weight estimate,  $w_{M,i-1}^f$ , while  $\beta_M(i)$  denotes the backward prediction error in estimating  $u(i-M)$  from the  $M$  future values using the prior weight estimate,  $w_{M,i-1}^b$ . The resulting *a priori*-based lattice filter is listed in Alg. 41.2 and shown in Fig. 41.3.



# A-PRIORI LATTICE FILTERS

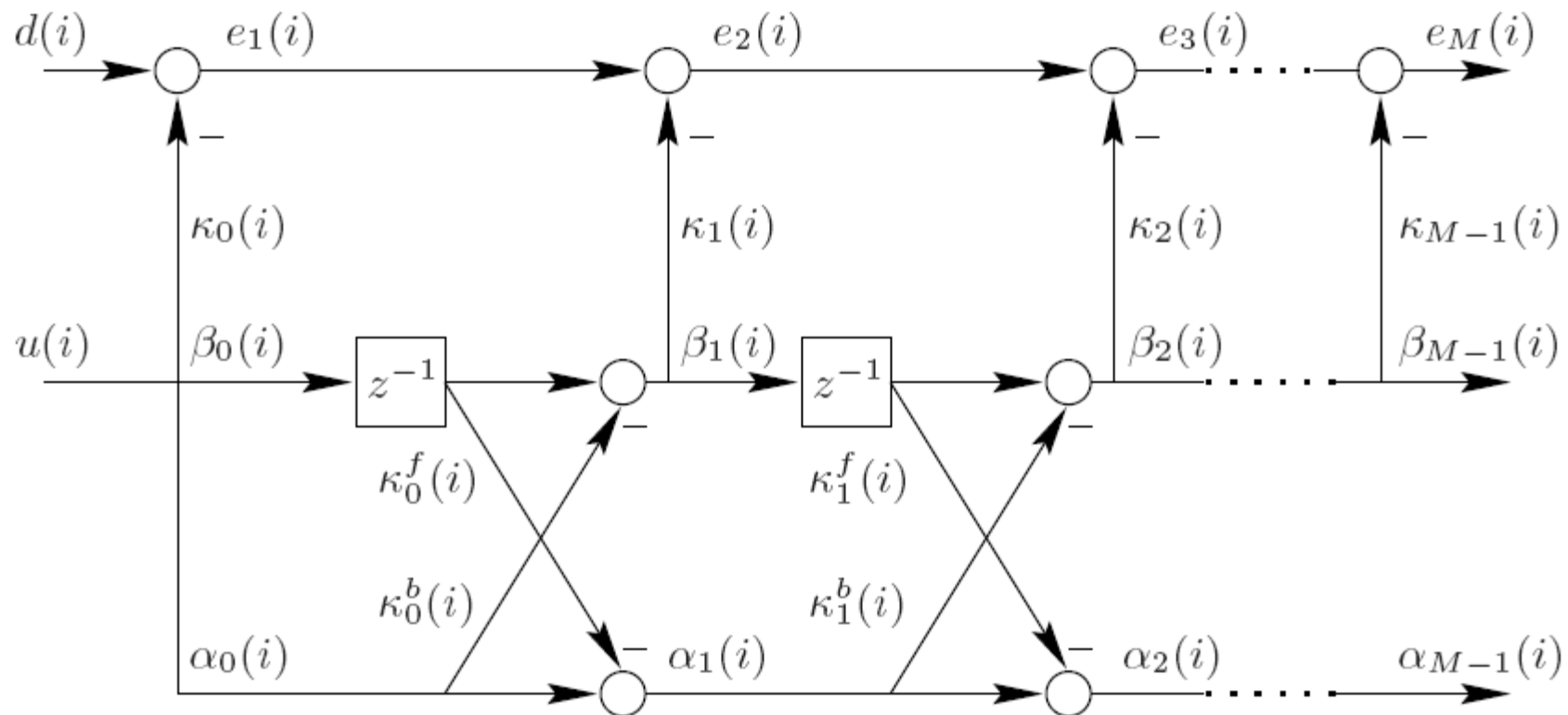


FIGURE 41.3 The *a priori*-based lattice filter.



# A-PRIORI LATTICE FILTERS

**Algorithm 41.2 (A priori lattice filter)** Consider the same setting of Alg. 41.1. For each  $i \geq 0$ , the  $M$ -th order *a priori* estimation error,  $e_M(i) = d(i) - u_{M,i} w_{M,i-1}$ , that results from the solution of the regularized least-squares problem

$$\min_{w_M} \left[ \lambda^i w_M^* \Pi_M w_M + \sum_{j=0}^{i-1} \lambda^{i-1-j} |d(j) - u_{M,j} w_M|^2 \right]$$

can be computed as follows:

1. Initialization. From  $m = 0$  to  $m = M - 1$  set:

$$\begin{aligned} \delta_m(-1) &= \rho_m(-1) = 0, \quad \gamma_m(-1) = 1, \quad \beta_m(-1) = 0 \\ \kappa_m^f(-1) &= \kappa_m^b(-1) = \kappa_m(-1) = 0 \\ \zeta_m^f(-1) &= \eta^{-1} \lambda^{-2}, \quad \zeta_m^b(-1) = \eta^{-1} \lambda^{-m-2} \end{aligned}$$

2. For  $i \geq 0$ , repeat:

- Set  $\gamma_0(i) = 1$ ,  $\beta_0(i) = \alpha_0(i) = u(i)$ , and  $e_0(i) = d(i)$
- From  $m = 0$  to  $m = M - 1$ , repeat:

$$\begin{aligned} \zeta_m^f(i) &= \lambda \zeta_m^f(i-1) + |\alpha_m(i)|^2 \gamma_m(i-1) \\ \zeta_m^b(i) &= \lambda \zeta_m^b(i-1) + |\beta_m(i)|^2 \gamma_m(i) \\ \delta_m(i) &= \lambda \delta_m(i-1) + \alpha_m^*(i) \beta_m(i-1) \gamma_m(i-1) \\ \rho_m(i) &= \lambda \rho_m(i-1) + e_m^*(i) \beta_m(i) \gamma_m(i) \\ \beta_{m+1}(i) &= \beta_m(i-1) - \kappa_m^b(i-1) \alpha_m(i) \\ \alpha_{m+1}(i) &= \alpha_m(i) - \kappa_m^f(i-1) \beta_m(i-1) \\ e_{m+1}(i) &= e_m(i) - \kappa_m(i-1) \beta_m(i) \\ \gamma_{m+1}(i) &= \gamma_m(i) - |\gamma_m(i) \beta_m(i)|^2 / \zeta_m^b(i) \\ \kappa_m^b(i) &= \delta_m(i) / \zeta_m^f(i) \\ \kappa_m^f(i) &= \delta_m^*(i) / \zeta_m^b(i-1) \\ \kappa_m(i) &= \rho_m^*(i) / \zeta_m^b(i) \end{aligned}$$



# ERROR-FEEDBACK FILTERS

**TABLE 42.1** A listing of time and order-update relations for the *a priori* estimation errors; these updates are independent of data structure.

