

N86 - 30072

D₃
10P.

On Realizations of Least-Squares Estimation
and Kalman Filtering by Systolic Arrays

18338

CD 146017

M.J. Chen and K. Yao

1. INTRODUCTION

Least-squares (LS) estimation is a basic operation in many signal processing problems. Given $y=Ax+v$, where A is a $m \times n$ coefficient matrix, y is a $m \times 1$ observation vector, and v is a $m \times 1$ zero mean white noise vector, a simple least-squares solution is finding \hat{x} which minimizes $\|Ax-y\|$. It is well known that for an ill-conditioned matrix A , solving least-squares problems by orthogonal triangular (QR) decomposition and back substitution has robust numerical properties under finite word length effect since 2-norm is preserved. Many fast algorithms have been proposed and applied to systolic arrays. Gentleman-Kung (1981) first presented the triangular systolic array for a basic Givens reduction. McWhirter (1983) used this array structure to find the least-squares estimation errors. Then by geometric approach, several different systolic array realizations of the recursive least-squares estimation algorithms of Lee et al (1981) were derived by Kalson-Yao (1985). We consider basic QR decomposition algorithms and find that under one-row time updating situation, the Householder transformation degenerates to a simple Givens reduction. Next, we derive an improved least-squares estimation algorithm by considering a modified version of fast Givens reduction. From this approach, the basic relationship between Givens reduction and Modified-Gram-Schmidt transformation can easily be understood. We also can see this improved algorithm has simpler computational and inter-cell connection complexities while compared with other known least-squares algorithms and is more realistic for systolic array implementation.

Minimum variance estimation (popularized by Kalman (1960)) is the generalized form of a least-squares problem, where the state vector x is characterized by the state equation $x_{k+1}=Fx_k+w$, the system noise w and the observation noise v are colored. The original algorithm presented by Kalman can have poor numerical property. Some algorithms for improving numerical properties, such as square-root covariance and square-root information methods have been studied. Now, we find that after the whitening processing, this minimum variance estimation can be formulated as the modified square-root information filter and be solved by the simple least-squares processing. This new approach contains advantages in both numerical accuracy as well as computational efficiency as compared to the original Kalman algorithm. Since all these processings can be implemented by systolic arrays, high throughput rate computation for Kalman filtering problems become feasible.

2. SIMPLE LEAST-SQUARES ESTIMATION

Given the equation $b=Ax+v$, it is well known that we can solve the least-squares solution \hat{x} by normal equation. However, this approach not only requires the computation of a matrix inverse but also doubles the condition number when we form $A'A$. Although using singular value decomposition for least-squares solution can improve numerical properties, the computational complexity involved in SVD is not low. Besides, fast algorithm for SVD is still underdevelopment. Lattice structure for least-squares solution was proposed and studied by Lee et al (1981). This approach was shown to have stable numerical property and regular hardware structure. However, this method required shifting property of the coefficient matrix and can not apply to all general cases. QR decomposition is another solution to obtain \hat{x} , since 2-norm is preserved by multiplying an orthogonal matrix Q , then by letting $QA=R$ be a upper triangular matrix, the \hat{x} can be obtained by using back substitution for the equation $Rx=b$. This approach has robust numerical properties since the 2-norm is fixed, the rounding error caused by finite word length effect will not grow. Basically, there are three ways for performing QR decomposition, namely, Householder transformation, Givens reduction, and Modified-Gram-Schmidt orthogonalization. It can be shown that under one row time updating situation (as in the systolic array implementation), the Householder transformation matrix will degenerate to a simple Givens reduction case.

Systolic array implementation for QR decompositions in least-squares estimation was first explored by Gentleman-Kung and followed by McWhirter and Kalson-Yao. By using a triangular systolic array, it was shown that the estimation error for the last observation can be solved at every clock period. The systolic array structure for least-squares estimation is shown in Figure 2.1. To achieve fully pipelined operation, the input rows are skewed and propagated like wavefronts in the diagonal direction. There are only two basic processing units, boundary cell and internal cell, are required by this systolic array. Communication between different processing units are all local. The properties of regularity and local communication are consistent with the philosophy of VLSI implementation. Summary of input/output formats and operation functions for two kinds of processing units are shown in Table 1 and Table 2 respectively.

