Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

Subspace System Identification Using a Multichannel Lattice Filter*

Neil Yi-Nan Chen
Mechanical and Aerospace Engineering
University of California, Los Angeles
Los Angeles, CA 90095-1597, USA
neilchen@ucla.edu

James S. Gibson
Mechanical and Aerospace Engineering
University of California, Los Angeles
Los Angeles, CA 90095-1597, USA
gibson@ucla.edu

Abstract— This paper presents a new class of subspace algorithms for system identification, in which the projection of future data onto past data is performed by a multichannel least-squares lattice filter with parameterizations and channel arrangements particularly suited to subspace system identification. Because an adaptive lattice filter provides the core computational engine of the new subspace algorithms, the main computational burden of the algorithms can be executed in real time, as data is being sampled. This approach produces much faster system identification algorithms than existing batch algorithms for subspace identification.

I. INTRODUCTION

In recent years, a class of methods usually referred to as subspace methods [1], [2], [3] has become recognized as the best approach to system identification in the presence of high levels of broad-band noise and unmodeled disturbance. All such methods require identification by some least-squares algorithm of high-order prediction models that predict future outputs from past inputs and outputs. The coefficient matrices in the identified prediction models are used to build a Hankel matrix, often weighted in some fashion by the original data, and then a state-space realization of the estimated system is constructed from a singular-value decomposition of the Hankel matrix. Because at least several thousand data points are required to obtain nearly unbiased estimates of unknown systems from noisy data, the least-squares estimation of the prediction models is the most computationally intensive part of a subspace identification algorithm. Least-squares estimation is always equivalent to orthogonal projection, and the preferred method for computing the projections required in subspace identification has come to be QR factorization of certain data matrices.

Previous subspace algorithms have used batch (i.e., nonadaptive) least-squares parameter estimation to determine certain Hankel matrices from the data. The required least-squares solutions have been obtained by batch QR factorization of certain data matrices. This paper presents a class of subspace algorithms that use least-squares lattice filters to perform adaptive QR factorization of data matrices and estimate Hankel matrix.

For lattice-filter based subspace algorithms, we have developed a method for using our lattice filters[4] to construct the triangular matrix R in the QR factorization of a Toeplitz data matrix W whose columns are shifted versions of the input and output data sequences. For subspace system identification, only this matrix R is needed. Although, it is generally understood among adaptive-filtering researchers that RLS lattice filters perform a block orthogonalization of the columns of the data matrix, this orthogonalization is only implicit and sometimes only approximate. Our RLS lattice filters are unique in that they perform a complete orthogonalization (not just a block orthogonalization) of the data matrix, and this orthogonalization is exact even for short data sequences because of the unwindowed initialization of our algorithms. However, even with our basic lattice filters, this orthogonalization is still implicit. Our procedure for constructing the R matrix for the QR factorization from the reflection coefficients and residual errors in the lattice filter is a recent discovery-which we believe is one of our most significant discoveries over the past few years. As far as we know, it has not been possible to construct such an R matrix with any previous lattice filters.

The outline of this paper is as following: Section II describes the concepts of the subspace system identification methods. Section III discusses the adaptive lattice filter we will use in this paper and the implementation of the subspace method. An example for comparison is shown in Section IV. In Section V, we conclude the research which has been done.

II. SUBSPACE SYSTEM IDENTIFICATION

Fig. 1 illustrates the basic steps in subspace identification of a state space model of a linear time invariant (LTI) system.

A. Kalman Preditor

Consider the LTI system

$$\begin{aligned} x(t+1) &= A_0 x(t) + B^u u(t) + B^w w(t)\,, \\ y(t) &= Cx(t) + v(t)\,, \end{aligned} \tag{1}$$

where A_0 is a stable matrix. The sequences w, and v are stationary zero-mean white-noise sequences. Only the sequences y and u are measured. Our goal is to use the measured y and u to identify unbiased system matrices (A_0, B^u, C).