$$\xi_M^f(i) = \lambda \xi_M^f(i-1) + |\alpha_M(i)|^2 \bar{\gamma}_M(i)$$

$$\xi_M^b(i) = \lambda \xi_M^b(i-1) + |\beta_M(i)|^2 \gamma_M(i)$$

$$\xi_M^{\bar{b}}(i) = \lambda \xi_M^{\bar{b}}(i-1) + |\bar{\beta}_M(i)|^2 \bar{\gamma}_M(i)$$

$$\xi_{M+1}(i) = \xi_{M+1}(i) - |\rho_M(i)|^2 / \zeta_M^b(i)$$

$$\xi_{M+1}^b(i) = \xi_M^{\bar{b}}(i) - |\delta_M(i)|^2 / \zeta_M^f(i)$$

$$\xi_{M+1}^f(i) = \xi_M^f(i) - |\delta_M(i)|^2 / \zeta_M^b(i)$$

$$e_{M+1}(i) = e_M(i) - \kappa_M(i-1) \beta_M(i)$$

$$\beta_{M+1}(i) = \bar{\beta}_M(i) - \kappa_M^b(i-1) \alpha_M(i)$$

$$\alpha_{M+1}(i) = \alpha_M(i) - \kappa_M^f(i-1) \bar{\beta}_M(i)$$

$$\kappa_M(i) = \kappa_M(i-1) + \beta_M^*(i) \gamma_M(i) e_{M+1}(i) / \zeta_M^b(i)$$

$$\kappa_M^f(i) = \kappa_M^f(i-1) + \bar{\beta}_M^*(i) \bar{\gamma}_M(i) \alpha_{M+1}(i) / \zeta_M^b(i)$$

$$\kappa_M^b(i) = \kappa_M^b(i-1) + \alpha_M^*(i) \bar{\gamma}_M(i) \beta_{M+1}(i) / \zeta_M^f(i)$$

$$\gamma_{M+1}(i) = \gamma_M(i) - |\gamma_M(i) \beta_M(i)|^2 / \zeta_M^b(i)$$

$$\gamma_{M+1}(i) = \bar{\gamma}_M(i) - |\bar{\gamma}_M(i) \alpha_M(i)|^2 / \zeta_M^f(i)$$

$$\bar{\gamma}_{M+1}(i) = \bar{\gamma}_M(i) - |\bar{\gamma}_M(i) \bar{\beta}_M(i)|^2 / \zeta_M^b(i)$$



# A-POSTERIORI ERROR FEEDBACK FILTER

**Algorithm 42.2 (A posteriori error-feedback filter)** Let  $0 \ll \lambda \leq 1$  be a forgetting factor and define  $\Pi_M = \eta^{-1} \text{diag}\{\lambda^{-2}, \lambda^{-3}, \dots, \lambda^{-(M+1)}\}$ . Consider a reference sequence  $\{d(j)\}$  and a regressor sequence  $\{u_{M,j}\}$  with shift structure and of dimension  $1 \times M$ , say  $u_{M,j} = [u(j) \dots u(j-M+1)]$ . For each  $i \geq 0$ , the  $M$ -th order *a posteriori* estimation error,  $r_M(i) = d(i) - u_{M,i} w_{M,i}$ , that results from the solution of the regularized least-squares problem:

$$\min_{w_M} \left[ \lambda^{i+1} w_M^* \Pi_M w_M + \sum_{j=0}^i \lambda^{i-j} |d(j) - u_{M,j} w_M|^2 \right]$$

can be computed as follows:

1. Initialization. From  $m = 0$  to  $m = M - 1$  set:

$$\gamma_m(-1) = 1, \quad b_m(-1) = 0, \quad \kappa_m^f(-1) = \kappa_m^b(-1) = \kappa_m(-1) = 0$$

$$\zeta_m^f(-1) = \eta^{-1} \lambda^{-2}, \quad \zeta_m^b(-1) = \eta^{-1} \lambda^{-m-2}$$

2. For  $i \geq 0$ , repeat:

- Set  $\gamma_0(i) = 1$ ,  $b_0(i) = f_0(i) = u(i)$ , and  $r_0(i) = d(i)$
- From  $m = 0$  to  $m = M - 1$ , repeat:

$$\begin{aligned} \zeta_m^f(i) &= \lambda \zeta_m^f(i-1) + |f_m(i)|^2 / \gamma_m(i-1) \\ \zeta_m^b(i) &= \lambda \zeta_m^b(i-1) + |b_m(i)|^2 / \gamma_m(i) \\ \gamma_{m+1}(i) &= \gamma_m(i) - |b_m(i)|^2 / \zeta_m^b(i) \\ \kappa_m(i) &= \frac{\gamma_{m+1}(i)}{\gamma_m(i)} \left[ \kappa_m(i-1) + \frac{b_m^*(i) r_m(i)}{\lambda \gamma_m(i) \zeta_m^b(i-1)} \right] \\ \kappa_m^f(i) &= \frac{\gamma_{m+1}(i-1)}{\gamma_m(i-1)} \left[ \kappa_m^f(i-1) + \frac{b_m^*(i-1) f_m(i)}{\lambda \gamma_m(i-1) \zeta_m^b(i-2)} \right] \\ \kappa_m^b(i) &= \frac{\gamma_{m+1}(i)}{\gamma_m(i-1)} \left[ \kappa_m^b(i-1) + \frac{f_m^*(i) b_m(i-1)}{\lambda \gamma_m(i-1) \zeta_m^f(i-1)} \right] \\ r_{m+1}(i) &= r_m(i) - \kappa_m(i) b_m(i) \\ b_{m+1}(i) &= b_m(i-1) - \kappa_m^b(i) f_m(i) \\ f_{m+1}(i) &= f_m(i) - \kappa_m^f(i) b_m(i-1) \end{aligned}$$



# A-PRIORI ERROR FEEDBACK FILTER

**Algorithm 42.1 (A priori error-feedback filter)** Let  $0 \ll \lambda \leq 1$  be a forgetting factor and define  $\Pi_M = \eta^{-1} \text{diag}\{\lambda^{-2}, \lambda^{-3}, \dots, \lambda^{-(M+1)}\}$ . Consider a reference sequence  $\{d(j)\}$  and a regressor sequence  $\{u_{M,j}\}$  with shift structure and of dimension  $1 \times M$ , say  $u_{M,j} = [u(j) \ \dots \ u(j-M+1)]$ . For each  $i \geq 0$ , the  $M$ -th order a priori estimation error,  $e_M(i) = d(i) - u_{M,i} w_{M,i-1}$ , that results from the solution of the regularized least-squares problem:

$$\min_{w_M} \left[ \lambda^i w_M^* \Pi_M w_M + \sum_{j=0}^{i-1} \lambda^{i-1-j} |d(j) - u_{M,j} w_M|^2 \right]$$

can be computed as follows:

1. Initialization. From  $m = 0$  to  $m = M - 1$  set:

$$\begin{aligned} \gamma_m(-1) &= 1, \quad \beta_m(-1) = 0 \\ \kappa_m^f(-1) &= \kappa_m^b(-1) = \kappa_m(-1) = 0 \\ \zeta_m^f(-1) &= \eta^{-1} \lambda^{-2}, \quad \zeta_m^b(-1) = \eta^{-1} \lambda^{-m-2} \end{aligned}$$

2. For  $i \geq 0$ , repeat:

- Set  $\gamma_0(i) = 1$ ,  $\beta_0(i) = \alpha_0(i) = u(i)$ , and  $e_0(i) = d(i)$
- From  $m = 0$  to  $m = M - 1$ , repeat:

$$\begin{aligned} \zeta_m^f(i) &= \lambda \zeta_m^f(i-1) + |\alpha_m(i)|^2 \gamma_m(i-1) \\ \zeta_m^b(i) &= \lambda \zeta_m^b(i-1) + |\beta_m(i)|^2 \gamma_m(i) \\ \beta_{m+1}(i) &= \beta_m(i-1) - \kappa_m^b(i-1) \alpha_m(i) \\ \alpha_{m+1}(i) &= \alpha_m(i) - \kappa_m^f(i-1) \beta_m(i-1) \\ e_{m+1}(i) &= e_m(i) - \kappa_m(i-1) \beta_m(i) \\ \kappa_m(i) &= \kappa_m(i-1) + \beta_m^*(i) \gamma_m(i) e_{m+1}(i) / \zeta_m^b(i) \\ \kappa_m^f(i) &= \kappa_m^f(i-1) + \beta_m^*(i-1) \gamma_m(i-1) \alpha_{m+1}(i) / \zeta_m^b(i-1) \\ \kappa_m^b(i) &= \kappa_m^b(i-1) + \alpha_m^*(i) \gamma_m(i-1) \beta_{m+1}(i) / \zeta_m^f(i) \\ \gamma_{m+1}(i) &= \gamma_m(i) - |\gamma_m(i) \beta_m(i)|^2 / \zeta_m^b(i) \end{aligned}$$



## 42.3 NORMALIZED LATTICE FILTER

---

The lattice filters studied so far require the evaluation of three reflection coefficients,

$$\{\kappa_M^f(i), \kappa_M^b(i), \kappa_M(i)\}$$

There is another equivalent lattice form that employs only two reflection coefficients. We denote these new coefficients by  $\{\kappa_M^a(i), \kappa_M^c(i)\}$ .



