CDC00–INV0206

An Improved Subspace Identification Method for Bilinear Systems

Huixin Chen and **Jan Maciejowski** *

Department of Engineering
University of Cambridge
Cambridge CB2 1PZ U.K.
Tel. +44 1223 332732, Fax. +44 1223 332662
Email: hc240@eng.cam.ac.uk, jmm@eng.cam.ac.uk

Keywords:

System Identification, Bilinear Systems, Subspace Method.

Abstract

Several subspace algorithms for the identification of bilinear systems have been proposed recently. A key practical problem with all of these is the very large size of the data-based matrices which must be constructed in order to 'linearise' the problem and allow parameter estimation essentially by regression. Favoreel *et al* [5] proposed an algorithm which gave unbiased results only if the measured input signal was white. Favoreel and De Moor [6] suggested an alternative algorithm for general input signals, but which gave biased estimates. Chen and Maciejowski proposed algorithms for the deterministic [2] and combined deterministic-stochastic [3] cases which give asymptotically unbiased estimates with general inputs, and for which the rate of reduction of bias can be estimated. The computational complexity of these algorithms was also significantly lower than the earlier ones, both because the matrix dimensions were smaller, and because convergence to correct estimates (with sample size) appears to be much faster. In this paper, we reduce the matrix dimensions further, by making different choices of subspaces for the decomposition of the input-output data. In fact we propose two algorithms: an unbiased one for the case of $l \geq n$, (where l: number of outputs, n: number of states), and an asymptotically unbiased one for the case $l < n$. In each case, the matrix dimensions are smaller than in earlier algorithms. Even with these improvements, the dimensions remain large, so that the algorithms are currently practical only for low values of n.

*Corresponding author

1 Introduction

Several subspace algorithms for the identification of bilinear systems have been proposed recently. A key practical problem with all of these is the very large size of the data-based matrices which must be constructed in order to 'linearise' the problem and allow parameter estimation essentially by regression.

Favoreel *et al* [5] proposed a 'bilinear N4SID' algorithm which gave unbiased results only if the measured input signal was white. Favoreel and De Moor [6] suggested an alternative algorithm for general input signals. Verdult and Verhaegen [11] pointed out that this algorithm gives biased results, and proposed an alternative algorithm, which involved a nonlinear optimization step. Chen and Maciejowski proposed algorithms for the deterministic [2] and combined deterministic-stochastic [3] cases which give asymptotically unbiased estimates with general inputs, and for which the rate of reduction of bias can be estimated. The computational complexity of these algorithms was also significantly lower than the earlier ones, both because the matrix dimensions were smaller, and because convergence to correct estimates (with sample size) appears to be much faster.

In this paper, we reduce the matrix dimensions further for the combined deterministic-stochastic case, by making different choices of subspaces for the decomposition of the input-output data. In fact we propose two algorithms: an unbiased 'three-block' algorithm for the case of $l \geq n$, (where l is the number of outputs and n is the number of states), and an asymptotically unbiased 'four-block' algorithm for the case $l < n$. In each case, the matrix dimensions are smaller than in earlier algorithms. Even with these improvements, the dimensions remain large, so that the algorithms are currently practical only for low values of n. We include three examples, which illustrate the cases ($l = 1, n = 2$), ($l = 2, n = 2$), ($l = 1, n = 3$).

The outline of the paper is as follows. Some notations for block data matrices are introduced in section 2. Some important new notations (compared with [3]) are introduced. Some theoretical results are given in section 3. The three block algorithm for the case $l \geq n$ is introduced in section 4. The four block algorithm for the case $l < n$ is presented in section 5. Section 6 contains the examples. All proofs are omitted here, but can be found in [1].

2 Notation

The use of much specialised notation seems to be unavoidable in the current context. Mostly we follow the notation used in [7, 2], but we introduce all the notation here for completeness.

We use $\otimes$ to denote the Kronecker product and $\odot$ the Khatri-Rao product of two matrices with $F \in \mathbf{R}^{t\times p}$ and $G \in \mathbf{R}^{u\times p}$ defined in [8, 10]:

$$F \odot G \triangleq [f_1 \otimes g_1, f_2 \otimes g_2, \ldots , f_p \otimes g_p]$$

$+, \oplus$ and $\cap$ denote the sum, the direct sum and the intersection of two vector spaces, $\cdot^{\perp}$ denotes the orthogonal complement of a subspace with respect to the predefined ambient space, the Moore-Penrose inverse is written as $\cdot^{\dagger}$, and the Hermitian as $\cdot^{*}$.