Table 1. Input/Output format of systolic array algorithms

	BI_1	BI_2	BO_1	BO_2	II_1	II_2	IO_1	IO_2
Givens	σ	x	σ'	$d/d', x/d'$	$d/d', x/d'$	b	$d/d', x/d'$	b'
F-Givens	σ	x	σ'	$d/d', \sigma x/d'$ x	$d/d', \sigma x/d'$ x	b	$d/d', \sigma x/d'$ x	b'
M-G-S(I)	σ	x, e	σ'	$x/d, x/(1-\sigma)$ d	$x/d, x/(1-\sigma)$ d	b, e	$x/d, x/(1-\sigma)$ d	b', e'
M-G-S(II)	σ	x	σ'	$x/d', x/(1-\sigma)$	$x/d', x/(1-\sigma)$	b	$x/d', x/(1-\sigma)$	b'

The above symbols are for notations only, their physical meaning may change for different algorithms.

Table 2. Operational functions of processing units

	Boundary cells	Internal cells
Givens	$d' = (d^2 + x^2)^{1/2}$ $BO_2 \leftarrow d/d', x/d'$ $\sigma' \equiv (d/d') * \sigma$	$b' = (d/d') * b - (x/d') * k$ $k' = (x/d') * b + (d/d') * k$
F-Givens	$d' = d + (\sigma * x) * x$ $BO_2 \leftarrow (\sigma * x) / d', d/d'$ $\sigma' \equiv \sigma * (d/d')$	$b' = b - x * k$ $k' = (d/d') * k + (\sigma * x / d') * b$
M-G-S(I)	$d = e$ $BO_2 \leftarrow x/d, x/(1-\sigma)$ $\sigma' \equiv \sigma + (x/d) * x$	$k' = k + b * (x / (1-\sigma))$ $b' = b - k' * (x/d)$ $e' = e - k'^2 / d$
M-G-S(II)	$d' = d + (x / (1-\sigma)) * x$ $BO_2 \leftarrow x/d', x / (1-\sigma)$ $\sigma' \equiv \sigma + (x/d') * x$	$k' = k + b * (x / (1-\sigma))$ $b' = b - k' * (x/d')$

From systolic array point of view, the difference between algorithms proposed by McWhirter and Kalson-Yao lies in the basic computations in two kinds of processing units. Since these algorithms were derived from two different approaches, specifically Givens reduction and Modified-Gram-Schmidt orthogonalization, the basic relationship for these two QR decomposition methods under one row time updating can be compared as follows. First, we derived the modified expression for the fast-Givens reduction as given by

$$Q = \begin{bmatrix} (1/\sqrt{d})d, (1/\sqrt{d})dk_2, \dots, (1/\sqrt{d})dk_k \\ \sqrt{\sigma}x, \quad \sqrt{\sigma}b_2, \quad \dots \quad \sqrt{\sigma}b_k \end{bmatrix}$$

$$= \begin{bmatrix} (1/\sqrt{d'})d', (1/\sqrt{d'})d'k_2', \dots, (1/\sqrt{d'})d'k_k' \\ 0, \quad \sqrt{\sigma'}b_2', \quad \dots \quad \sqrt{\sigma'}b_k' \end{bmatrix},$$

the updating equation for this modified-fast-Givens algorithm becomes,

$$\begin{array}{lll} \text{Boundary cell:} & d' = d + x^2 / (1/\sigma) & (1/\sigma') = (1/\sigma) + x^2 / d \\ \text{Internal cell:} & b' = b - (x/d) * dk & d'k' = dk + b * x / (1/\sigma) \end{array} \quad [1]$$

By comparing the computational complexity between the fast Givens algorithm by Gentleman (1973) and that in [1], we can see [1] has one multi

plication less than the original algorithm. And since we do not have interest on the real rotated elements like $(1/\sqrt{d})dk_k$, we do not have the risk of dividing by a very small d . The numerical properties of the modified algorithm is then expected to be comparable to the numerical properties of the original one. By equation [1], the basic duality associations between Givens reduction and Modified-Gram-Schmidt orthogonalization is summarized in Table 3, which allows us to derive different algorithms for least-squares estimation from different approaches with efficiency.