*This research was supported by AFOSR Grants F-49620-02-1-0319 and F-49620-03-1-0234

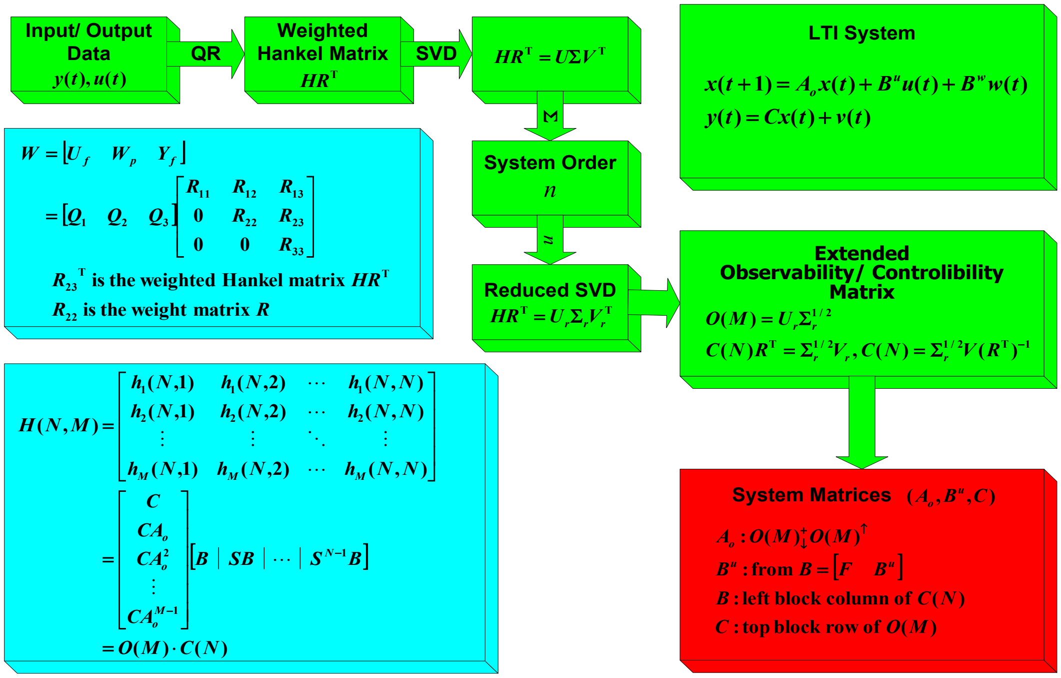

Fig. 1. Subspace State-Space System Identification.

Consider the LTI system in (1) and the $(M+1)$-step ahead Kalman predictor

$$\hat{y}_M(t) = \sum_{i=1}^{t} h_M(t,i) \begin{pmatrix} y(t-i) \\ u(t-i) \end{pmatrix}, \tag{2}$$

$$h_M(t_n,i) = CA_o^{M-1} S(t_n,i) [\, F(t_n - i) \quad B^u \,], \tag{3}$$

where $S(t_n,i)$ is function of A_o, C and F.

Define the Hankel matrix at given time t_n as

$$H_{M,N} = \begin{bmatrix} h_1(t_n,1) & h_1(t_n,2) & \dots & h_1(t_n,N) \\ h_2(t_n,1) & h_2(t_n,2) & \dots & h_2(t_n,N) \\ \vdots & \vdots & \ddots & \vdots \\ h_M(t_n,1) & \dots & \dots & h_M(t_n,N) \end{bmatrix}. \tag{4}$$

It is true that

$$H_{M,N} = \mathcal{O}(M)\mathcal{C}(N), \tag{5}$$

where $\mathcal{C}(N)$ and $\mathcal{O}(M)$ have the following form

$$\mathcal{C}(N) = [\ B(t_n) \mid \dots \mid S(t_n,n)B(t_n - n + 1)\ ], \tag{6}$$

with $B(t)$ defined as $[F(t)\ \ B^u]$, and

$$\mathcal{O}(M) = \begin{bmatrix} C \\ CA_o \\ \vdots \\ CA_o^{M-1} \end{bmatrix}, \tag{7}$$

also

$$\mathcal{O}^{\uparrow}(M) = \begin{bmatrix} CA_o \\ CA_o^2 \\ \vdots \\ CA_o^{M-1} \end{bmatrix}, \ \mathcal{O}_{\downarrow}(M) = \begin{bmatrix} C \\ CA_o \\ \vdots \\ CA_o^{M-2} \end{bmatrix}. \tag{8}$$