In this paper we consider combined deterministic-stochastic time-invariant bilinear system of the form:

$$\begin{aligned} x_{t+1} &= Ax_t + Nu_t \otimes x_t + Bu_t + w_t \\ y_t &= Cx_t + Du_t + v_t \end{aligned} \quad (1)$$

where $x_t \in \mathbf{R}^n, y_t \in \mathbf{R}^l, u_t \in \mathbf{R}^m$, and $N = [N_1\ N_2\ ...N_m] \in \mathbf{R}^{n\times nm}$, $N_i \in \mathbf{R}^{n\times n}$ $(i = 1, ..., m)$.

The input u_t is assumed to be independent of the measurement noise v_t and the process noise w_t. The covariance matrix of w_t and v_t is:

$$\mathbf{E}\left[\begin{pmatrix} w_p \\ v_p \end{pmatrix}\begin{pmatrix} w_q \\ v_q \end{pmatrix}^T\right] = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix}\delta_{pq} \geq 0$$

We assume that the sample size is $\tilde{N}$, namely that input-output data $\{u(t), y(t) : t = 0, 1, \ldots , \tilde{N}\}$ are available. For arbitrary t we define

$$X_t \triangleq [x_t\ x_{t+1}\ ...\ x_{t+j-1}] \in \mathbf{R}^{n\times j}$$

but for the special cases $t = 0$ and $t = k$ we define, with some abuse of notation,

$$\begin{aligned} X_p &\triangleq [x_0\ x_1\ ...\ x_{j-1}] \in \mathbf{R}^{n\times j} \\ X_c &\triangleq [x_k\ x_{k+1}\ ...\ x_{k+j-1}] \in \mathbf{R}^{n\times j} \\ X_f &\triangleq [x_{2k}\ x_{2k+1}\ ...\ x_{2k+j-1}] \in \mathbf{R}^{n\times j} \\ X_r &\triangleq [x_{3k}\ x_{3k+1}\ ...\ x_{3k+j-1}] \in \mathbf{R}^{n\times j} \end{aligned}$$

where k is the *row block size*. The suffices p, c, f and r are supposed to be mnemonic, representing 'past', 'current', 'future' and 'remote future' respectively. We define U_t, U_p, U_f, U_r, Y_t, Y_p, Y_f, Y_r, W_t, U_p, W_f, W_r, V_t, V_p, V_f, V_r, similarly. These matrices will later be used to construct larger matrices with a 'generalised block-Hankel' structure. In order to use all the available data in these, the number of columns j is such that $\tilde{N} = 3k + j - 1$ in the case of $l \geq n$ and $\tilde{N} = 4k + j - 1$ in the case of $l < n$. Let $d_i = \Sigma_{p=1}^{i}(m + 1)^{p-1}l$, $e_i = \Sigma_{p=1}^{i}(m+1)^{p-1}m$, $f_k = e_k + \frac{m}{2}(m+1)^k + l[(m+1)^k - 1]$ and $g_k = e_k + e_k^2$.

For arbitrary q and $i \geq q + 2$, we define

$$\begin{aligned}
X_{q|q} &\triangleq \begin{pmatrix} X_q \\ U_q \odot X_q \end{pmatrix} \in \mathbf{R}^{(m+1)n\times j} \\
X_{i-1|q} &\triangleq \begin{pmatrix} X_{i-2|q} \\ U_{i-1} \odot X_{i-2|q} \end{pmatrix} \in \mathbf{R}^{(m+1)^{i-q}n\times j} \\
Y_{q|q} &\triangleq Y_q \\
Y_{i-1|q} &\triangleq \begin{pmatrix} Y_{i-1} \\ Y_{i-2|q} \\ U_{i-1} \odot Y_{i-2|q} \end{pmatrix} \in \mathbf{R}^{d_{i-q}\times j} \\
U^+_{q|q} &\triangleq U_q \\
U^+_{i-1|q} &\triangleq \begin{pmatrix} U^+_{i-2} \\ U_{i-1} \\ U_{i-1} \odot U^+_{i-2|q} \end{pmatrix} \in \mathbf{R}^{e_{i-q}\times j} \\
U^{++}_{q|q} &\triangleq \begin{pmatrix} U_{q,1} \odot U_q \\ U_{q,2} \odot U_q(2:m,:) \\ U_{q,3} \odot U_q(3:m,:) \\ \vdots \\ U_{q,m} \odot U_{q,m} \end{pmatrix} \in \mathbf{R}^{\frac{m(m+1)}{2}\times j} \\
U^{++}_{i-1|q} &\triangleq \begin{pmatrix} U^{++}_{i-2|q} \\ U_{i-1} \odot U^{++}_{i-2|q} \end{pmatrix} \in \mathbf{R}^{\frac{m}{2}(m+1)^{i-q}\times j} \\
U^u_{i+k-1|q+k} &\triangleq \begin{pmatrix} U_{i+k-1|q+k} \\ U^+_{i+k-1|q+k} \odot U_{i-1|q} \end{pmatrix} \\
U^y_{i-1|q} &\triangleq U^+_{i-1|q} \odot Y_q \\
\tilde{U}^{u,y}_{i+k-1|k+q} &\triangleq \begin{pmatrix} U_{i+k-1|k+q} \\ U^{++}_{i+k-1|k+q} \\ U^y_{i+k-1|k+q} \end{pmatrix}
\end{aligned}$$