Table 3. Duality association for M-G-S and Fast-Givens reduction.

M-G-S(II)	k_{mgs}	σ_{mgs}	x_{mgs}	b_{mgs}	d_{mgs}
F-Givens	$d_{fg} * k_{fg}$	$1 - \sigma_{fg}$	$\sigma_{fg} * x_{fg}$	$\sigma_{fg} * b_{fg}$	d_{fg}

With systolic array implementation, comparison of computational complexity for algorithms discussed above can be made by comparing the number of operations required in each processing unit. When the dimension of the coefficient matrix becomes large, wavefront array processing of Kung (1983) becomes more appropriate for the control scheme. In this case, the speed of this "wavefront" will be decided by the slowest processing unit along each wavefront. In modified fast Givens algorithm, equations for boundary cell are non-recursive and can be done in parallel if we can double the computational capability of each boundary cell. In this case, the wavefront speed and then the throughput rate can be doubled. The systolic array we discussed above will generate estimation error at each clock period. While the estimated vector \hat{x} is not shown explicitly, \hat{x} can be solved by back substitution which can be done by just appending a $n \times n$ identity matrix after the coefficient matrix A.

3. MINIMUM VARIANCE ESTIMATIONS AND KALMAN FILTERING

Often the signal vector x is a random process and can be modeled as a first order recursive equation. In this case, a first order recursive estimation (or Kalman filtering) problem can be stated as follows,

$$\begin{aligned} x_{k+1} &= Fx_k + w_k, \\ y_k &= Cx_k + v_k, \end{aligned} \tag{2}$$

where F and C are time-varying coefficient matrices with dimension $n \times n$ and $m \times n$ respectively. w_k is a $n \times 1$ and v_k is a $m \times 1$ zero mean noise vectors with known covariance matrices W_k and V_k respectively. It is assumed that noises w and v are uncorrelated and $E[w_i w_j] = E[v_i v_j] = 0$ for all $i \neq j$. Under the minimum variance criterion, we want to find \hat{x}_k for all k , such that $E\|(x_k - \hat{x}_k)^2\|$ is minimized. Kalman showed that \hat{x}_k can be obtained by the recursive algorithm given as

$$\begin{aligned} \hat{x}_k &= F \hat{x}_{k-1} + K [y_k - C F \hat{x}_{k-1}], \\ K_k &= P_k \{ C^T (C P_k C^T + V) \}^{-1}, \\ \text{where } P_k &= F P_{k-1} F^T + W, \\ P_k &= P_k - K_k C P_{k-1}. \end{aligned} \quad [3]$$

The information matrix is defined as the inverse of the error covariance matrix P . Besides [3], it is shown that instead of propagating the error covariance matrix, the Kalman filtering problem can be solved by propagating the information matrix during the iterations. Both covariance and information filters are recursive since the current updating depends only on results from previous stage. The choice between covariance filter and information filter depends on the values of n and m . When $n > m$, which is usually the case, the original Kalman filtering is chosen to avoid the inverse of the $n \times n$ matrix. However, Kalman algorithm is known for its poor numerical properties, especially for non-observable coefficient matrices. The original Kalman filter needs an approximate $O(n^3)$ multiplication time for each iteration. If $m > 1$, computation of a matrix inversion is inevitable. Since all equations are sequential in manner, if real time computation is required for a Kalman filtering problem, some modifications must be done to insure the capability for parallel computation. Among many possible modified algorithms, square-root filtering have been proved to have computational efficiency and robust numerical properties under finite word length effect (Kaminski 1971). The main advantage of the square root filter is that we can handle the covariance matrix by its square root form which has condition number smaller than the original one. Therefore, for ill-conditioned problems, when we used the square root filter with a single precision machine, we can expect the same numerical result as if we have used the original algorithm on a double precision machine. Updating processings for both square root covariance filter and square root information filter can be expressed in matrix forms and handled by the QR decomposition method which is capable of systolic array implementation. However, only square-root information filter allows us to update the estimated state vector as well as the information matrix by using the same transformation matrix Q . When both updated covariance matrix and state vector are important to us, we find square-root information filter is a better solution for the systolic array implementation. The square-root information filter requires computation of the inverse of the coefficient matrix F , which will cause bad numerical properties for F being near singular. One version of the square root information matrix method for Kalman filtering was considered by Paige and Saunders (1977). It is shown that by using whitening processing through Cholesky decomposition, the Kalman filtering can be represented as a simple least-squares problem. This approach does not require the computation of the inverse of the matrix F and is more suitable for systolic array implementation.