Suppose that $U\Sigma V^*$ is a minimal singular-value decomposition of $H_{M,N}$. Use Σ to identify the system order n. Define the first n columns of U and V are U_r and V_r, and the diagonal matrix with the first n singular values in Σ is Σ_r. Then

$$\mathcal{O}(M) = U_r \Sigma_r^{1/2}, \qquad \mathcal{C}(N) = \Sigma_r^{1/2} V_r^*. \tag{9}$$

We can get the system matrices B, C and Kalman gain matrix $F(t)$ from

$$C \quad = \quad \text{first top block row of } \mathcal{O}(M), \tag{10}$$

$$[F(t)\ \ B^u] \quad = \quad \text{first left block column of } \mathcal{C}(N). \tag{11}$$

The matrix A_o can be obtained by

$$A_o = \mathcal{O}_{\downarrow}(M)^{\dagger} \mathcal{O}^{\uparrow}(M). \tag{12}$$

B. Subspace Method

For the input sequence $u(t)$ and the output sequence $y(t)$, define the matrices of past data W_p, future input

U_f and future output Y_f as

$$W_p = \begin{bmatrix} Y(t-1) & \dots & Y(t-N) \\ \vdots & \ddots & \vdots \\ Y(N) & \dots & Y(1) \\ Y(N-1) & \dots & Y(0) \end{bmatrix}, \quad (13)$$

$$U_f = \begin{bmatrix} u(t+M-1) & \dots & u(t) \\ \vdots & \ddots & \vdots \\ u(M+N) & \dots & u(N+1) \\ u(M+N-1) & \dots & u(N) \end{bmatrix}, \quad (14)$$

$$Y_f = \begin{bmatrix} y(t+M-1) & \dots & y(t) \\ \vdots & \ddots & \vdots \\ y(M+N) & \dots & y(N+1) \\ y(M+N-1) & \dots & y(N) \end{bmatrix}. \quad (15)$$

N is the number of history data used and M is the number of future data used.

Apply the QR factorization to the data matrix W,

$$\mathbf{W} = [\, U_f \quad W_p \quad Y_f \,] \quad (16)$$

$$= [\, Q_1 \quad Q_2 \quad Q_3 \,] \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ 0 & R_{22} & R_{23} \\ 0 & 0 & R_{33} \end{bmatrix}. \quad (17)$$

It can be proved that the Hankel matrix in (5) will converge to the following:

$$H_{M,N} = ({R_{22}}^{-1} R_{23})^T . \quad (18)$$

The R_{22} can be treated as a weighting function. That is, the R_{23}^T represent the weighted Hankel matrix:

$$R_{23}^T = H_{M,N}\, R_{22}^T \quad (19)$$

The system matrices can then be retrieved from the weighted Hankel matrix as discussed in the last subsection of the next section.

III. LATTICE FILTER AND QR FACTORIZATION

In recent years, lattice filters are recognized as one of the fastest and most numerically stable methods for recursive least-squares (RLS) estimation. Many applications have been done by applying these methods. This paper uses a class of unwindowed multichannel lattice algorithms with orthogonal channels recently developed at UCLA [4] for RLS estimation. The adaptive lattice algorithms contain two parts: the residual-error lattice algorithm and the model-parameter algorithm. The first part has to be computed in each time step to generate all the information needed in the algorithm, while the second part only has to be done whenever the model parameters are needed. This is an unwindowed lattice filter, which can eliminate the error caused by the prewindowed assumption (i.e., that all initial data is zero). It is also a multichannel lattice filter, which is essential to the algorithm delevoped in the paper.

In subspace identification algorithms, the major computational cost is QR factorization to the data history matrix. In [5], Cho introduced a fast QR factorization algorithm for subspace system identification via exploiting the displacement of the data matrix. For a $i \times j$ data history matrix, where i is the length of the data and j is the width of the channel history, the algorithm lower the computational load from $O(ij^2)$ to $O(ij)$, which is essential. The lattice filter algorithm used in the research provides a numerically stable way to extract the R matrix recursively from the data matrix with computational load $O(ij)$.