# ANGLE-NORMALIZED ERRORS

Define also the normalized estimation errors

$$b_M''(i) \triangleq b_M(i)/\gamma_M^{1/2}(i)\zeta_M^{b/2}(i)$$

$$\bar{b}_M''(i) \triangleq \bar{b}_M(i)/\bar{\gamma}_M^{1/2}(i)\zeta_M^{\bar{b}/2}(i)$$

$$f_M''(i) \triangleq f_M(i)/\bar{\gamma}_M^{1/2}(i)\zeta_M^{f/2}(i)$$

$$r_M''(i) \triangleq r_M(i)/\gamma_M^{1/2}(i)\zeta_M^{1/2}(i)$$



# NORMALIZED REFLECTION COEFFICIENTS

The normalized reflection coefficients that we are interested in are defined as follows:

$$\begin{aligned}\kappa_M^a(i) &\triangleq \frac{\delta_M^*(i)}{\zeta_M^{\bar{b}/2}(i)\zeta_M^{f/2}(i)} = \kappa_M^f(i) \cdot \frac{\zeta_M^{\bar{b}/2}(i)}{\zeta_M^{f/2}(i)} = \kappa_M^{b*}(i) \cdot \frac{\zeta_M^{f/2}(i)}{\zeta_M^{\bar{b}/2}(i)} \\ \kappa_M^c(i) &\triangleq \frac{\rho_M^*(i)}{\zeta_M^{b/2}(i)\zeta_M^{1/2}(i)} = \kappa_M(i) \cdot \frac{\zeta_M^{b/2}(i)}{\zeta_M^{1/2}(i)}\end{aligned}$$

That is, we scale the forward projection coefficient  $\kappa_M^f(i)$  by  $\zeta_M^{\bar{b}/2}(i)/\zeta_M^{f/2}(i)$ . Likewise for  $\kappa_M^c(i)$ .



# NORMALIZED LATTICE FILTER

**Algorithm 42.3 (Normalized lattice filter)** Consider the same setting of Alg. 42.2. For each  $i \geq 0$ , the  $M$ -th order *a posteriori* and *a priori* errors,  $\{e_M(i), r_M(i)\}$  can be computed as follows:

1. Initialization. Set

$$\zeta_0^b(-1) = \eta^{-1}\lambda^{-2}, \quad \zeta_0(-1) = \eta^{-1}\lambda^{-2}, \quad b_m''(-1) = 0 \quad \text{for } m = 0, \dots, M-1$$

2. For  $i \geq 0$ , repeat:

• Set

$$\begin{aligned} \zeta_0^b(i) &= \lambda \zeta_0^b(i-1) + |u(i)|^2, \quad \zeta_0(i) = \lambda \zeta_0(i-1) + |d(i)|^2, \quad \sigma_0(i) = \zeta_0^{1/2}(i) \\ b_0''(i) &= f_0''(i) = u(i)/\zeta_0^{b/2}(i), \quad r_0''(i) = d(i)/\zeta_0^{1/2}(i), \quad \gamma_0(i) = 1 \end{aligned}$$

• From  $m = 0$  to  $m = M-1$ , repeat:

$$\begin{aligned} p_m^b(i) &= \sqrt{1 - |b_m''(i)|^2}, \quad p_m^f(i) = \sqrt{1 - |f_m''(i)|^2} \\ p_m(i) &= \sqrt{1 - |r_m''(i)|^2} \\ \kappa_m^a(i) &= \kappa_m^a(i-1)p_m^b(i-1)p_m^f(i) + f_m''(i)b_m''^*(i-1) \\ \kappa_m^c(i) &= \kappa_m^c(i-1)p_m^b(i)p_m(i) + b_m''^*(i)r_m''(i) \\ p_m^a(i) &= \sqrt{1 - |\kappa_m^a(i)|^2}, \quad p_m^c(i) = \sqrt{1 - |\kappa_m^c(i)|^2} \\ r_{m+1}''(i) &= \frac{1}{p_m^b(i)p_m^c(i)} (r_m''(i) - \kappa_m^c(i)b_m''(i)) \\ b_{m+1}''(i) &= \frac{1}{p_m^f(i)p_m^a(i)} (b_m''(i-1) - \kappa_m^a(i)f_m''(i)) \\ f_{m+1}''(i) &= \frac{1}{p_m^b(i-1)p_m^a(i)} (f_m''(i) - \kappa_m^a(i)b_m''(i-1)) \\ \sigma_{m+1}(i) &= \sigma_m(i) p_m^c(i) p_m^b(i), \quad \gamma_{m+1}(i) = \gamma_m(i) (p_m^b(i))^2 \\ r_{m+1}(i) &= \sigma_{m+1}(i) r_{m+1}''(i) \\ e_{m+1}(i) &= r_{m+1}(i) / \gamma_{m+1}(i) \end{aligned}$$



# ARRAY LATTICE FILTERS

$$q_M(i) \triangleq \zeta_M^{b/2}(i) \kappa_M(i) \quad (43.2)$$

$$q_M^b(i) \triangleq \zeta_M^{f/2}(i) \kappa_M^b(i) \quad (43.7)$$

$$q_M^f(i) \triangleq \zeta_M^{\bar{b}/2}(i) \kappa_M^f(i) \quad (43.12)$$

$$\kappa_M(i) = q_M(i) / \zeta_M^{b/2}(i) \quad (43.6)$$

$$\kappa_M^b(i) = q_M^b(i) / \zeta_M^{f/2}(i) \quad (43.11)$$

$$\kappa_M^f(i) = q_M^b(i) / \zeta_M^{\bar{b}/2}(i) \quad (43.16)$$



# ARRAY LATTICE FILTERS

**Algorithm 43.1 (Array lattice filter)** Consider again the same setting of Alg. 42.2. For each  $i \geq 0$ , the  $M$ -th order *a posteriori* estimation error,  $r_M(i) = d(i) - u_{M,i} w_{M,i}$ , that results from the solution of the regularized least-squares problem

$$\min_{w_M} \left[ \lambda^{i+1} w_M^* \Pi_M w_M + \sum_{j=0}^i \lambda^{i-j} |d(j) - u_{M,j} w_M|^2 \right]$$

can be computed as follows:

1. Initialization. From  $m = 0$  to  $m = M - 1$  set:

$$\zeta_m^{f/2}(-1) = \sqrt{\eta^{-1} \lambda^{-2}}, \quad \zeta_m^{b/2}(-1) = \sqrt{\eta^{-1} \lambda^{-m-2}} \\ q_m(-1) = 0, \quad q_m^f(-1) = 0, \quad q_m^b(-1) = 0, \quad b_m(-1) = 0$$

2. For  $i \geq 0$ , repeat:

- Set  $\gamma_0^{1/2}(i) = 1$ ,  $b_0(i) = f_0'(i) = u(i)$ , and  $r_0'(i) = d(i)$
- For  $m = 0$  to  $m = M - 1$ , apply  $2 \times 2$  unitary rotations  $\Theta_{m,i}^f$ ,  $\Theta_{m,i}^b$ , and  $\Theta_{m,i}^{b*}$ , with positive (2,2) entries, in order to annihilate the (1,2) entries of the post-arrays below:

$$\begin{bmatrix} \lambda^{1/2} \zeta_m^{b/2}(i-2) & b_m^*(i-1) \\ \lambda^{1/2} q_m^f(i-1) & f_m^*(i) \end{bmatrix} \Theta_{m,i}^f = \begin{bmatrix} \zeta_m^{b/2}(i-1) & 0 \\ q_m^f(i) & f_{m+1}'(i) \end{bmatrix} \\ \begin{bmatrix} \lambda^{1/2} \zeta_m^{b/2}(i-1) & b_m^*(i) \\ \lambda^{1/2} q_m^f(i-1) & r_m^*(i) \\ 0 & \gamma_m^{1/2}(i) \end{bmatrix} \Theta_{m,i}^b = \begin{bmatrix} \zeta_m^{b/2}(i) & 0 \\ q_m^f(i) & r_{m+1}'(i) \\ b_m(i)/\zeta_m^{b/2}(i) & \gamma_{m+1}^{1/2}(i) \end{bmatrix} \\ \begin{bmatrix} \lambda^{1/2} \zeta_m^{f/2}(i-1) & f_m^*(i) \\ \lambda^{1/2} q_m^b(i-1) & b_m^*(i-1) \end{bmatrix} \Theta_{m,i}^{b*} = \begin{bmatrix} \zeta_m^{f/2}(i) & 0 \\ q_m^b(i) & b_{m+1}'(i) \end{bmatrix}$$

and set  $r_{m+1}(i) = r_{m+1}'(i) \gamma_{m+1}^{1/2}(i)$ .