The whitening processing can be briefly described as below. Assume $W = \tilde{L} \tilde{L}^T$ and $V = \tilde{L} \tilde{L}^T$ are the Cholesky decomposition of covariance matrices W and V . With $W^{-1} = \tilde{L}^{-1} \tilde{L}^{-T}$ and $V^{-1} = \tilde{L}^{-1} \tilde{L}^{-T}$, it can be proved that $L_w = \tilde{L}^{-1}$ and $L_v = \tilde{L}^{-1}$. $\tilde{w}_k = L_w^T w_k$ and $\tilde{v}_k = L_v^T v_k$ are whitened noises with identity covariance matrices.

Denote $\tilde{F} = L_w^T F$, $\tilde{C} = L_v^T C$, and $\tilde{y}_k = L_v^T y_k$. We can express the whitened system equations in the matrix-vector form as

$$\begin{bmatrix} 0 \\ \bar{y}_1 \\ 0 \\ \vdots \\ \bar{y}_k \end{bmatrix} = \begin{bmatrix} -L \\ F_1 C_W^{-1} & & 0 \\ & \ddots & \\ & & C_W^{-1} \\ 0 & & & \bar{C} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_k \end{bmatrix} + \begin{bmatrix} \bar{v}_1 \\ \bar{v}_2 \\ \vdots \\ \bar{v}_k \end{bmatrix} \quad [4]$$

Since the noise vector in [4] has zero mean and identity covariance matrix, we can get $X_{\text{min}} = [\hat{x}_1, \dots, \hat{x}_k]$ by solving [4] as a LS problem. After applying QR decomposition to [4] at time k, we have

$$\begin{bmatrix} R_1 & & & & 0 \\ & R_{12} & & & \\ & R_2 & R_{23} & & \\ \cdot & & & & \\ \cdot & & & & \\ 0 & & & R_{k-1} & R_{k-1,k} \\ & & & & R_k \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \\ \vdots \\ \cdot \\ x_{k-1,k} \\ x_{k,k} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ y_{k-1} \\ y_k \end{bmatrix} \quad [5]$$

We can see that $R_i, i=1,2,\dots,k$, in [5] are all upper triangular matrices, and \hat{x}_k , the optimum estimated vector at time k, depends only on the last line, i.e., $R_k \hat{x}_k = y_k$. Furthermore, at $T=k+1$, the updating equation depends on the last row of [5] only. That is, the QR decomposition at $T=k+1$ only depends on a $(2n+m) \times (2n+1)$ matrix as in [6]. When the QR decomposition of [6] is completed, we have \bar{R}_{k+1} (upper triangular) and y_{k+1} ready for iteration of next stage.

$$Q \begin{bmatrix} \bar{R}_k & 0 & \bar{y}_k \\ F^k & -L^k & 0^k \\ 0 & C^k & \bar{y}_k \end{bmatrix} = \begin{bmatrix} R^k & R_{k,k+1} & y_k \\ 0^k & R_{k+1} & y_{k+1} \\ 0 & 0^{k+1} & *_{k+1} \end{bmatrix} \quad [6]$$

where * is the term used to compute the residue.