$$\begin{aligned}
X^c &\triangleq X_{2k-1|k},\ X^f \triangleq X_{3k-1|2k},\ X^r \triangleq X_{4k-1|3k} \\
U^p &\triangleq U_{k-1|0},\ U^c \triangleq U_{2k-1|k},\ U^f \triangleq U_{3k-1|2k} \\
U^{p,y} &\triangleq U^{+p} \odot Y_p,\ U^{c,y} \triangleq U^{+c} \odot Y_c \\
U^{f,y} &\triangleq U^{+f} \odot Y_f, \\
\tilde{U}^{p,u,y} &\triangleq \begin{pmatrix} U^p \\ U^{++p} \\ U^{p,y} \end{pmatrix},\ \tilde{U}^{c,u,y} \triangleq \begin{pmatrix} U^c \\ U^{++c} \\ U^{c,y} \end{pmatrix} \\
U^{c,u} &\triangleq \begin{pmatrix} U^c \\ U^{+c} \odot U^p \end{pmatrix},\ U^{f,u} \triangleq \begin{pmatrix} U^f \\ U^{+f} \odot U^c \end{pmatrix}
\end{aligned}$$

U^r, Y^p, Y^c, Y^f, Y^r, W^c, W^f, W^r, V^c, V^f, V^r, U^{+c}, U^{+f}, U^{+r}, U^{++c}, U^{++f}, $U_{i-1|q}$, $W_{i-1|q}$, $V_{i-1|q}$, $\tilde{U}^{f,u,y}$ and $U^{r,u}$ can be defined similarly.

Remark 1. The meaning of $U^{+}_{i-1|q}$ is different from that in [4]. $U^{++}_{i-1|q}, U^{u}_{i+k-1|q+k}$, $U^{y}_{i-1|q}$, $\tilde{U}^{u,y}_{i+k-1|k+q}$ and $U^{c,u}$ etc are newly introduced in this paper.

Finally, we denote by $\mathcal{U}_p$ the space spanned by all the rows of the matrix U_p. That is,

$$\mathcal{U}_p := \text{span}\{\alpha^* U_p, \quad \alpha \in \mathbf{R}^{km}\}$$

$\mathcal{U}_c$, $\mathcal{U}_f$, $\mathcal{U}_r$, $\mathcal{Y}_p$, $\mathcal{Y}_c$, $\mathcal{Y}_f$, $\mathcal{Y}_r$, $\mathcal{U}^p$, $\mathcal{Y}^p$, $\mathcal{U}^f$, $\mathcal{Y}^f$, $\tilde{\mathcal{U}}^{p,u,y}$, $\tilde{\mathcal{U}}^{f,u,y}$ and $\mathcal{U}^{r,u}$ *etc* are defined similarly.

3 Analysis

Lemma 1 *The system (1) can be rewritten in the following matrix equation form:*

$$\begin{aligned} X_{t+1} &= AX_t + NU_t \odot X_t + BU_t + W_t \\ Y_t &= CX_t + DU_t + V_t \end{aligned} \qquad (2)$$

Lemma 2 *For $j \geq 0$, and the block size k, we have*

$$X_{k-1+j|j} = \begin{pmatrix} X_j \\ U^{+}_{k-1+j|j} \odot X_j \end{pmatrix}$$

Lemma 3 *For F, G, H, J of compatible dimensions, $F \in \mathbf{R}^{k\times l}$, $G \in \mathbf{R}^{l\times m}$, $H \in \mathbf{R}^{p\times l}$, $J \in \mathbf{R}^{l\times m}$:*

$$\begin{aligned} (FG \otimes HJ) &= (F \otimes H)(G \otimes J) \\ (FG \odot HJ) &= (F \otimes H)(G \odot J) \end{aligned}$$

Lemma 4 (Input-Output Equation) *For the combined deterministic-stochastic system (1) and $j \geq 0$, we have the following Input-Output Equation*

$$\begin{aligned} X_{k+j+1} &= \triangle^X_k X_{k+j|j-1} + \triangle^U_k U_{k+j|j-1} \\ &\quad + \triangle^W_k W_{k-1+j|j} \\ Y_{k+j|j} &= \mathcal{L}^X_k X_{k+j|j-1} + \mathcal{L}^U_k U_{k+j|j-1} \\ &\quad + \mathcal{L}^W_k W_{k+j|j-1} + \mathcal{L}^V_k V_{k+j|j-1} \end{aligned}$$