A. Residual-Error Lattice

As shown in Fig. 2, the residual-error lattice filter has two parallel architectures, the forward-propagating and backward-propagating blocks. Details of the lattice stages are given in [4]. The cross arrows connecting forward and backward blocks give rise to the term lattice filter. There is a pair of uncoupled blocks for each n, one for processing the forward-propagating block and one for processing the backward-propagating block. Each such pair of blocks constitutes a stage of the lattice filter. Data is exchanged between the forward-propagating block and the backward propagating block on completion of each stage to initialize the next stage. The error vectors $\hat{e}^i_{n;k}$, $\hat{r}^i_{n;k}$, $\check{e}^i_{n;k}$ and $\check{r}^i_{n;k}$ are defined as forward-propagating forward residual errors, forward-propagating backward residual errors, backward-propagating forward residual errors and backward-propagating backward residual errors, respectively.

The residual-error lattice algorithm does the QR factorization implicitly but obtains all the information needed to construct the R matrix explicitly whenever it is needed. The lattice algorithm in [4] using the following forward and backward error vectors $\hat{f}^i_{n;k}(t)$ and $\check{b}^i_{n;k}(t)$. The correlation matrices for $\hat{f}^i_{n;k}(t)$ and $\check{b}^i_{n;k}(t)$ are

$$\hat{\alpha}^{ij}_{n;k}(t) = \langle \hat{f}^i_{n;k}(t), \hat{f}^j_{n;k}(t) \rangle , \quad (20)$$

$$\check{\alpha}^{ij}_{n;k}(t) = \langle \check{b}^i_{n;k}(t), \check{b}^j_{n;k}(t) \rangle . \quad (21)$$

From here on, the all the subscript index k will be omitted because we only use $k = n$ for all the following equations. In (20) and (21), for $i = j$, we have the norm of $\hat{f}^i_n$ and $\check{b}^i_n$ as

$$|\, \hat{f}^i_n \,| = \sqrt{\langle \hat{f}^i_n, \hat{f}^i_n \rangle} = \sqrt{\hat{\alpha}^{ii}_n} , \quad (22)$$

$$|\, \check{b}^i_n \,| = \sqrt{\langle \check{b}^i_n, \check{b}^i_n \rangle} = \sqrt{\check{\alpha}^{ii}_n} . \quad (23)$$

There are artificial channels embeded in the data to implement the unwindowed lattice filter. It was discussed in [6] that the effect of the artificial channels can be ignored when the t is large.

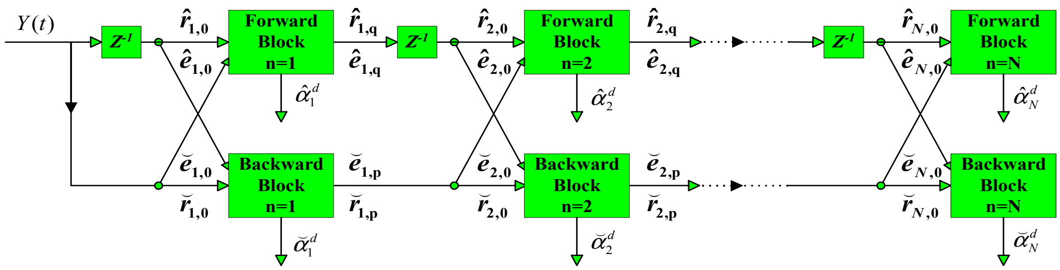

Fig. 2. Residual Lattice Filter.

We define the $\hat{\mathbf{f}}$, $\breve{\mathbf{b}}$, $\hat{\alpha}$ and $\breve{\alpha}$ as

$$\hat{\mathbf{f}} = \left[z^{-n} \hat{f}_1^{1:q} \mid z^{-(n-1)} \hat{f}_2^{1:q} \mid \ldots \mid z^{-1} \hat{f}_n^{1:q} \right] , \quad (24)$$

$$\breve{\mathbf{b}} = \left[\breve{b}_1^{1:q} \mid \breve{b}_2^{1:q} \mid \ldots \mid \breve{b}_n^{1:q} \right] , \quad (25)$$

$$\hat{\alpha} = diag\left(\hat{\alpha}_1^{11:qq} \mid \hat{\alpha}_2^{11:qq} \mid \ldots \mid \hat{\alpha}_n^{11:qq} \right) , \quad (26)$$