# ARRAY LATTICE FILTERS

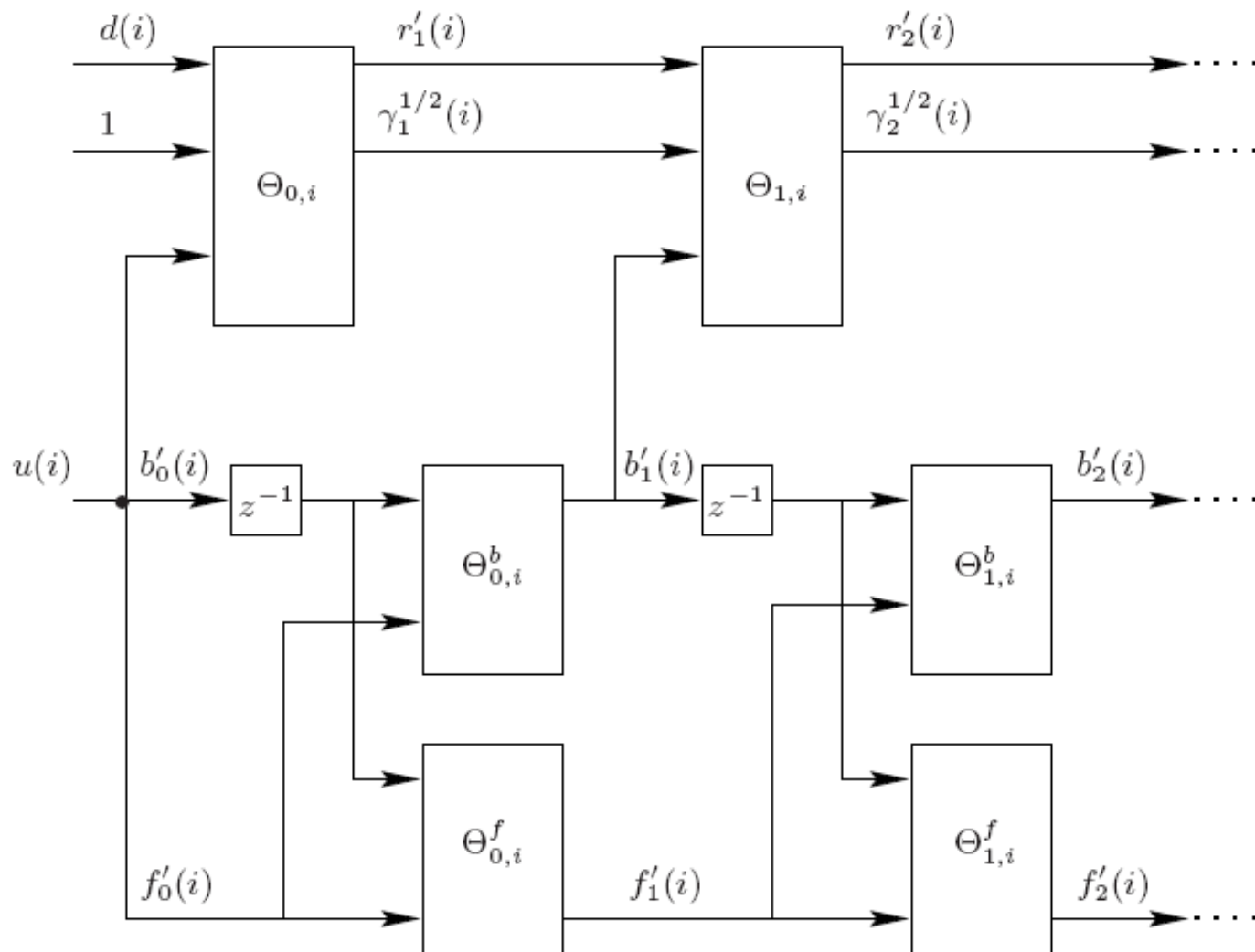


FIGURE 43.1 The QRD-based lattice filter.



**TABLE 43.1** Estimated computational cost per iteration for various lattice filters.

Algorithm	$\times$	$+$	$\div$	$\sqrt{\cdot}$	Reference
<i>A posteriori</i> lattice	$16M$	$8M$	$8M$	-	Algorithm 41.1
<i>A priori</i> lattice	$16M$	$8M$	$8M$	-	Algorithm 41.2
<i>A priori</i> error-feedback lattice	$18M$	$8M$	$2M$	-	Algorithm 42.1
<i>A posteriori</i> error-feedback lattice	$16M$	$9M$	$7M$	-	Algorithm 42.2
Normalized lattice	$18M$	$5M$	$3M$	$5M$	Algorithm 42.3
QRD-based lattice	$23M$	$8M$	$6M$	$2M$	Algorithm 43.1
Givens-based lattice	$27M$	$8M$	$4M$	$2M$	Problem X.7



# COMPUTER PROJECT

**Project X.1 (Performance of lattice filters in finite precision)** Although equivalent from a theoretical point of view, the performance of the varied lattice filters differ under finite-precision conditions. The purpose of this computer project is to illustrate these differences, as well as illustrate the recovery mechanism of some of the filters during the occurrence of impulsive interferences.

- (a) Generate 10 random coefficients of a channel and normalize its energy to unity. Feed unit-variance Gaussian input data through the channel and add Gaussian noise to its output. Set the noise power at 30 dB below the input signal power. Choose  $\lambda = 0.999$  and  $\eta = 10^6$  and train the following lattice filters using the input sequence of the channel as input to the lattice implementations and the noisy output of the channel as the reference sequence: 1. *A posteriori* lattice form; 2. *a priori* lattice form; 3. *a priori* lattice form with error feedback; 4. *a posteriori* lattice form with error feedback; 5. normalized lattice form and 6. array lattice form. Assume in your simulations a finite-precision implementation with  $B$  bits for signals including the sign bit; use the routine `quantize.m` from Computer Project IX.1. For each algorithm, generate an ensemble-average learning curve by averaging over 50 experiments of duration  $N = 200$  iterations each for the following choices: 1.  $B = 35$  bits; 2.  $B = 25$  bits; 3.  $B = 20$  bits; 4.  $B = 16$  bits and 5.  $B = 10$  bits. Which lattice forms appear to be most reliable in finite precision?



# COMPUTER PROJECT

- (b) For this part, assume first a floating-point implementation. Introduce an impulsive interference of unit amplitude to the input sequence at time instant  $i = 200$ . Generate ensemble-average learning curves for the lattice filters over  $N = 500$  iterations and observe whether they recover from the impulsive disturbance.
- (c) Repeat the simulations of part (b) in finite precision using  $B = 20$  bits and  $B = 10$  bits.