where

$$\begin{aligned} \triangle^X_n &\triangleq [A\triangle^X_{n-1}, N_1\triangle^X_{n-1}, \dots, N_m\triangle^X_{n-1}] \\ \triangle^X_1 &\triangleq [A, N_1, \dots, N_m] \\ \triangle^U_n &\triangleq [B, A\triangle^U_{n-1}, N_1\triangle^U_{n-1}, \dots, N_m\triangle^U_{n-1}] \\ \triangle^U_1 &\triangleq B \\ \triangle^W_n &\triangleq [I_{n\times n}, A\triangle^W_{n-1}, N_1\triangle^W_{n-1}, \dots, N_m\triangle^W_{n-1}] \\ \triangle^W_1 &\triangleq I_{n\times n} \end{aligned}$$

$$\mathcal{L}^X_k \triangleq \begin{bmatrix} C\triangle^X_{k-1} & 0 & \dots & 0 \\ \mathcal{L}^X_{k-1} & 0 & \dots & 0 \\ 0 & \mathcal{L}^X_{k-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \vdots & \mathcal{L}^X_{k-1} \end{bmatrix}$$

$$\mathcal{L}^U_k \triangleq \begin{bmatrix} D & C\triangle^U_{k-1} & 0 & \dots & 0 \\ 0 & \mathcal{L}^U_{k-1} & 0 & \dots & 0 \\ 0 & 0 & \mathcal{L}^U_{k-1} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \vdots & \mathcal{L}^U_{k-1} \end{bmatrix}$$

$$\mathcal{L}^W_k \triangleq \begin{bmatrix} 0 & C\triangle^W_{k-1} & 0 & \dots & 0 \\ 0 & \mathcal{L}^W_{k-1} & 0 & \dots & 0 \\ 0 & 0 & \mathcal{L}^W_{k-1} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \vdots & \mathcal{L}^W_{k-1} \end{bmatrix}$$

$$\mathcal{L}^V_k \triangleq \begin{bmatrix} I_{l\times l} & 0 & \dots & 0 \\ 0 & \mathcal{L}^V_{k-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathcal{L}^V_{k-1} \end{bmatrix}$$

with

$\mathcal{L}^X_1 \triangleq [C, 0_{l\times m}]$, $\mathcal{L}^U_1 \triangleq D$, $\mathcal{L}^W_1 \triangleq 0_{l\times n}$, $\mathcal{L}^V_1 \triangleq I_{l\times l}$

Lemma 5 *For system (1), if*

$$\lambda = \max_{j=0,\dots,\tilde{N}} |\overline{\sigma}(A + \sum_{i=1}^{n} u_{j,i} N_i)| < 1, \qquad (3)$$

then

$$X_t = \triangle^U_n U_{t-1|t-k} + \triangle^W_n W_{t-1|t-k} + \varepsilon(\lambda^k)$$

where $\overline{\sigma}$ is the maximum sigular value of a matrix and $\varepsilon(\lambda^k)$ is used to denote a matrix M, such that $\|M\|_1 = o(\lambda^k)$.

4 Three Block Algorithm

In this section, a three-block algorithm is set up for the case of $l \geq n$. Here only data blocks 'p', 'c' and 'f' are used (hence '3-block') and $\tilde{N} = 3k + j - 1$.

Theorem 1 (Three Block Form Equation) *The system (1) can be written in the following 'three block' form:*

$$\begin{aligned} Y^p &= \mathcal{O}_k X_p + \mathcal{T}^u_k \tilde{U}^{p,u,y} \\ &\quad + \mathcal{T}^v_k U^{+p} \odot V_p + \mathcal{L}^W_k W^p + \mathcal{L}^V_k V^p \qquad (4) \\ Y^c &= \mathcal{O}_k X_c + \mathcal{T}^u_k \tilde{U}^{c,u,y} \\ &\quad + \mathcal{T}^v_k U^{+c} \odot V_c + \mathcal{L}^W_k W^c + \mathcal{L}^V_k V^c \qquad (5) \\ Y^f &= \mathcal{O}_k X_f + \mathcal{T}^u_k \tilde{U}^{f,u,y} \\ &\quad + \mathcal{T}^v_k U^{+f} \odot V_f + \mathcal{L}^W_k W^f + \mathcal{L}^V_k V^f \qquad (6) \\ X_c &= \mathcal{F}_k X_p + \mathcal{G}^u_k \tilde{U}^{p,u,y} + \mathcal{G}^v_k U^{+p} \odot V_p + \triangle^W_k W^p \\ X_f &= \mathcal{F}_k X_c + \mathcal{G}^u_k \tilde{U}^{c,u,y} + \mathcal{G}^v_k U^{+c} \odot V_c + \triangle^W_k W^f \end{aligned}$$

where $\mathcal{O}_k, \mathcal{T}_k^u, \mathcal{T}_k^v, \mathcal{F}_k, \mathcal{G}_k^u$ and $\mathcal{G}_k^v$ are system-dependent constant matrices.

