Eigensystem Realization

CEE 629. System Identification

Duke University, Fall 2017

1 Markov Parameters

The response sequence of a system driven by a particular input sequence and assuming initial conditions of zero, can be found from $y(i) = Cx(i) + Du(i)$

$$\begin{aligned}
y(1) &= Cx(1) + Du(1) = CBu(0) + Du(1) \\
y(2) &= CAx(1) + CBu(1) + Du(2) = CABu(0) + CBu(1) + Du(2) \\
y(3) &= CAx(2) + CBu(2) + Du(3) = CA^2Bu(0) + CABu(1) + CBu(2) + Du(3) \\
\vdots & \quad \vdots \\
y(i) &= CA^{(i-1)}Bu(0) + \cdots + CA^2Bu(i-3) + CABu(i-2) + CBu(i-1) + Du(i)
\end{aligned}$$

$$y(i) = \begin{bmatrix} D & CB & CAB & CA^2B & \cdots & CA^{(i-2)}B & CA^{(i-1)}B \end{bmatrix} \begin{bmatrix} u(i) \\ u(i-1) \\ u(i-2) \\ u(i-3) \\ \vdots \\ u(1) \\ u(0) \end{bmatrix} \tag{1}$$

This is a matrix input-output relationship in which the matrix on the left of the product is called a matrix of *Markov parameters*, $Y(0) = D$, and $Y(i) = CA^{(i-1)}B$ for $i > 0$.

$$\begin{aligned}
y(i) &= \begin{bmatrix} Y(0) & Y(1) & Y(2) & Y(3) & \cdots & Y(i-1) & Y(i) \end{bmatrix} \begin{bmatrix} u(i) \\ u(i-1) \\ u(i-2) \\ u(i-3) \\ \vdots \\ u(1) \\ u(0) \end{bmatrix} & (2) \\
&= \sum_{p=0}^{i} Y(i-p)\ u(p) & (3)
\end{aligned}$$

This is a discrete-time convolution of an input sequence with a sequence of Markov parameters. So the sequence of Markov parameters can be interpreted as the unit impulse response of the discrete-time system. Given a sequence of Markov parameters and an input sequence (and assuming an initial state of zero), the output sequence can be found from a linear matrix operation.

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} D & 0 & 0 & 0 & 0 & \cdots & 0 \\ CB & D & 0 & 0 & 0 & \cdots & 0 \\ CAB & CB & D & 0 & 0 & \cdots & 0 \\ CA^2B & CAB & CB & D & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & & & \\ CA^{N-1}B & CA^{N-2}B & CA^{N-3}B & \cdots & CAB & CB & D \end{bmatrix} \begin{bmatrix} u(0) \\ u(1) \\ u(2) \\ u(3) \\ \vdots \\ u(N) \end{bmatrix} \tag{4}$$

This matrix equation can by analyzed via the singular value decomposition of its lower-triangular Toeplitz matrix.

$$[y_N] = \left[Y \Sigma U^{\mathsf{T}} \right] [u_N] \tag{5}$$

Any input sequence equal to the j-th column of U results in an output sequence equal to $\Sigma_{j,j}$ times the j-th column of Y. The input sequence of the first column of U gives the largest magnification, in the sense that the ratio of norms $(||y_n||_2^2/||u_n||_2^2)$ is maximized and is $\Sigma_{1,1}$.

Given a long sequence of input and output data (long enough to strip-out the transient response from initial state $x(0)$), Markov parameters can be estimated in a least-squares sense

$$\begin{bmatrix} y(k) & y(k+1) & \cdots & y(j+k) \end{bmatrix} = \begin{bmatrix} Y(k) & Y(k-1) & \cdots & Y(1) & Y(0) \end{bmatrix} \begin{bmatrix} u(0) & u(1) & \cdots & u(j) \\ u(1) & u(2) & \cdots & u(j+1) \\ \vdots & \vdots & & \vdots \\ u(k-1) & u(k) & \cdots & u(j+k-1) \\ u(k) & u(k+1) & \cdots & u(j+k) \end{bmatrix} \tag{6}$$

Which can be represented more compactly by introducing new notation,

$$y_{k,j+1} = Y_{k,0} \; U_{0,k+1,j+1} \tag{7}$$

The normal equations to the ordinary least squares problem is

$$y_{k,j+1} \; U_{0,k+1,j+1}^{\mathsf{T}} = Y_{k,0} \; \left[\; U_{0,k+1,j+1} \; U_{0,k+1,j+1}^{\mathsf{T}} \; \right] \tag{8}$$