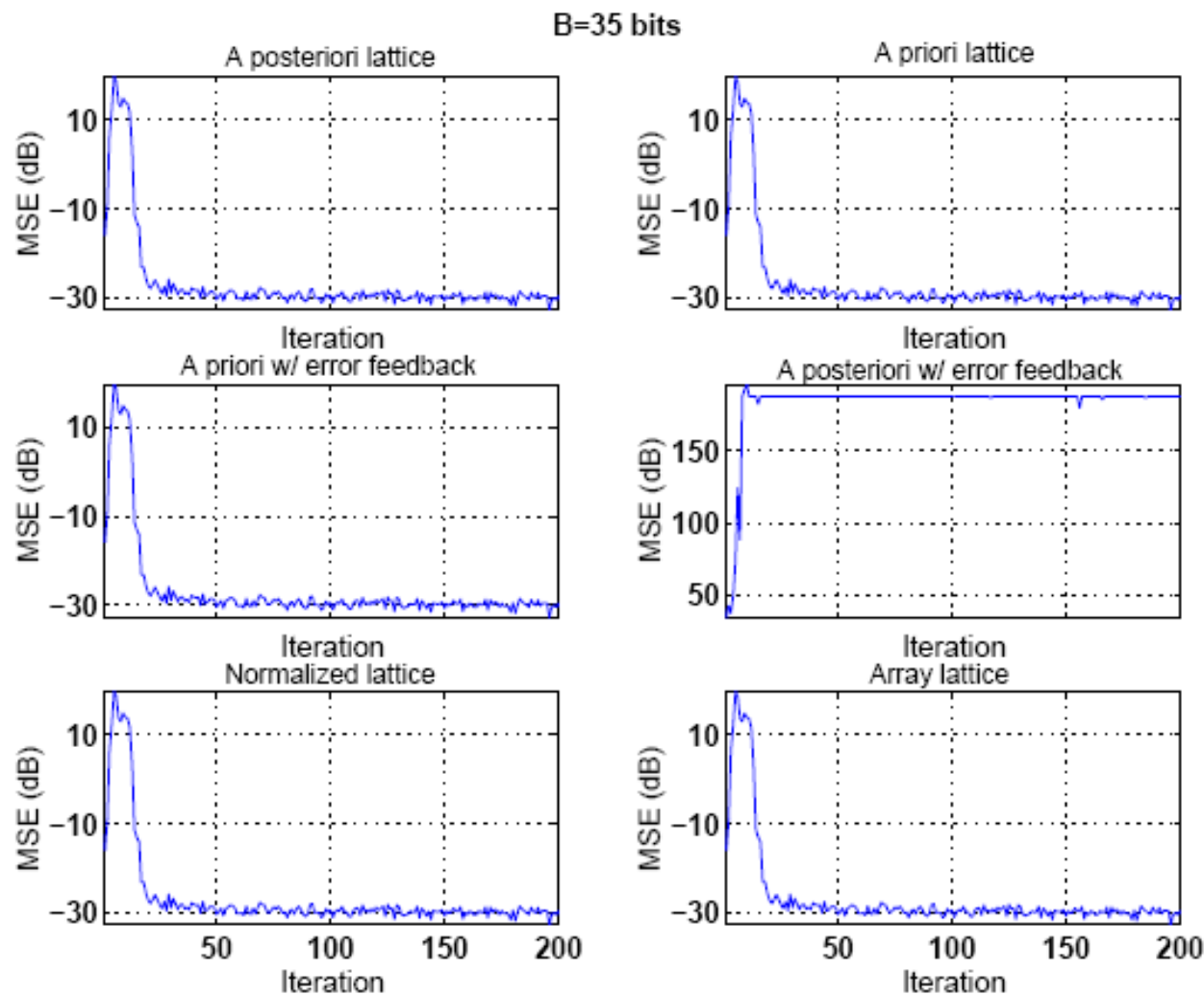
# COMPUTER PROJECT

**Project X.1 (Performance of lattice filters in finite precision)** The programs that solve this project are the following.

1. partA.m This program generates ensemble-average learning curves for the various lattice filters for different choices of the number of bits. The results are shown in Figs. 1 through 5. All filters work well at 35 bits, but some filters start facing difficulties at lower number of bits. It seems from the figures that the array lattice form is the most reliable in finite precision, while the *a posteriori* lattice form with error feedback is the least reliable.
2. partB.m This program generates ensemble-average learning curves for the various lattice filters in the presence of an impulsive interference at iteration  $i = 200$  and assuming floating-point arithmetic. The result is shown in Fig. 6. It is seen that all algorithms recover from the effect of the impulsive disturbance.
3. partC.m This program generates ensemble-average learning curves for the various lattice filters in the presence of an impulsive interference at iteration  $i = 200$  and assuming finite-precision arithmetic with  $B = 20$  and  $B = 16$  bits. The results are shown in Figs. 7 and 8.