Theorem 2 *If the linear part of the system (1) is observable and*

$$\begin{pmatrix} Y^p \\ \tilde{U}^{p,u,y} \\ \tilde{U}^{c,u,y} \\ \tilde{U}^{f,u,y} \end{pmatrix} \tag{7}$$

is a full row rank matrix, then suppose condition (7) holds. Denote $\tilde{\mathcal{S}} = \mathcal{Y}^p + \tilde{\mathcal{U}}^{p,u,y} + \tilde{\mathcal{U}}^{c,u,y} + \tilde{\mathcal{U}}^{f,u,y}$ and $\tilde{\mathcal{R}} = \Pi_{\tilde{\mathcal{S}}}\mathcal{Y}^c + \tilde{\mathcal{U}}^{c,u,y}$, then,

$$\Pi_{\tilde{\mathcal{R}}^\perp}\Pi_{\tilde{\mathcal{S}}}\mathcal{Y}^f = \mathcal{T}_k^u \Pi_{\tilde{\mathcal{R}}^\perp}\tilde{U}^{f,u,y} \tag{8}$$

The orthogonal projection operator Π is defined as in [4]

Algorithm:
1. Decompose Y^f into $\mathcal{O}_k X_f$ and $\mathcal{T}_k^u \tilde{U}^{f,u,y}$ using orthogonal projection: from (8) of Theorem 2, estimate $\mathcal{T}_k^u$ as

$$\hat{\mathcal{T}}_k^u = (\Pi_{\tilde{\mathcal{R}}^\perp}\Pi_{\tilde{\mathcal{S}}}\mathcal{Y}^f)(\Pi_{\tilde{\mathcal{R}}^\perp}\tilde{U}^{f,u,y})^\dagger \tag{9}$$

2. Obtain the SVD decomposition and partition as

$$\begin{aligned}\left[\Pi_{\tilde{\mathcal{S}}}Y_{3k-1|2k}\ \Pi_{\tilde{\mathcal{S}}}Y_{3k|2k+1}\right] - \hat{\mathcal{T}}_k^u\left[\tilde{U}^{u,y}_{3k-1|2k}\ \tilde{U}^{u,y}_{3k|2k+1}\right] \\ =: \Gamma\Sigma\Omega^* = \begin{bmatrix} \Gamma_1 & \Gamma_2 \end{bmatrix}\begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}\begin{bmatrix} \Omega_1^* \\ \Omega_2^* \end{bmatrix}\end{aligned}$$

Since we expect

$$\Gamma\Sigma\Omega^* = \Gamma_1\Sigma_1\Omega_1^* = \mathcal{O}_k\left[X_{3k-1} \quad X_{3k}\right]$$

from (4-6), ($\mathrm{rank}(\Sigma_1) = n$ and $\mathrm{rank}(\Sigma_2) = 0$), form the estimates $\hat{\mathcal{O}}_k = \Gamma_1\Sigma_1^{1/2}$ and $[\hat{X}_{3k-1} \quad \hat{X}_{3k}] = \Sigma_1^{1/2}\Omega_1^*$, retaining only $\hat{n}$ significant singular values in Σ_1. ($\hat{\mathcal{O}}_k$ is not needed later.)

3. Estimate the parameters A, B, C, D, N on the basis of equation (2), by solving

$$\begin{bmatrix} \hat{X}_{3k} \\ Y_{3k-1} \end{bmatrix} = \begin{bmatrix} A & N & B \\ C & 0 & D \end{bmatrix}\begin{bmatrix} \hat{X}_{3k-1} \\ U_{3k-1} \odot \hat{X}_{3k-1} \\ U_{3k-1} \end{bmatrix} \tag{10}$$

in a least-squares sense.

4. Estimate the covariance matrix by calculating

$$\begin{aligned}\begin{bmatrix} \epsilon_w \\ \epsilon_v \end{bmatrix} &= \begin{bmatrix} \hat{X}_{3k} \\ Y_{3k-1} \end{bmatrix} \\ &\quad - \begin{bmatrix} \hat{A} & \hat{N} & \hat{B} \\ \hat{C} & 0 & \hat{D} \end{bmatrix}\begin{bmatrix} \hat{X}_{3k-1} \\ U_{3k-1} \odot \hat{X}_{3k-1} \\ U_{3k-1} \end{bmatrix}\end{aligned}$$

then estimating Q, S, R from the sample covariance of $[\epsilon_w^T, \epsilon_v^T]^T$.

5 Four Block Algorithm

In this section, a 'four-block' algorithm is proposed for the case $l < n$. Now all four data blocks: 'p','c', 'f' and 'r' are needed (U^p is involved in the definition of $U^{c,u}$ etc). Here $\tilde{N} = 4k + j - 1$.