2 Block Hankel Matrices of Markov Parameters

Define a finite-dimensional block Hankel matrix of the Markov parameters of a discrete-time LTI system

$$H(0) = \begin{bmatrix} Y(1) & Y(2) & \cdots & Y(j) \\ Y(2) & Y(3) & \cdots & Y(j+1) \\ \vdots & \vdots & \ddots & \vdots \\ Y(k) & Y(k+1) & \cdots & Y(j+k-1) \end{bmatrix} \tag{9}$$

The Hankel matrix $H(0)$ is the product of the observability and controllability matrices,

$$H(0) = \mathcal{O}_k \mathcal{C}_j \tag{10}$$

The number of block-rows and block-columns of $H(0)$ (k and j) could be set to just under half the length of the sequence of Markov parameters, $k = j = K/2 - 1$

If the LTI is observable and controllable, and the Markov parameters are noise-free, then the rank of $H(0)$ is the model order, n. If the Markov parameters contain measurement noise then $H(0)$ can be full-rank, but the spectrum of singular values of $H(0)$ can guide in the selection of the model order.

3 Product of Observability and Controllability and Eigensystem Realization

The product of the Observability and Controllability matrices is a Hankel matrix of Markov parameters, $Y(k) = CA^{(k-1)}B$.

$$
\begin{aligned}
H(0) &= \mathcal{O}_k \mathcal{C}_j \\
&= \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{(k-1)} \end{bmatrix} \begin{bmatrix} B \; AB \; A^2B \; \cdots \; A^{(j-2)}B \; A^{(j-1)}B \end{bmatrix} \\
&= \begin{bmatrix} CB & CAB & CA^2B & \cdots & CA^{(j-2)}B & CA^{(j-1)}B \\ CAB & CA^2B & CA^3B & \cdots & CA^{(j-1)}B & CA^{(j)}B \\ CA^2B & CA^3B & CA^4B & \cdots & CA^{(j)}B & CA^{(j+1)}B \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ CA^{(k-1)}B & CA^{(k)}B & CA^{(k+1)}B & \cdots & CA^{(k-1+j-2)}B & CA^{(k-1+j-1))}B \end{bmatrix} \qquad (11) \\
&= \begin{bmatrix} Y(1) & Y(2) & Y(3) & \cdots & Y(j-1) & Y(j) \\ Y(2) & Y(3) & Y(4) & \cdots & Y(j) & Y(j+1) \\ Y(3) & Y(4) & Y(5) & \cdots & Y(j+1) & Y(j+2) \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ Y(k) & Y(k+1) & Y(k+2) & \cdots & Y(k+j-2) & Y(k+j-1) \end{bmatrix} \qquad (12)
\end{aligned}
$$

$$
\begin{aligned}
H(1) &= \mathcal{O}_k A \mathcal{C}_j \\
&= \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{(k-1)} \end{bmatrix} A \begin{bmatrix} B \; AB \; A^2B \; \cdots \; A^{(j-2)}B \; A^{(j-1)}B \end{bmatrix} \\
&= \begin{bmatrix} CAB & CA^2B & CA^3B & \cdots & CA^{(j-1)}B & CA^{(j)}B \\ CA^2B & CA^3B & CA^4B & \cdots & CA^{(j)}B & CA^{(j+1)}B \\ CA^3B & CA^4B & CA^5B & \cdots & CA^{(j+1)}B & CA^{(j+2)}B \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ CA^{(k)}B & CA^{(k+1)}B & CA^{(k+2)}B & \cdots & CA^{(k+j-2)}B & CA^{(k+j-1)}B \end{bmatrix} \qquad (13) \\
&= \begin{bmatrix} Y(2) & Y(3) & Y(4) & \cdots & Y(j) & Y(j+1) \\ Y(3) & Y(4) & Y(5) & \cdots & Y(j+1) & Y(j+2) \\ Y(4) & Y(5) & Y(6) & \cdots & Y(j+2) & Y(j+3) \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ Y(k+1) & Y(k+2) & Y(k+3) & \cdots & Y(k+j-1) & Y(k+j) \end{bmatrix} \qquad (14)
\end{aligned}
$$

- Neither $H(0)$ nor $H(1)$ is necessarily symmetric, since the Markov parameters $Y(k)$ are m by r matrices.
- The estimate of the feedthrough matrix $\hat{D} = Y(0)$ is not a part of these block Hankel matrices.