The upper triangular matrix \bar{R}_k can be shown to be the square-root of the inverse of the error covariance matrix $P_k = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)']$. That is, this algorithm, which propagates the square root information matrix for next iteration, is actually a modified square-root information filtering.

4. SYSTOLIC ARRAY IMPLEMENTATIONS FOR KALMAN FILTERING

From last section, we can see that the basic operations for square root Kalman filtering can be described in two parts. The first one, whitening processing includes operations such as Cholesky decomposition, inverse of triangular matrix, and matrix multiplication. Secondly, the QR decomposition is applied. Obviously, these two parts can be operated in parallel. That is, we can start the whitening processing for the $(k+1)$ st iteration as well as the QR decomposition for the k -th iteration at the same time in a pipelined manner.

The original square-root information filter involves the computation of the inverse of the coefficient matrix F which not only increases the computational complexity but also causes bad numerical properties when coefficient matrix F is singular or near singular. This shortcoming can be

recovered by choosing the modified square root information filtering in [4]. As shown from [4]-[6], formulation of the modified square-root information filter involves only multiplication between coefficient matrices and the inverse of the square root noise covariance matrices. For noise with positive definite covariance, square root covariance matrix always exists.

4.1 Whitening Processing

The whitening processing is done by multiplying the coefficient matrix with a whitening operator L' where $(LL')^{-1}$ is the given covariance matrix of the additive noise. Since a covariance matrix is a positive definite symmetric matrix, the square root matrix can be obtained by the Cholesky decomposition. A triangular systolic array for Cholesky decomposition is designed for this purpose with outputs skewed to match the input format of the QR systolic array.

The inversion of an upper triangular matrix is simple after we built the basic systolic array for QR decomposition. The idea for the inversion of an upper triangular matrix is the same as solving the back substitution.

With $UU^{-1}=I$, let $U^{-1}=[u_1, u_2, \dots, u_n]$, with u_i being an $n \times 1$ column vector. A matrix inversion can be divided into n sets of linear equations, each having the form of $Uu_i = e_i$, $i=1,2,\dots,n$, where e_i is an $n \times 1$ column vector with i^{th} element equals to 1, and all others being 0, and can be solved by a systolic array.

4.2 QR Decomposition for Kalman Filtering

Equation [6] suggests that \hat{x}_k can be solved as a least-squares solution by a $2n \times 2n$ QR_systolic array. However, serious delay will be caused by the fact that R_k and R_{k+1} are not in-place computations. That is, we have trouble to move the newly formed R from the upper-right corner to the lower-left corner in our triangular array for the next iteration. That is, the computation at stage $k+1$ can not start until the last element of R_k is completed. In this "waiting" period, most of processing units are idle and the pipeline is empty. It will cause delay for at least $2n$ clock periods.

This disadvantage can be overcome by in-place computations for \bar{R}_k and \bar{R}_{k+1} . This can be done by partitioning the original matrix into two strips, and perform the partitioned QR decomposition by the systolic array structure proposed in Figure 2. In this approach, an $n \times n$ QR systolic array as well as a rotation array which consists of $n \times (n+1)$ internal cells are used. Once elements of \bar{R}_k are formed, it is ready to be used for computations at stage $k+1$. Here we need only to pass transformed elements generated by the first strip to the rectangular rotation array for the pre-processing of the second strip. This input format is shown in Figure 3. Since all these can be done in fully pipelined manner and in-place computations are obtained, complicated inter-cell connection and control scheme can both be avoided. To obtain the estimated value \hat{x}_k , we can just append an identity matrix I after the second strip, and we get result every $3n+m$ clock periods.