$$\breve{\alpha} = diag\left(\breve{\alpha}_1^{11:qq} \mid \breve{\alpha}_2^{11:qq} \mid \ldots \mid \breve{\alpha}_n^{11:qq} \right) , \quad (27)$$

also the data matrices $\hat{\mathbf{W}}$, $\breve{\mathbf{W}}$, and their QR factorization:

$$\hat{\mathbf{W}} = \begin{bmatrix} Y(t-n) & \ldots & Y(t-1) \\ \vdots & \ddots & \vdots \\ Y(1) & \ldots & Y(n) \\ Y(0) & \ldots & Y(n-1) \end{bmatrix} = \hat{Q}\,\hat{R} , \quad (28)$$

$$\breve{\mathbf{W}} = \begin{bmatrix} Y(t-1) & \ldots & Y(t-n) \\ \vdots & \ddots & \vdots \\ Y(n) & \ldots & Y(1) \\ Y(n-1) & \ldots & Y(0) \end{bmatrix} = \breve{Q}\,\breve{R} . \quad (29)$$

From (20) – (29), we conclude the following:

$$\hat{\mathbf{f}} \cdot \hat{\alpha}^{-1/2} = \hat{Q} , \quad \breve{\mathbf{b}} \cdot \breve{\alpha}^{-1/2} = \breve{Q} . \quad (30)$$

B. Model Parameters

The model parameters algorithm and initializations shown in [4] generate the model parameters at any time step t from the data in the residual-error lattice. Fig. 3 illustrates the signal flow of the algorithm. The model parameters need not be generated at every t. For each order n, there are l pairs of uncoupled blocks, which contain one forward-propagating and one backward-propagating block. Data is exchanged between the forward-propagating block and the backward propagating block on completion of each stage to initialize the next stage.

In the model parameters, the components of the forward-propagating model-parameter matrix $\hat{A}_{n;l}$ and backward-propagating model-parameter matrix $\breve{B}_{n;l}$ are

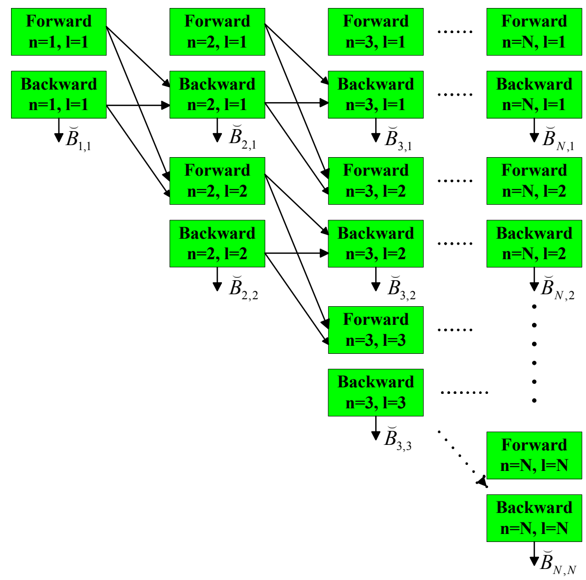

Fig. 3. Model Parameters Algorithm.

defined as

$$\left[\hat{f}_n^1 \ldots \hat{f}_n^p \right] = -\sum_{l=1}^{n} [z^{-l}\Psi]\hat{A}_{n;l} , \quad (31)$$

$$\left[\breve{b}_n^1 \ldots \breve{b}_n^p \right] = -\sum_{l=1}^{n} [z^{-l}\Psi]\breve{B}_{n;l} , \quad (32)$$

from (24), (25), (31) and (32), we have

$$\begin{aligned} \hat{\mathbf{f}} &= -\left[z^{-n}\Psi \mid z^{-(n-1)}\Psi \mid \ldots \mid z^{-1}\Psi \right] \cdot \hat{\mathbf{A}} \\ &= -\hat{W} \cdot \hat{\mathbf{A}} = -\hat{Q} \cdot \hat{R} \cdot \hat{\mathbf{A}} , \end{aligned} \quad (33)$$