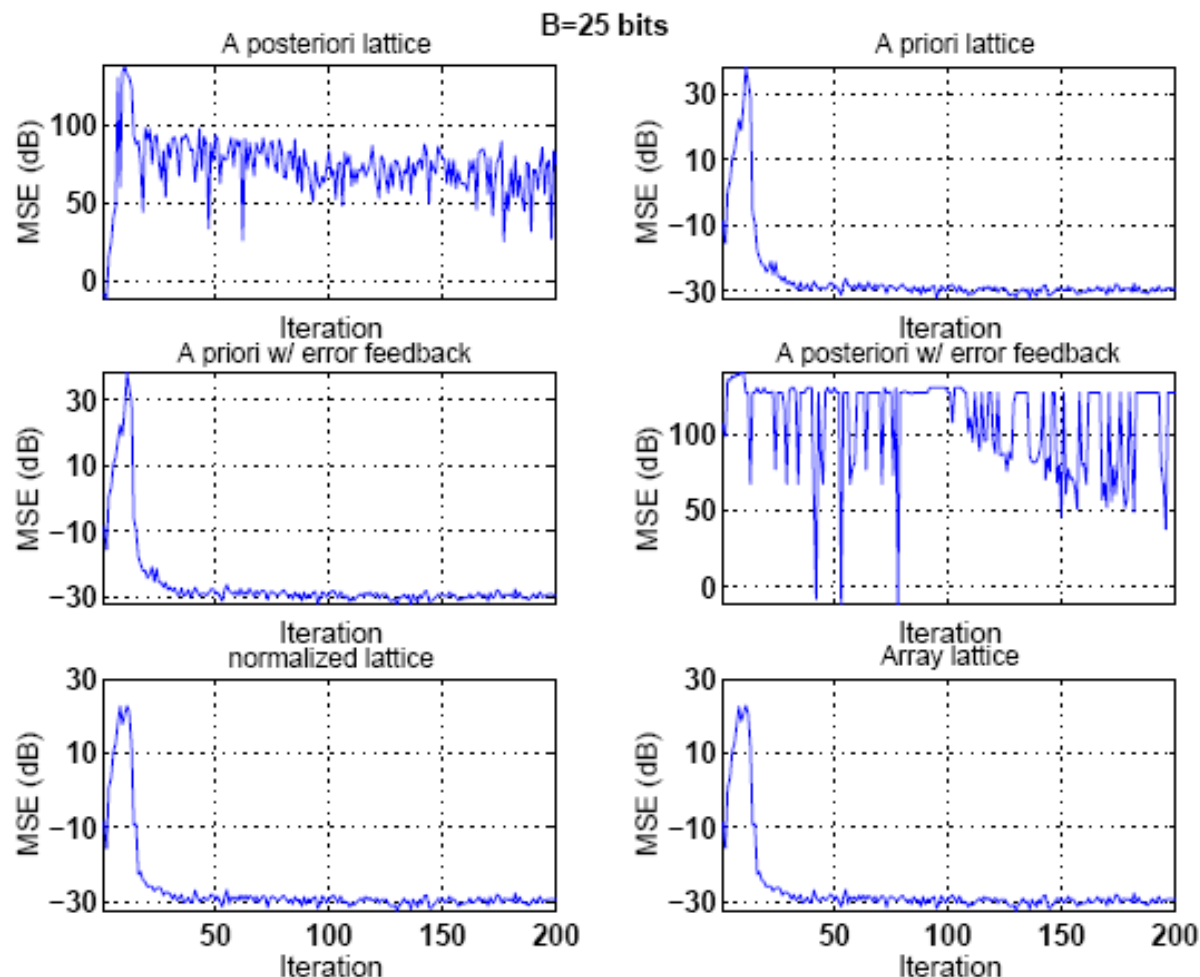
# COMPUTER PROJECT



**Figure X.1.** Ensemble-average learning curves for various lattice implementations using 35 bits.



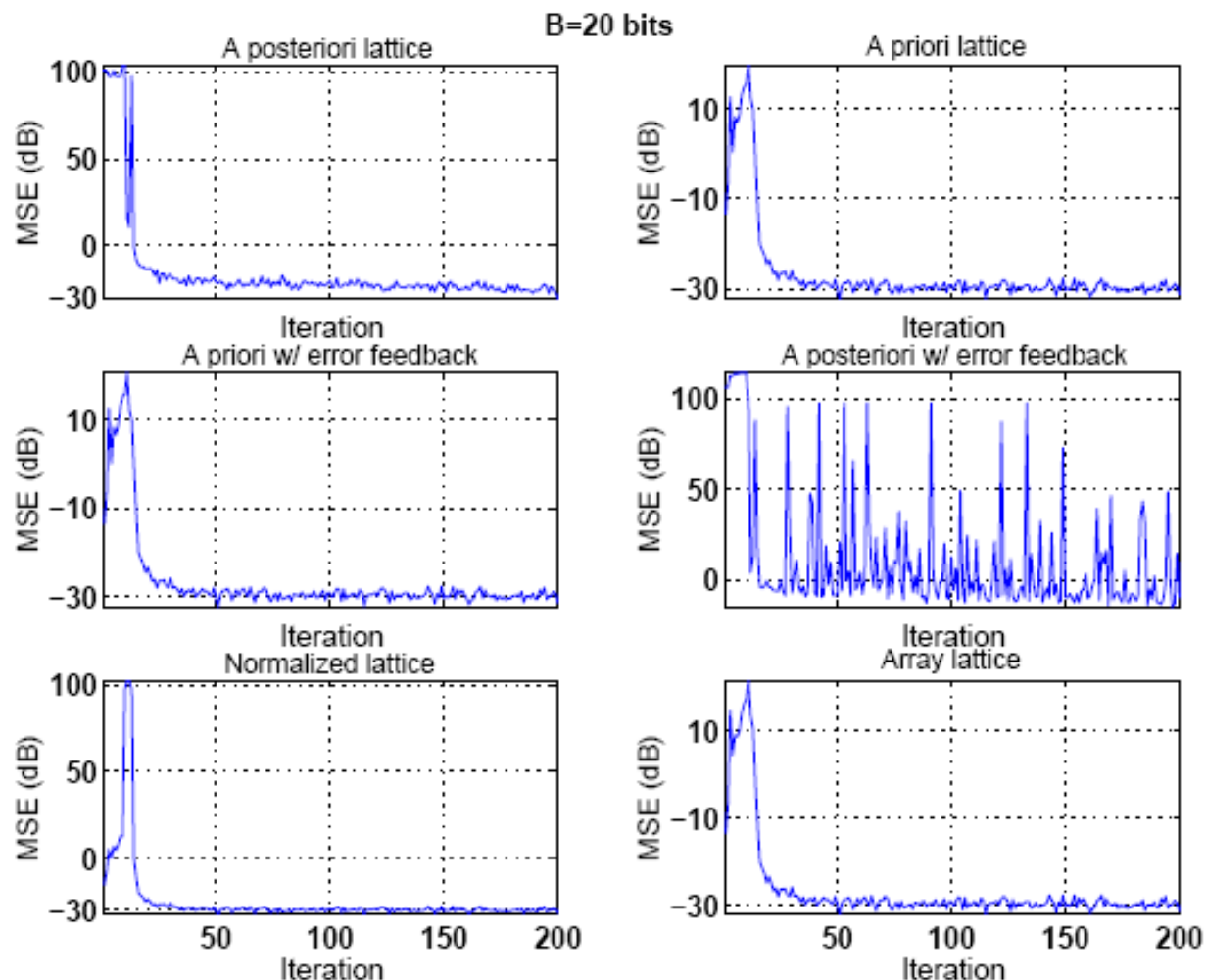
# COMPUTER PROJECT



**Figure X.2.** Ensemble-average learning curves for various lattice implementations using 25 bits.



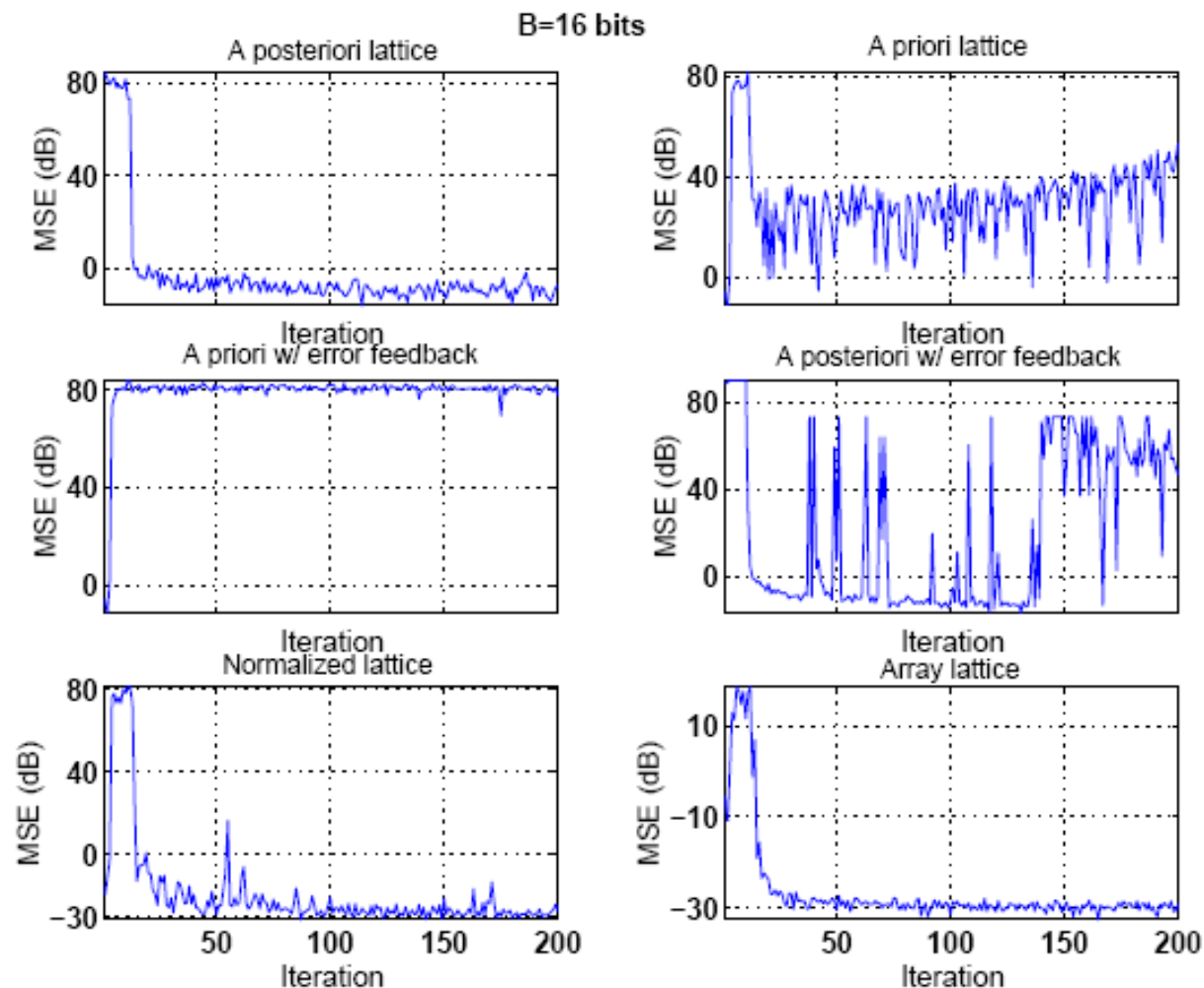
# COMPUTER PROJECT



**Figure X.3.** Ensemble-average learning curves for various lattice implementations using 20 bits.



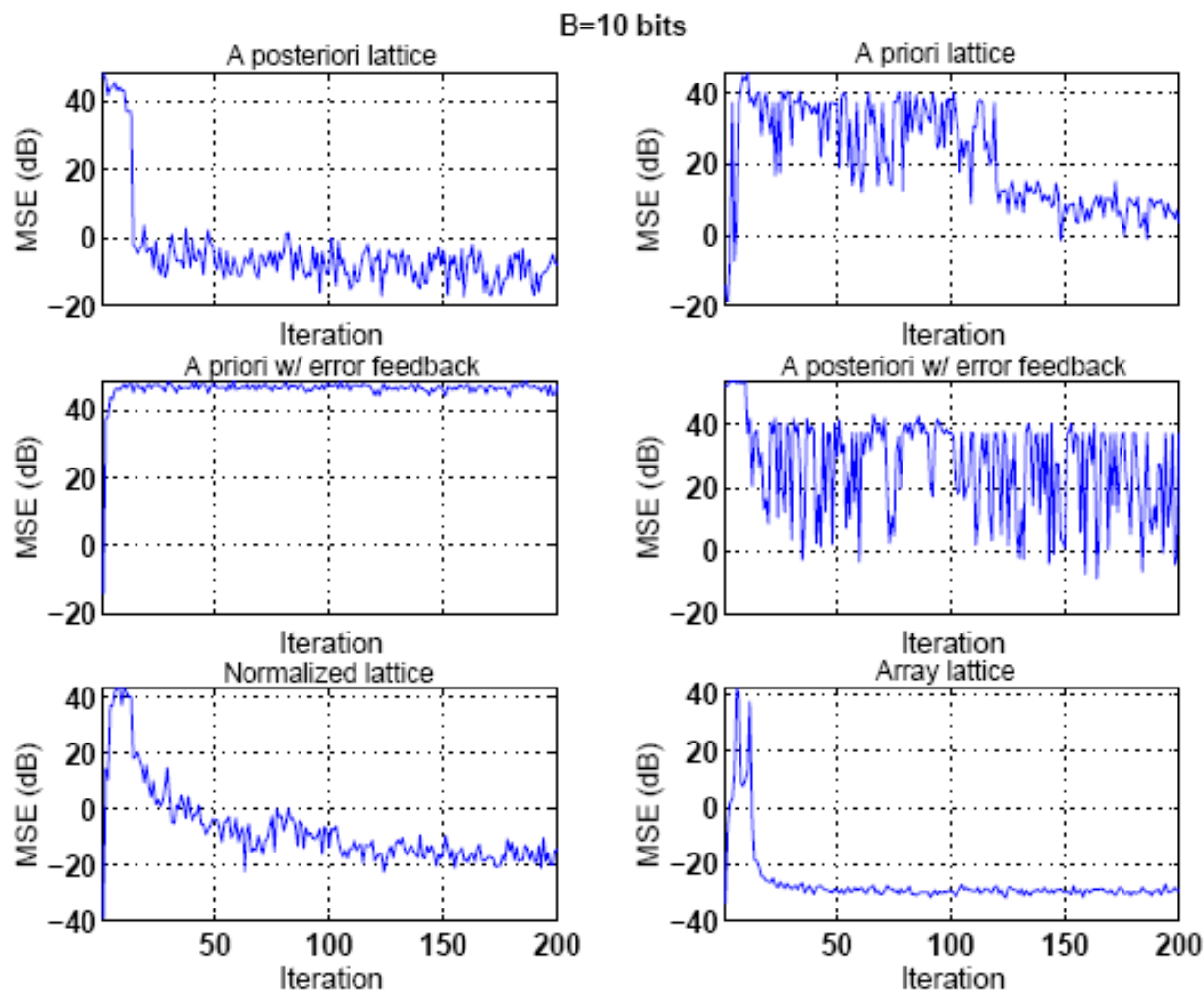