Theorem 3 (Four Block Form Equation) *The system (1) can be written in the following form:*

$$\begin{aligned}
Y^c &= \mathcal{O}_{k,1}X_c + \mathcal{T}_{k,1}^u U^{c,u} + \mathcal{T}_{k,1}^v U^{+c} \odot V_c \\
&\quad \mathcal{T}_{k,1}^w U^{+c} \odot W^p + \mathcal{L}_k^W W^c + \mathcal{L}_k^V V^c + \varepsilon(\lambda^k) \\
Y^f &= \mathcal{O}_{k,1}X_f + \mathcal{T}_{k,1}^u U^{f,u} + \mathcal{T}_{k,1}^v U^{+f} \odot V_f \\
&\quad \mathcal{T}_{k,1}^w U^{+f} \odot W^c + \mathcal{L}_k^W W^f + \mathcal{L}_k^V V^f + \varepsilon(\lambda^k) \\
Y^r &= \mathcal{O}_{k,1}X_r + \mathcal{T}_{k,1}^u U^{r,u} + \mathcal{T}_{k,1}^v U^{+r} \odot V_r \\
&\quad \mathcal{T}_{k,1}^w U^{+r} \odot W^f + \mathcal{L}_k^W W^r + \mathcal{L}_k^V V^r + \varepsilon(\lambda^k) \\
X_f &= \mathcal{F}_{k,1}X_c + \mathcal{G}_{k,1}^u U^{c,u} + \mathcal{G}_{k,1}^v U^{+c} \odot V_c \\
&\quad \mathcal{G}_{k,1}^w U^{+c} \odot W^p + \triangle_{k,1}^W W^c + \varepsilon(\lambda^k) \\
X_r &= \mathcal{F}_{k,1}X_f + \mathcal{G}_{k,1}^u U^{f,u} + \mathcal{G}_{k,1}^v U^{+f} \odot V_f \\
&\quad \mathcal{G}_{k,1}^w U^{+f} \odot W^c + \triangle_{k,1}^W W^f + \varepsilon(\lambda^k)
\end{aligned}$$

where $\mathcal{O}_{k,1}, \mathcal{T}_{k,1}^u,\ \mathcal{T}_{k,1}^v, \mathcal{T}_{k,1}^w, \mathcal{F}_{k,1}, \mathcal{G}_{k,1}^u, \mathcal{G}_{k,1}^w$ and $\mathcal{G}_{k,1}^v$ are system-dependent constant matrices.

Remark 3 This differs from Theorem 1 of [3] by the use of $U^{c,u}$ instead of $U^{c,u,y}$, $U^{f,u}$ instead of $U^{f,u,y}$, and $U^{r,u}$ instead of $U^{r,u,y}$.

Theorem 4 *Suppose that the linear part of the system (2) is observable and*

$$\begin{pmatrix} Y^c \\ U^{c,u} \\ U^{f,u} \\ U^{r,u} \end{pmatrix} \tag{11}$$

is a full row rank matrix. Denote $\mathcal{S}_1 = \mathcal{Y}^c + \mathcal{U}^{c,u} + \mathcal{U}^{f,u} + \mathcal{U}^{r,u}$ and $\mathcal{R}_1 = \Pi_{\mathcal{S}_1}\mathcal{Y}^f + \mathcal{U}^{f,u}$. Then

$$\Pi_{\mathcal{R}_1^\perp}\Pi_{\mathcal{S}_1}\mathcal{Y}^r = \mathcal{T}_{k,1}^u\Pi_{\mathcal{R}_1^\perp}U^{r,u} + \varepsilon(\lambda^k) \tag{12}$$

Algorithm:
1. Decompose Y^r into $\mathcal{O}_{k,1}X_r$ and $\mathcal{T}_{k,1}^u U^{r,u}$ using orthogonal projection: from (12) of Theorem 4, estimate $\mathcal{T}_{k,1}^u$ as

$$\hat{\mathcal{T}}_{k,1}^u = (\Pi_{\mathcal{R}_1^\perp}\Pi_{\mathcal{S}_1}\mathcal{Y}^r)(\Pi_{\mathcal{R}_1^\perp}U^{r,u})^\dagger \tag{13}$$

2. Obtain the SVD decomposition and partition accordingly by selecting a model order as shown in the three-block algorithm.

$$\begin{aligned}\left[\Pi_{\mathcal{S}_1}Y_{4k-1|3k}\ \Pi_{\mathcal{S}_1}Y_{4k|3k+1}\right] - \hat{\mathcal{T}}_{k,1}^u\left[U^u_{4k-1|3k}\ U^u_{4k|3k+1}\right] \\ =: \Gamma\Sigma\Omega^* = \begin{bmatrix} \Gamma_1 & \Gamma_2 \end{bmatrix}\begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}\begin{bmatrix} \Omega_1^* \\ \Omega_2^* \end{bmatrix}\end{aligned}$$

3. Estimate the parameters A, B, C, D, N, Q, S, R as in steps 3 and 4 of the three-block algorithm in the previous section.