$$\begin{aligned} \breve{\mathbf{b}} &= -\left[z^{-1}\Psi \mid z^{-2}\Psi \mid \ldots \mid z^{-n}\Psi \right] \cdot \breve{\mathbf{B}} \\ &= -\breve{W} \cdot \breve{\mathbf{B}} = -\breve{Q} \cdot \breve{R} \cdot \breve{\mathbf{B}} , \end{aligned} \quad (34)$$

where $\hat{\mathbf{A}}$ and $\breve{\mathbf{B}}$ are defined as

$$\hat{\mathbf{A}} = \begin{bmatrix} \hat{A}_{1,0} & \hat{A}_{2,1} & \ldots & \hat{A}_{n,n-1} \\ 0 & \hat{A}_{2,0} & \ldots & \hat{A}_{n,n-2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \hat{A}_{n,0} \end{bmatrix} , \quad (35)$$

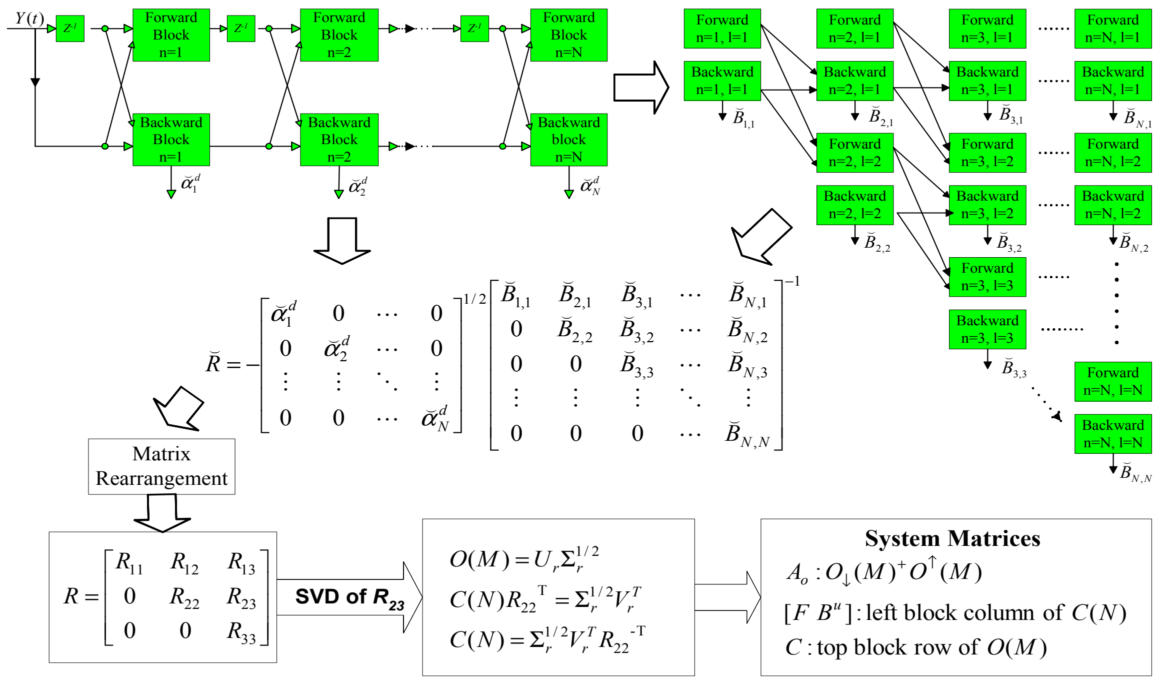

Fig. 4. Subspace System Identification Based on Multichannel RLS Lattice Filter.

$$\breve{\mathbf{B}} \quad = \quad \begin{bmatrix} \breve{B}_{1,1} & \breve{B}_{2,1} & \dots & \breve{B}_{n,1} \\ 0 & \breve{B}_{2,2} & \dots & \breve{B}_{n,2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \breve{B}_{n,n} \end{bmatrix} . \tag{36}$$

From (30), (33) and (34), the following are true:

$$-\hat{\mathbf{A}} \cdot \hat{\alpha}^{-1/2} = \hat{R}^{-1} \ , \quad -\breve{\mathbf{B}} \cdot \breve{\alpha}^{-1/2} = \breve{R}^{-1} \ . \tag{37}$$

C. Subspace Method Implementation

Apply the lattice filter with order $(N+M)$, $Y(t) = [y(t)\ u(t)]$, from (29) and (37), we have

$$\breve{R} = -\breve{\alpha}^{1/2}\breve{\mathbf{B}}^{-\mathbf{1}} \ . \tag{38}$$

Note that the $\breve{R}$ is the R factor of the QR factorization to the lattice filter data matrix $\breve{W}$, which Toeplitz structure is different from W. We developed a method to transfer the $\breve{R}$ to the matrix suitable for the subspace method.