The observability and controllability matrices can be extracted from the SVD of $H(0)$ in many different ways. First, truncate the SVD expansion of $H(0)$ to the n most-significant singular values. A plot of the spectrum of singular values frequently reveals the order of the system represented by the data (i.e., the number of dynamic states in the system).

$$H(0) \approx U_n \Sigma_n V_n^{\mathsf{T}} = (U_n \Sigma_n^{(1/2+p)})\ T\ T^{-1}\ (\Sigma_n^{(1/2-p)} V_n^{\mathsf{T}}) = \mathcal{O}_k \mathcal{C}_j \tag{15}$$

(for $-1/2 \leq p \leq 1/2$). The matrix T is an arbitrary unitary transformation matrix.

- for $p = -1/2$: $\mathcal{O}_k = U_n T$ and $\mathcal{C}_j = T^{-1} \Sigma_n V_n^{\mathsf{T}}$ from which the gramians are $P_n = I_n$ and $Q_n = \Sigma_n^2$
- for $p = 0$: $\mathcal{O}_k = U_n \Sigma_n^{1/2} T$ and $\mathcal{C}_j = T^{-1} \Sigma_n^{1/2} V_n^{\mathsf{T}}$, from which the gramians are diagonal and equal $P_n = \Sigma_n$ and $Q_j = \Sigma_n$. The resulting state-space realization is called *internally balanced* because it is as observable as it is controllable. (More on this later.)
- for $p = +1/2$: $\mathcal{O}_k = U_n \Sigma_n T$ and $\mathcal{C}_j = T^{-1} V_n^{\mathsf{T}}$ from which the gramians are $P_n = \Sigma_n^2$ and $Q_n = I_n$.

(Recall that $U_n^{\mathsf{T}} U_n = I_n$, $V_n^{\mathsf{T}} V_n = I_n$, $\Sigma_n = \mathrm{diag}(\sigma_1, \sigma_2, \cdots, \sigma_n)$, $P_n = \mathcal{O}_k^{\mathsf{T}} \mathcal{O}_k$, and $Q_n = \mathcal{C}_j \mathcal{C}_j^{\mathsf{T}}$) Note that the singular values of the Hankel matrix of Markov parameters are invariant to way that the SVD is split, (i.e., invariant to the basis of the state space). These are called the *Hankel singular values* of the system.

The choice of p affects the gramians, but does not affect the state-space realization, to within an arbitrary unitary transformation and to within the effects of finite-precision computation. For the cases itemized above,

- for $p = -1/2$:

$$\begin{aligned}
U\bar{T}^{-1}\ \bar{T}\Sigma V^{\mathsf{T}} &\quad = \quad U\Sigma^{1/2}T^{-1}\ T\Sigma^{1/2}V^{\mathsf{T}} \\
\mathcal{O} = U\bar{T}^{-1},\ \ \mathcal{C} = \bar{T}\Sigma V^{\mathsf{T}} &\qquad\qquad \mathcal{O} = U\Sigma^{1/2}T^{-1},\ \ \mathcal{C} = T\Sigma^{1/2}V^{\mathsf{T}} \\
\mathcal{O} = U\bar{T}^{-1} = U\Sigma^{1/2}T^{-1} &\qquad\qquad \mathcal{C} = \bar{T}\Sigma V^{\mathsf{T}} = T\Sigma^{1/2}V^{\mathsf{T}} \\
\bar{T}^{-1} = \Sigma^{1/2}T^{-1} &\qquad\qquad \bar{T}\Sigma = T\Sigma^{1/2} \\
T &= \Sigma^{1/2}\bar{T}
\end{aligned}$$

- for $p = +1/2$:

$$\begin{aligned}
U\bar{\Sigma}T^{-1}\ \bar{T}V^{\mathsf{T}} &\quad = \quad U\Sigma^{1/2}T^{-1}\ T\Sigma^{1/2}V^{\mathsf{T}} \\
\mathcal{O} = U\Sigma\bar{T}^{-1},\ \ \mathcal{C} = \bar{T}V^{\mathsf{T}} &\qquad\qquad \mathcal{O} = U\Sigma^{1/2}T^{-1},\ \ \mathcal{C} = T\Sigma^{-1/2}V^{\mathsf{T}} \\
\mathcal{O} = U\Sigma\bar{T}^{-1} = U\Sigma^{1/2}T^{-1} &\qquad\qquad \mathcal{C} = \bar{T}V^{\mathsf{T}} = T\Sigma^{1/2}V^{\mathsf{T}} \\
\Sigma\bar{T}^{-1} = \Sigma^{1/2}T^{-1} &\qquad\qquad \bar{T} = T\Sigma^{1/2} \\
\bar{T} &= T\Sigma^{1/2}
\end{aligned}$$

So, the choice made in splitting the SVD of $H(0)$ affects the resulting observability matrix and controllability matrix only up to an arbitrary unitary transformation.