Remark 4 The 'full row rank' requirement in Theorems 2 and 4 can only be met if $k \geq n$.

Remark 5 We envisage that one would usually start by using the 'four-block' algorithm. If the singular values indicated that $l \geq n$ might be a possibility, then one could try the 'three-block' algorithm.

6 Examples

The first two examples are taken from [4, 7], respectively.

Example 1. The true system is

$$A = \begin{pmatrix} 0 & 0.5 \\ -0.5 & 0 \end{pmatrix}, B = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, C = \begin{pmatrix} 1 & 1 \end{pmatrix},$$
$$D = 2, \quad N_1 = \begin{pmatrix} 0.4 \\ 0 \end{pmatrix}, \quad N_2 = \begin{pmatrix} 0 \\ 0.3 \end{pmatrix}$$

and the noise covariance matrices are

$$Q = \begin{pmatrix} 0.16 & 0 \\ 0 & 0.04 \end{pmatrix}, R = 0.09, S = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad (14)$$

Since $l < n$, the four-block algorithm is applied. In [4], the input was white noise and $k = 3, j = 8191$ were used. In the cases of I and II, the system input is a uniform distribution with mean value 0, variance 1, and $\lambda = 0.7809$. Case I is for the system noise (14). For case II we increased the signal to noise ratio:

$$Q = \begin{pmatrix} 0.0016 & 0 \\ 0 & 0.0004 \end{pmatrix}, R = 0.0009, S = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad (15)$$

For cases III and IV, we used a coloured noise input signal u with mean 0, standard deviation 1.1664, $\lambda = 0.7906$ and $r_q = Eu_k u_{k+q} = 0.5^q$, $q = 0, 1, 2, \ldots$. Case III had noise covariances (14) and case IV had noise covariances (15). For all the cases I–IV we used $j = 595$ with our new algorithm. The results are shown in Table 1.

	eig(A)	eig(N)
True	$\pm 0.5i$	0.4, 0.3
N4SID	$-0.0027 \pm 0.4975i$	0.4011, 0.3055
Case I	$-0.0076 \pm 0.4960i$	0.3838, 0.2829
Case II	$0.0000 \pm 0.5000i$	0.4005, 0.3030
Case III	$0.0044 \pm 0.4847i$	0.4048, 0.2688
Case IV	$0.0089 \pm 0.4945i$	0.3906, 0.3149

Table 1: Example 1: Results with different inputs, noise ratios and algorithms

Example 2. The true system is defined by:

$$A = \begin{pmatrix} 0.5 & 0 \\ 0 & 0.3 \end{pmatrix}, B = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$$

$D = I$, $N_1 = \text{diag}[0.6, 0.4]$, $N_2 = \text{diag}[0.2, 0.5]$, $Q = R = 0.01I$, $S = 0$. Now $l = n$, so the three-block algorithm is applied. The input was a two-dimensional uniform distribution notation for cases V and VI and coloured noise input u with $Eu_i u_{i+q} = 0.9^i I_2$ for cases VII and VIII, with $\tilde{N} = 1000, k = 2$ in all cases. In cases V and VII, ordinary least-squares was used in solving (10), while in cases VI and VIII a constrained LS method was used, to take account of the known structure of the solution (the zero block). Table 2 summarises the results, including a comparison with the results obtained in [7], where $\tilde{N} = 4095$ and $k = 2$ were used. From Table 2, we know that there is no significant difference solving equation (10) by using ordinary least-squares and constrained LS method.

	eig(A)	eig(N_1)	eig(N_2)
True	0.5, 0.3	0.6, 0.4	0.2, 0.5
N4SID	0.5001	0.4020	0.1914
($\tilde{N} = 4095$)	0.2979	0.5994	0.5016
case V	0.4936	0.6020	0.5030
($\tilde{N} = 1000$)	0.3022	0.4124	0.1965
case VI	0.5020	0.5990	0.4903
($\tilde{N} = 1000$)	0.3006	0.4028	0.2045
case VII	0.5002	0.5997	0.5005
($\tilde{N} = 1000$)	0.3009	0.4003	0.1996
case VIII	0.5000	0.6000	0.5009
($\tilde{N} = 1000$)	0.3011	0.4004	0.2000

Table 2: Example 2: Comparisons with different algorithms, LS and constrained LS

Example 3. The true system is:

$$A = \begin{pmatrix} 0 & 0.5 & 0 \\ -0.5 & 0 & 0 \\ 0 & 0 & 0.4 \end{pmatrix}, B = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, C = B^T,$$
$$D = 3, \quad N = \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & -0.2 & 0 \\ 0 & 0 & 0.2 \end{pmatrix}$$

and the noise is the same as (15). The four-block algorithm is used, since $l < n$. Here, a coloured input with mean 0, variance 0.01, $Eu_k u_{k+q} = 0.5^q$ and $\lambda = 0.8689$ was used. Results with different block and sample sizes are given in Table 3.