In (29), the lattice order n is $N+M$, and the $Y(t)$ is defined as $[y(t)\ u(t)]$. There exists an unitary T so that $W = \breve{W}T$. Apply the QR factorization to $\breve{R}T$ will give us the R factor of W matrix:

$$W \quad = \quad \breve{W}\,T = \breve{Q}\,\breve{R}\,T = \breve{Q}\,(Q_2\,R_2) = \underbrace{(\breve{Q}\,Q_2)}_{Q}\ \underbrace{(R_2)}_{R} \ .$$

Pick up the R_{23} part from R and apply SVD to its transpose, from (19), we have

$$R_{23}^T \quad = \quad H_M(N)\,R_{22}^T = U\Sigma V^* \ . \tag{39}$$

Use Σ to identify the system order n, reduce the SVD, then

$$\mathcal{O}(M) = U_r\Sigma_r^{1/2}, \qquad \mathcal{C}(N) = \Sigma_r^{1/2}V_r^* R_{22}^{-T} . \tag{40}$$

The system matrices and the Kalman gain matrix are

$$\begin{aligned} A_o \quad &= \quad \mathcal{O}_{\downarrow}(M)^{\dagger}\mathcal{O}^{\uparrow}(M), \\ [F\ \ B^u] \quad &= \quad \text{first left block column of } \mathcal{C}(N), \\ C \quad &= \quad \text{first top block row of } \mathcal{O}(M). \end{aligned}$$

This summary of the method is illustrated in Fig. 4

IV. SIMULATION RESULTS

In [7], sixth-order two-input/two-output models of microgyroscope have been identified. The model that we used in the simulations has two very closely spaced rocking modes near 4392Hz and 4394Hz. The plunging mode is near 2684Hz. Those modes are lightly damped. The two very close, lightly damped modes make identification difficult. The bode plot and the zoom-out near the rocking modes corresponding to the first input/first output is shown in Fig. 5. This example simulates the microgyroscope placed in a noisy environment.

The system is excited by known inputs $u(t)$ and white actuator noise $w(t)$ with $SNR = 25$dB. The measurements $y(t)$ are corrupted by white noise $v(t)$ with $SNR = 25$dB. The subsapce algorithm N4SID in MATLAB is used for comparison. We ran 100 cases and compared identified results. The average errors of the two method are shown in Fig. 6.

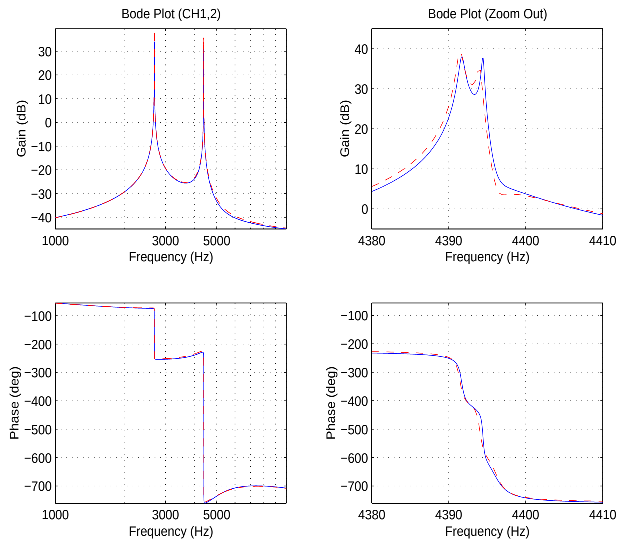

Fig. 5. Bode Plot of a Sixth-Order JPL Microgyroscope. Solid line: true model. Dash line: identified model.