Continuing with the case of balanced realizations, ($\mathcal{O}_k = U_n\Sigma_n^{1/2}$ and $\mathcal{C}_j = \Sigma_n^{1/2}V^{\mathsf{T}}$), the dynamics matrix can be found from . . .

$$
\begin{aligned}
H(1) &= \mathcal{O}_k A \mathcal{C}_j = U_n \Sigma^{1/2}\ A\ \Sigma^{1/2} V_n^{\mathsf{T}} \\
U_n^{\mathsf{T}}\ H(1)\ V_n &= U_n^{\mathsf{T}} U_n \Sigma_n^{1/2}\ A\ \Sigma_n^{1/2} V_n^{\mathsf{T}} V_n = \Sigma_n^{1/2}\ A\ \Sigma_n^{1/2} \\
\Sigma^{-1/2} U_n^{\mathsf{T}}\ H(1)\ V_n \Sigma_n^{-1/2} &\equiv \hat{A}
\end{aligned}
\qquad (16)
$$

The estimate of the input matrix $\hat{B}$ is recovered as the first r columns of $\mathcal{C}_j$ and the estimate of the output matrix $\hat{C}$ is recovered as the first m rows of $\mathcal{O}_k$. and the estimate of the feedthrough matrix $\hat{D}$ is recovered from the first r columns of the sequence of Markov parameters, $Y(0)$ from (6). The resulting realization ($\hat{A}$, $\hat{B}$, $\hat{C}$, $\hat{D}$) is minimal, meaning that no pole matches any zero of the transfer function matrix, and is therefore equivalent to any other realization via a change of basis of the state-space, e.g., $x = T\bar{x}$,

$$\bar{x}(k+1) = T^{-1}AT\bar{x}(k) + T^{-1}Bu(k)$$

$$y(k+1) = CT\bar{x}(k) + Du(k)$$

4 Convolution and Covariance

The responses of a discrete time LTI system to random loading u that started an arbitrarily long time ago may be expressed by the convolution

$$y(i) = \sum_{p=-\infty}^{i} Y(i-p)\ u(p) \tag{17}$$

Assuming responses have a mean value of zero, the covariance of responses between time instances i and $i+l$ (that is, at time instances separated by a *lag* of l time steps) is

$$\Lambda_{yy}(l,i) = \mathbb{E}\left\{\ y(i+l)\ y^{\mathsf{T}}(i)\ \right\} \tag{18}$$

To derive an expression for the covariance in terms of the Markov parameters, we substitute the convolution (17) into the covariance (18),

$$\Lambda_{yy}(l,i) = \mathbb{E}\left\{ \sum_{p=-\infty}^{i} Y(i-p+l)\ u(p) \sum_{q=-\infty}^{i} u^{\mathsf{T}}(q)\ Y^{\mathsf{T}}(i-q)\ \right\} . \tag{19}$$

Since convolution and expectation are both linear operators, they may be interchanged,

$$\Lambda_{yy}(l,i) = \sum_{p=-\infty}^{i} \sum_{q=-\infty}^{i} \mathbb{E}\left\{ Y(i-p+l)\ u(p) u^{\mathsf{T}}(q)\ Y^{\mathsf{T}}(i-q)\ \right\} \tag{20}$$

and since the Markov parameters are not uncertain, the expectation need only be applied to the stochastic forcing

$$\Lambda_{yy}(l,i) = \sum_{p=-\infty}^{i} \sum_{q=-\infty}^{i} Y(i-p+l)\ \mathbb{E}\left\{ u(p) u^{\mathsf{T}}(q) \right\}\ Y^{\mathsf{T}}(i-q)\ . \tag{21}$$

Assuming that the stochastic inputs are fully uncorrelated unit white noise sequences, the covariance between $u(p)$ and $u(q)$ is 1 only for $p = q$. So doing, $\mathbb{E}\left\{ u(p) u^{\mathsf{T}}(q) \right\} = I_r$; the double summation becomes a single summation since all terms with $p \neq q$ are zero; and the covariance relation is

$$\Lambda_{yy}(l,i) = \sum_{p=-\infty}^{i} Y(i-p+l)\ Y^{\mathsf{T}}(i-p)\ . \tag{22}$$