$(k, \tilde{N})$	eig(A)	eig(N)
True	$\pm 0.5i$, 0.4	0.5, ± 0.2
(3,800)	$0.00 \pm 0.49i$, 0.29	0.47, 0.17, -0.08
(4,800)	$-0.01 \pm 0.49i$,0.41	0.47, 0.22, -0.12
(3,1200)	$0.00 \pm 0.50i$, 0.40	0.49, 0.19, -0.09
(4,1200)	$0.00 \pm 0.50i$,0.40	0.48, 0.22, -0.20
(3,1500)	$0.00 \pm 0.50i$, 0.40	0.49, 0.18, -0.12
(4,1500)	$0.00 \pm 0.50i$, 0.40	0.51, 0.19, -0.19

Table 3: Example 3: Effect of sample size and block size

Remark 6 Our new algorithm has considerably lower computational complexity than the algorithms proposed in [6] and [3]. The major computational load is involved in finding the right-inverse in (9) and(13). The row dimensions of the relevant matrices which appear in the algorithms presented here, in [6], and in [3], are shown in Table 4 for the three examples. The algorithm presented in this paper is denoted as 'Algorithm I', where the row dimension is $g_k = e_k + e_k^2$ $(l < n)$ for Examples 1 and 3, and $f_k = e_k + (m/2)(m+1)^k + l[(m+1)^k - 1]$ $(l = n)$ for Example 2. The algorithm from [3] is denoted as 'Algorithm II'; in this case the row dimension is $e_k + (m/2)(m+1)^k + l[(m+1)^k - 1] + e_k^2$. For the bilinear N4SID algorithm of [6] the row dimension is $(d_k + 2e_k + e_k d_k + e_k^2)$. In Table 4 it is assumed that $k = 2$ for examples 1 and 2, and $k = 3$ for example 3.

Dimensions	Algorithm I	Algorithm II	N4SID
Example 1	12	17	27
Example 2	33	97	152
Example 3	56	67	119

Table 4: Comparison of dimensions of matrices for Examples 1–3 and various algorithms

7 Conclusion

A new subspace identification algorithm which consists of two sub-algorithms is proposed for the identification of bilinear systems.

The main advantage of this algorithm over earlier ones is that the computational complexity is lower, since the matrices involved are of smaller dimensions.

8 Acknowledgement

The work reported in this paper was supported by the UK Engineering and Physical Science Research Council under Grant GR/MO8332, and by the European Research Network on System Identification (ERNSI) under TMR contract ERB FMRX CT98 0206.

References

[1] Chen H. and Maciejowski J.M., *New Subspace Identification Method for Bilinear Systems* CUED/F-INFENG/TR.357, 1999, (Submitted to *Automatics*).

[2] Chen H. and Maciejowski J.M., Subspace Identification of Deterministic Bilinear Systems, *Accepted for ACC 2000, Chicago, Illinois, USA*, June, 2000

[3] Chen H. and Maciejowski J.M., Subspace Identification of Combined Deterministic-Stochastic Bilinear Systems, *Accepted for IFAC Symp. on System Identification, SYSID 2000, Santa Barbara, California, USA*, June, 2000

[4] Favoreel W, De Moor B. and Van Overschee P., *Subspace identification of bilinear systems subject to white inputs*, ESAT-SISTA/TR 1996-53I, Katholieke Universiteit Leuven, 1996.

[5] Favoreel W, De Moor B. and Van Overschee P., *Subspace identification of Balanced deterministic bilinear systems subject to white inputs*, ECC, 1997

[6] Favoreel W and De Moor B., *Subspace identification of bilinear systems*, Proc. MTNS, 1998. pp. 787-790.

[7] Favoreel W, De Moor B. and Van Overschee P., *Subspace identification of bilinear systems subject to white inputs*, IEEE Trans on AC, Vol 44., No. 6, June 1999. pp. 1157–1165.

[8] Khatri C.G. and Rao C.R., *Solutions to some functional equations and their applications to characterization of probability distributions,* Sankhya : The Indian J. Stat., series A, 30, pp. 167-180, 1968.

[9] Van Overschee P. and De Moor B., *N4SID: subspace algorithms of combined deterministic and stochastic systems* , Automatica, vol.30, No. 1, pp. 75-93, 1994

[10] Suda N., Kodama S. and Ikeda M., *Matrix Theory in Automatical Control,* Japanese Automatical Control Association, 1973.

[11] Verdult V., and Verhaegen M., *Subspace-based Identification of MIMO Bilinear Systems*, ECC 1999, 31.August-3.September, 1999, Karlsruhe, Germany