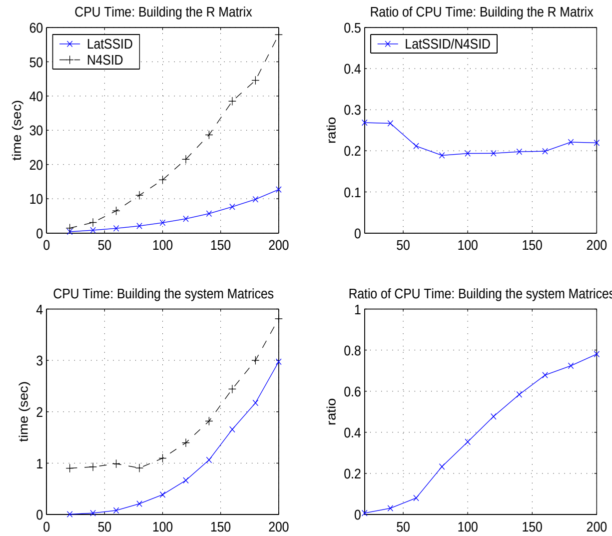

Fig. 7. CPU Time Comparison at different order. Solid line: LatSSID. Dash line: N4SID.

For the CPU time comparison, we measure the time spent on constructing the weighted Hankel matrix and the system matrices. As shown in Fig. 7, the lattice filter based method runs faster than the other without losing the accuracy.

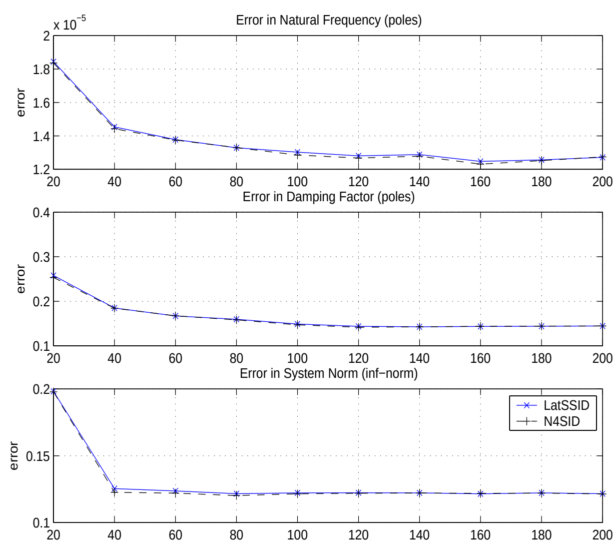

Fig. 6. Error Comparison at different orders. Solid line: LatSSID. Dash line: N4SID.

V. CONCLUSIONS

We have developed a fast, numerically stable subspace system identification algorithm based on a multhchannel adaptive lattice filter. Because an adaptive lattice filter provides the core computational engine of the new subspace algorithm, the main computational burden of the algorithm can be executed in real time, as data is being sampled. This approach produces significantly faster system identification algorithms than existing batch algorithms for subspace identification.

VI. REFERENCES

[1] Peter Van overschee and Bart De Moor, “N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems,” Automatica, vol. 30, pp. 75–93, 1994.

[2] Peter Van overschee and Bart De Moor, Subspace Identification for Linear Systems. Norwell, MA: Kluwer Academic Publishers, 1996.

[3] D. Bauer, M. Deistler, and W. Scherrer, “User choices in subspace algorithm,” in Proceedings of the 37th IEEE Conference on Decision and Control, (Tampa, FL), December 1998.

[4] S.-B. Jiang and J. S. Gibson, “An unwindowed multichannel lattice filter with orthogonal channels,” IEEE Transactions on Signal Processing, vol. 43, pp. 2831–2842, December 1995.

[5] Y. M. Cho, G. Xu, and T. Kailath, “Fast identification of state-space models via exploitation of displacement structure,” IEEE Transactions on Automatic Control, vol. 39, pp. 2004–2017, October 1994.

[6] Neil Y. Chen, Subspace Methods in Adaptive Filtering and System Identification. PhD thesis, University of California, Los Angeles, 2001.

[7] R. T. M’Closkey, J. S. Gibson, and J. Hui, “System identification of a MEMS gyroscope,” ASME Transactions of Dynamic Systems, Measurement, and Control, vol. 123, pp. 201–210, June 2001.