# COMPUTER PROJECT



**Figure X.4.** Ensemble-average learning curves for various lattice implementations using 16 bits.



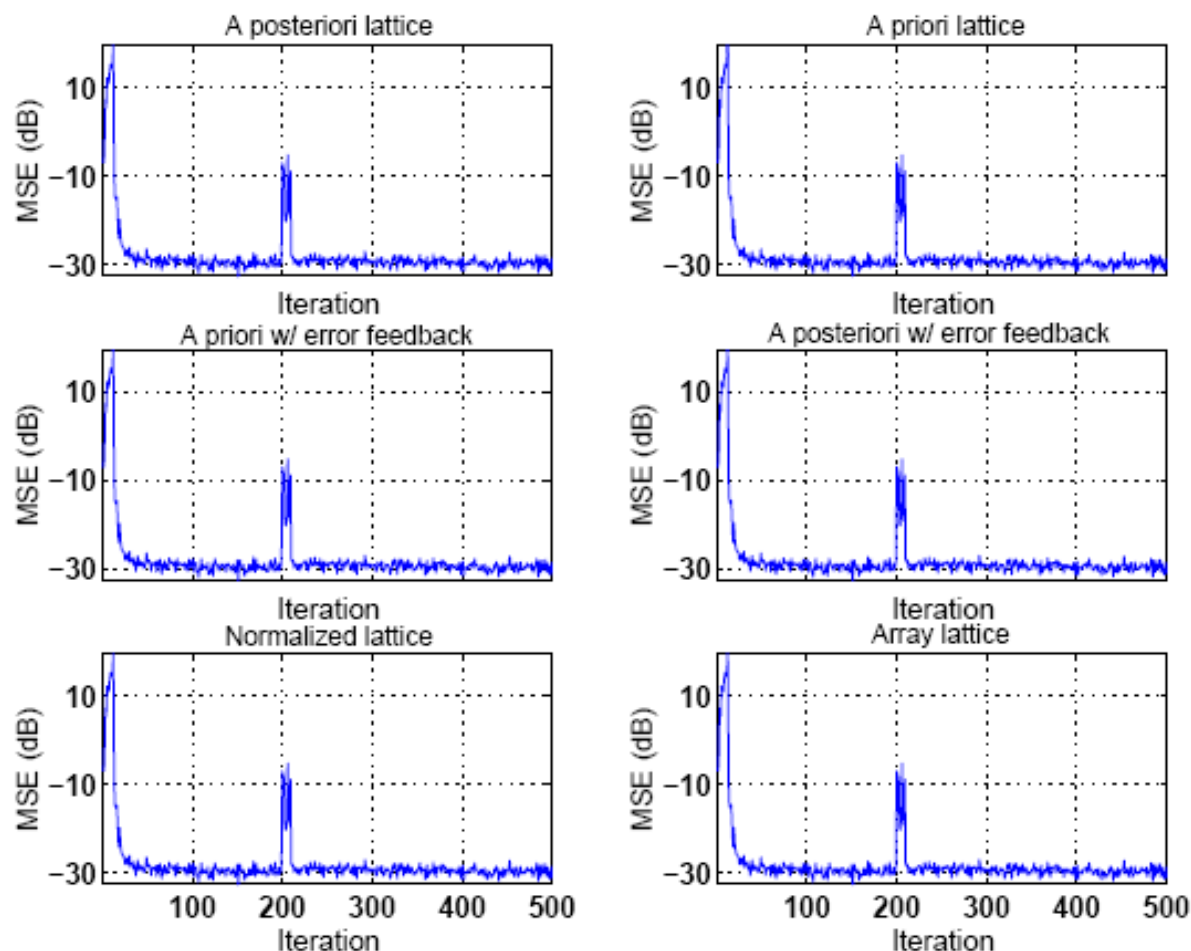
# COMPUTER PROJECT



**Figure X.5.** Ensemble-average learning curves for various lattice implementations using 10 bits.



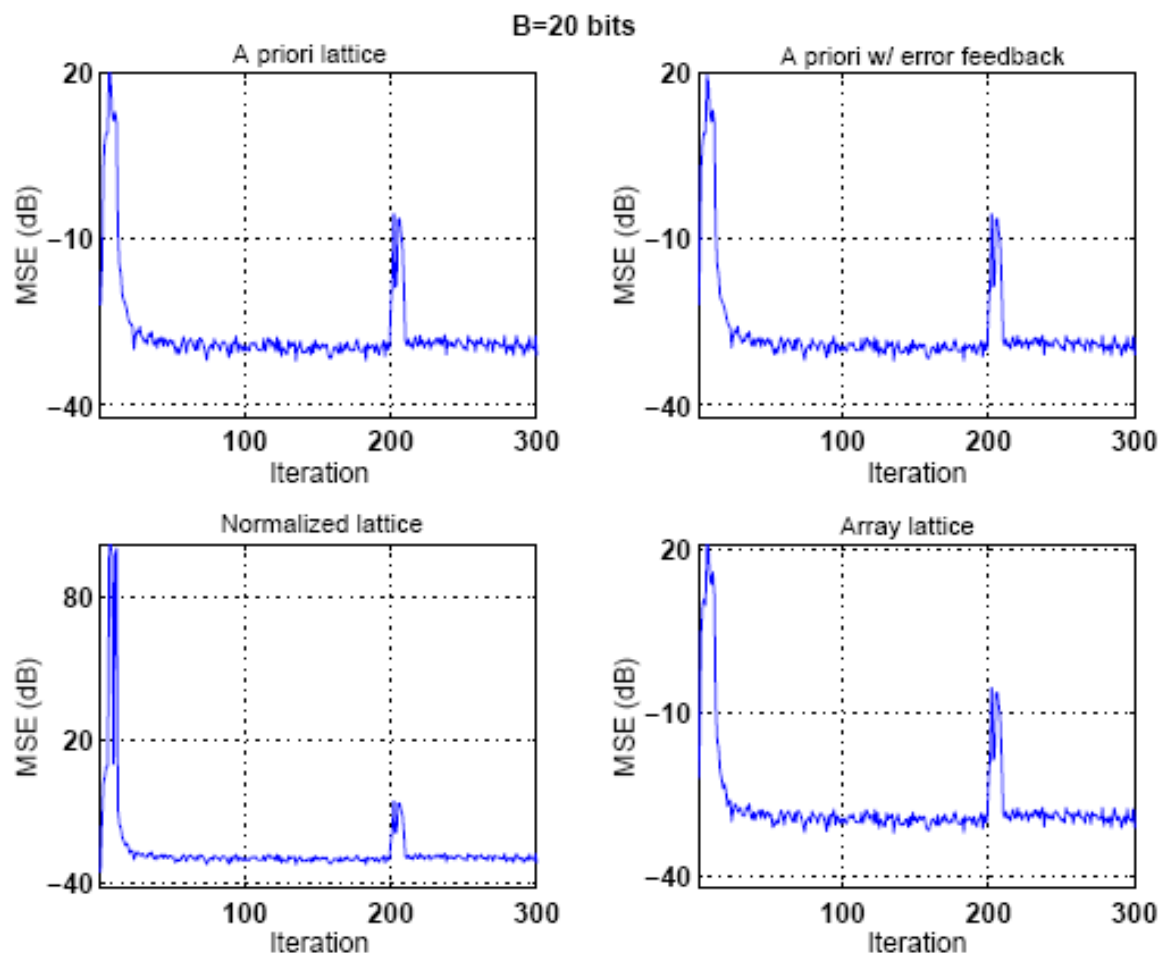
# COMPUTER PROJECT



**Figure X.6.** Ensemble-average learning curves for various lattice implementations in floating-point precision with an impulsive disturbance occurring at iteration  $i = 200$ .