Shifting the index of summation from p to q, where $p = i-q$, we have $i-p = q$, $p = i \leftrightarrow q = 0$, $p = -\infty \leftrightarrow q = +\infty$, and

$$\Lambda_{yy}(l) = \sum_{q=0}^{\infty} Y(q+l)\ Y^{\mathsf{T}}(q)\ . \tag{23}$$

Hence, under the assumption of uncorrelated unit white noise inputs, the output covariance matrices are identical to the Markov parameter covariances. This assumption is invoked in the ERA-NExT algorithm (Sandia report 92-1666).

In practice, data starting an arbitrarily long time ago ($u(p = -\infty)$) is not available. For (stable) systems in which the Markov parameters decay exponentially to zero, we can approximate the covariance with a truncated summation of j terms, which may be shifted in time by a variable index i,

$$\Lambda_{yy}(l,i) \approx R_{yy}(l,i) = \sum_{q=0}^{j-1} Y(q+i+l)\; Y^{\mathsf{T}}(q+i)\ . \tag{24}$$

With the substitutions and approximations above, covariances estimates may be computed via the outer product of the block Hankel matrices of Markov parameters, with values of i from 1 to k.

$$R_{yy}(l,i) = H(0)H(0)^{\mathsf{T}} = \begin{bmatrix} Y(1) & Y(2) & \cdots & Y(j) \\ Y(2) & Y(3) & \cdots & Y(j+1) \\ \vdots & \vdots & & \vdots \\ Y(k) & Y(k+1) & \cdots & Y(j+k-1) \end{bmatrix} \begin{bmatrix} Y^{\mathsf{T}}(1) & Y^{\mathsf{T}}(2) & \cdots & Y^{\mathsf{T}}(k) \\ Y^{\mathsf{T}}(2) & Y^{\mathsf{T}}(3) & \cdots & Y^{\mathsf{T}}(k+1) \\ \vdots & \vdots & & \vdots \\ Y^{\mathsf{T}}(j) & Y^{\mathsf{T}}(j+1) & \cdots & Y^{\mathsf{T}}(j+k-1) \end{bmatrix} \tag{25}$$

Here, j are the number of terms in the summation to approximate the covariance $R_{yy}(l,i)$ and there are k lag values l from 0 to $k-1$. This is examined below by considering individual summations within $R_{yy}(l,i)$

$$R_{yy}(l,i) = \begin{bmatrix} \sum Y(q+1)Y^{\mathsf{T}}(q+1) & \sum Y(q+1)Y^{\mathsf{T}}(q+2) & \cdots & \sum Y(q+1)Y^{\mathsf{T}}(q+k) \\ \sum Y(q+2)Y^{\mathsf{T}}(q+1) & \sum Y(q+2)Y^{\mathsf{T}}(q+2) & \cdots & \sum Y(q+2)Y^{\mathsf{T}}(q+k) \\ \vdots & \vdots & & \vdots \\ \sum Y(q+k)Y^{\mathsf{T}}(q+1) & \sum Y(q+k)Y^{\mathsf{T}}(q+2) & \cdots & \sum Y(q+k)Y^{\mathsf{T}}(q+k) \end{bmatrix} \tag{26}$$

where all sums are over indices $q = 0$ to $q = j-1$. From here we see that

$$R_{yy}(l,i) = \begin{bmatrix} R_{yy}(0,1) & R_{yy}(-1,2) & \cdots & R_{yy}((k-1),k) \\ R_{yy}(1,1) & R_{yy}(0,2) & \cdots & R_{yy}((k-2),k) \\ \vdots & \vdots & & \vdots \\ R_{yy}(k-1,1) & R_{yy}(k-2,2) & \cdots & R_{yy}(0,k) \end{bmatrix} \tag{27}$$

Here we see that the block diagonals represent the covariances with a lag l of zero and the columns represent the starting point of the summation i. In application, Markov sequence lengths j should be longer than roughly 10 or 20 times $\max_k(1/(\zeta_k\omega_k\Delta t))$ where ω_k and ζ_k are the natural frequency and damping ratio of the k-th mode, and Δt is the time step increment.

5 Eigensystem Realization with Data Correlation

In many cases, measurement noise will propagate to noise the sequence of estimated Markov parameters. In such cases, the application of the Eigensystem Realization Algorithm to covariance matrices of Markov parameters can serve to average-out the effect of noise on the estimated state-space realization.