5. CONCLUSION

In this paper, we first survey existing algorithms for least-squares estimations by systolic arrays. Basic comparisons are made based on computation and inter-cell connection complexities of elementary units. Finally, by choosing the square-root information filtering algorithm, we showed a simple way to solve the Kalman filtering as a least-squares problem that can be processed by systolic arrays. Systolic array for Cholesky decomposition is also proposed for whitening processing. By manipulating the data properly, the Kalman filtering can be processed under fully pipelined manner. There is no special constraint on our system equations and standard time-varying coefficient matrices and non-stationary colored noises are assumed in our model. Most of the processing units we need for this square root information filter do not involve square-root computations. The only exception is the computations for the Cholesky decomposition. However, for pipelined operation between whitening processing and QR decomposition, the later certainly involved more computational work than the former. Since there is only n square-root computations required in each iteration as compared with the operations required for QR decomposition, Cholesky decomposition will not become the bottleneck for this algorithm. For many real life problems where we can assume noises are stationary, then covariance matrices W and V are fixed during our operation. In this case, inversed square-root covariance matrices can be obtained by pre-processing and our Kalman filtering can be solved as a simple least-square problem. Since all operations can be performed by the designed systolic array processing, which have the input/output formats matched to each other, the entire hardware design can be viewed as a pipelined structure. The estimated vector can be obtained with the $O(n)$ in time while compared with the $O(n^3)$ for the original Kalman filter. Finally, since this is a square root matrix operation, good numerical property can also be obtained.

This work is partially supported by NASA/Ames Research Contract NAG-2-304

REFERENCES

- Bierman G 1977 Factorization Methods for Discrete Sequential Estimation, Academic Press, NY
- Bierman G 1982 Square-Root Information Filtering and Smoothing for Precision Orbit Determination, Math. Programming Study 18
- Bjorck A 1967 Solving Linear Least Squares Problems by Gram-Schmidt Orthogonalization, BIT 7 1-21
- Duncan D B and Horn S D 1972 Linear Dynamic Recursive Estimation from the Viewpoint of Regression Analysis, J. ASA 67 815-821
- Dyer P and McReynolds S 1969 Extension of Square-Root Filtering to Include Process Noise, J. Opt. Theory and Appl. 6 444-459
- Gentleman W M 1975 Error Analysis of QR decompositions by Givens Transformations, Linear Algebra and Its Applications, 10 189-197
- Gentleman W M 1973 Least Squares Computation by Givens Transformations Without Square Roots, J. Inst. Math. Appl. 12 329-336

ORIGINAL PAGE IS
OF POOR QUALITY

- Gentleman W M and Kung H T 1981 Matrix Triangularization by Systolic Arrays, Proc. SPIE, Real-Time Signal Processing IV, 298 19-26
- Hammarling S 1974 A Note on Modifications to the Givens Plane Rotation, J. Inst. Maths Appl. 13 215-218
- Kalman R E 1960 A New Approach to Linear Filtering and Prediction Problems, J. Basic Engineering, 82 35-45
- Kalson S and Yao K 1985 Systolic Array Processing for Order and Time Recursive Generalized Least-Squares Estimation, Proc. SPIE, Real-Time Signal Processing VIII, 564 28-38
- Kaminski P G et al 1971 Discrete Square Root Filtering: A Survey of Current Techniques, IEEE Tran. on Auto. Control, 6 727-736
- Kung H T 1982 Why Systolic Arrays, IEEE Computer, 15 37-46
- Kung S Y 1983 VLSI Design for Massively Parallel Signal Processors, Microprocessors and Microsystems
- Lawson C and Hanson R 1974 Solving Least Squares Problems, Prentice-Hall, NJ
- Lee D T, Morf M and Friedlander B 1981 Recursive least square ladder estimation algorithms, IEEE Tran. ASSP, 3 627-641
- Ling F and Proakis J 1984 A Recursive Modified Gram-Schmidt Algorithm with Application to Least Squares Estimation and Adaptive Filtering, Int. Sym. on Circuit and System
- Mead C and Conway L 1980 Introduction to VLSI systems, Addison-Wesley, Mass.
- McWhirter J G 1983 Recursive least-Squares Minimization Using a Systolic Array, Proc. SPIE, Real-Time Signal Processing VI, 431 103-112
- Paige C C and Saunders M A 1977 Least Squares Estimation of Discrete Linear Dynamic Systems Using Orthogonal Transformation, SIAM J. Numer. Anal., 14 180-193