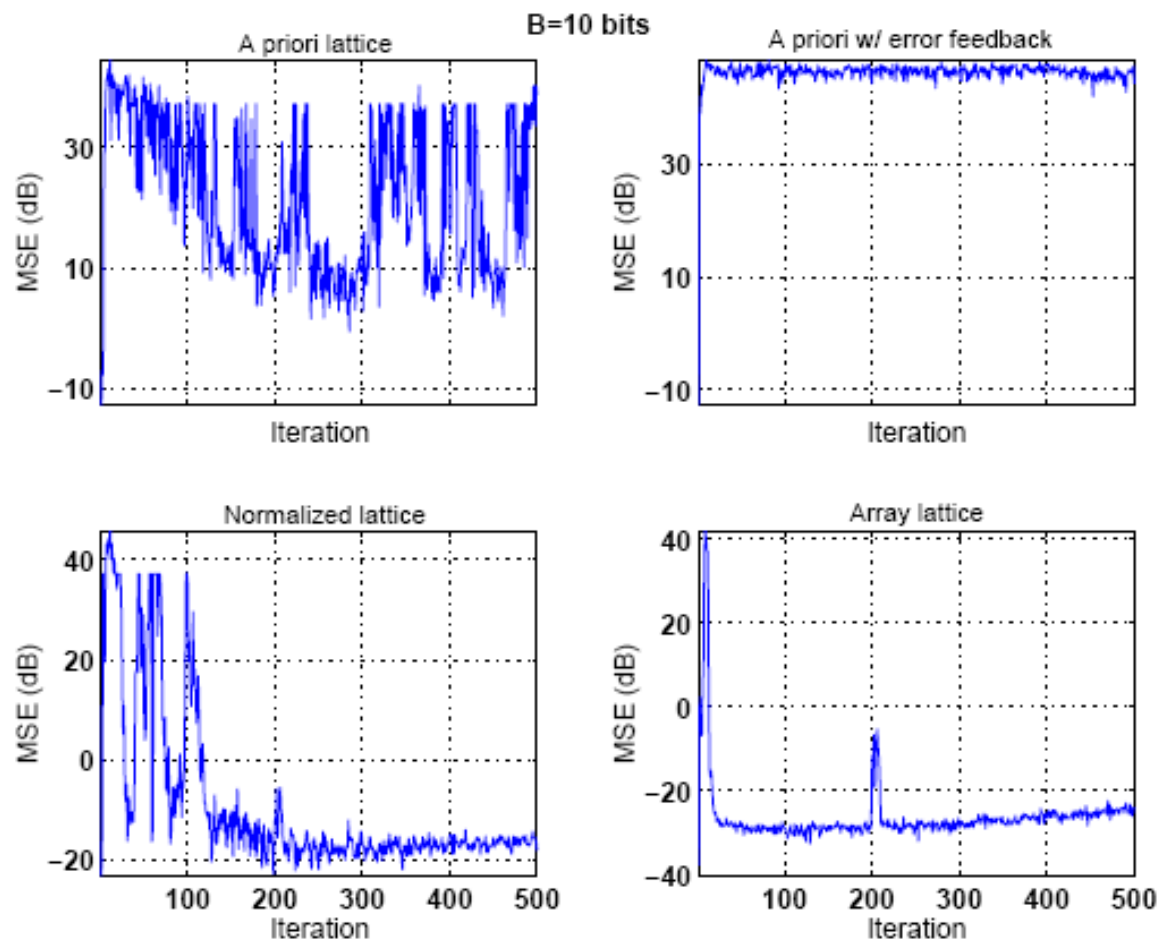
# COMPUTER PROJECT



**Figure X.7.** Ensemble-average learning curves for various lattice implementations in 20-bits precision with an impulsive disturbance occurring at iteration  $i = 200$ .



# COMPUTER PROJECT



**Figure X.8.** Ensemble-average learning curves for various lattice implementations in 10-bits precision with an impulsive disturbance occurring at iteration  $i = 200$ .