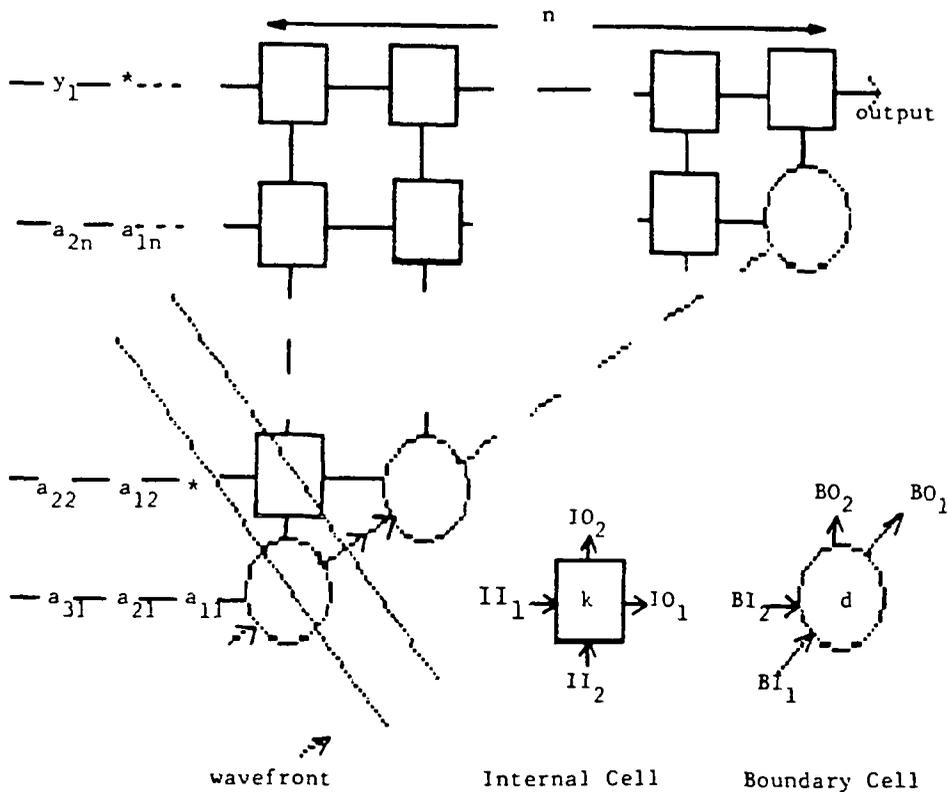


Figure 1: Systolic array for least-squares estimation.

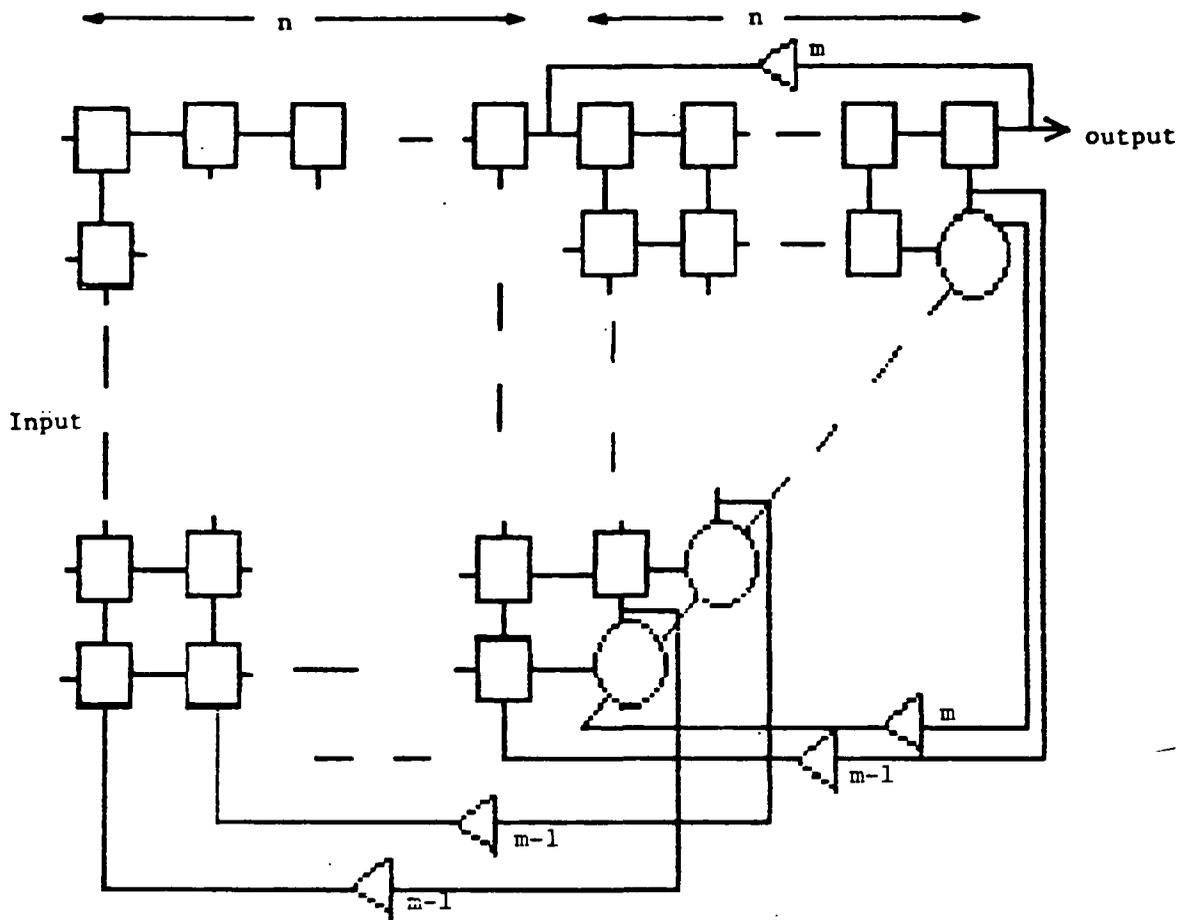


Figure 2: QR systolic array for Kalman filtering

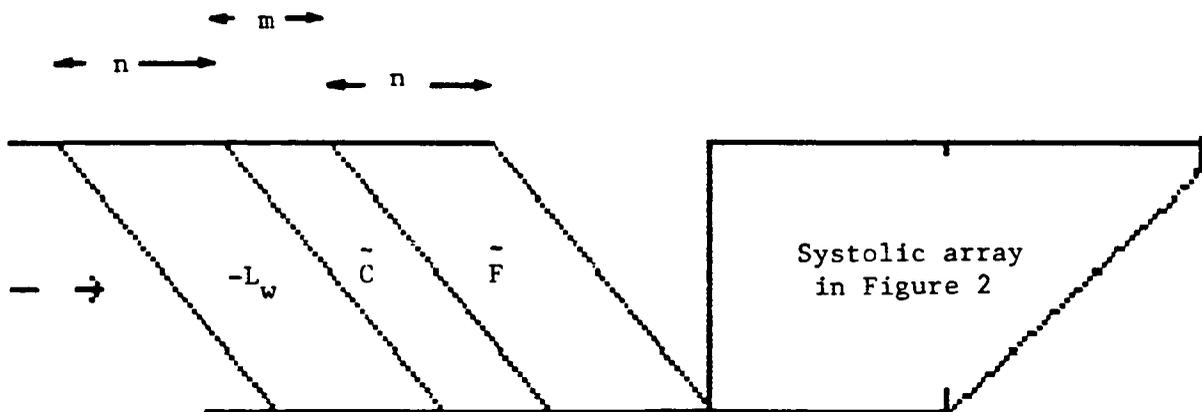


Figure 3: Input format for systolic array Kalman filtering