Define the "auto-covariance" of Markov Parameters as

$$\mathcal{R}(0) = H(0)H^{\mathsf{T}}(0) = [\mathcal{O}_k\mathcal{C}_j]\left[\mathcal{C}_j^{\mathsf{T}}\mathcal{O}_k^{\mathsf{T}}\right] = \mathcal{O}_k Q_\gamma \tag{28}$$

and the "cross-covariance" of Markov Parameters as

$$\mathcal{R}(1) = H(1)H^{\mathsf{T}}(0) = \mathcal{O}_k A\mathcal{C}_j\mathcal{C}_j^{\mathsf{T}}\mathcal{O}_k^{\mathsf{T}} = \mathcal{O}_k A Q_\gamma \tag{29}$$

where $Q_\gamma \equiv \mathcal{C}_j\mathcal{C}_j^{\mathsf{T}}\mathcal{O}_k^{\mathsf{T}}$.

Note here that larger values of j will result in more averaging in estimation of the auto-covariance values. For a fixed length of Markov parameters, $k+j-1$, increasing j will result in a smaller auto-covariance matrix, $\mathcal{R}$, i.e., fewer time-lags (k) in the covariance matrices.

The Eigensystem Realization Algorithm can then proceed with a truncated SVD

$$\mathcal{R}(0) = \mathcal{O}_k Q_\gamma \approx U_n \Sigma_n V_n^{\mathsf{T}} \ ,$$

from which, $\mathcal{O}_k = U_n\Sigma_n^{1/2}$ and $Q_\gamma = \Sigma_n^{1/2}V_n^{\mathsf{T}}$. And since $\mathcal{R}(1) = \mathcal{O}_k A Q_\gamma$,

$$\hat{A} = \Sigma_n^{-1/2}U_n^{\mathsf{T}}\mathcal{R}(1)V_n\Sigma_n^{-1/2}. \tag{30}$$

And, finally, since $\mathcal{O}_k\mathcal{C}_j = H(0)$,

$$\mathcal{C}_j = \mathcal{O}_k^{+}H(0) \tag{31}$$

and $\hat{B}$ is recovered from the first r columns of $\mathcal{C}_j$ and and $\hat{C}$ is recovered from the first m rows of $\mathcal{O}_k$.

Note that even though the SVD of $\mathcal{R}(0)$ is split as for a balanced realization, the estimated realization from the ERA-DC method $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ is not internally balanced.

6 Eigensystem Realization for Stochastic Forcing

If the sequence of input data $[u(1), ..., u(j+k)]$ is not measured and can be assumed to be uncorrelated unit white noise, (or if the system is in free-response, then the ERA-DC method can proceed from covariance matrices of Markov parameters $\mathcal{R}(0)$ and $\mathcal{R}(1)$ estimated as the covariances of the measured output data.

The equivalence of covariances the output data (18) and the Markov parameters (24) (under the assumption of uncorrelated unit white noise forcing) shows that the covariance matrices of the Markov parameters can be found directly from Hankel matrices of the data, without the need to first compute Markov parameters.

The auto-covariance of the measured output data is

$$\mathcal{R}(0) = H(0)H(0)^{\mathsf{T}} = \begin{bmatrix} y(1) & y(2) & \cdots & y(j) \\ y(2) & y(3) & \cdots & y(j+1) \\ \vdots & \vdots & & \vdots \\ y(k) & y(k+1) & \cdots & y(j+k-1) \end{bmatrix} \begin{bmatrix} y^{\mathsf{T}}(1) & y^{\mathsf{T}}(2) & \cdots & y^{\mathsf{T}}(k) \\ y^{\mathsf{T}}(2) & y^{\mathsf{T}}(3) & \cdots & y^{\mathsf{T}}(k+1) \\ \vdots & \vdots & & \vdots \\ y^{\mathsf{T}}(j) & y^{\mathsf{T}}(j+1) & \cdots & y^{\mathsf{T}}(j+k-1) \end{bmatrix} \quad (32)$$

and the cross-covariance of the measured output data is

$$\mathcal{R}(1) = H(1)H(0)^{\mathsf{T}} = \begin{bmatrix} y(2) & y(3) & \cdots & y(j+1) \\ y(3) & y(4) & \cdots & y(j+2) \\ \vdots & \vdots & & \vdots \\ y(k+1) & y(k+2) & \cdots & y(j+k) \end{bmatrix} \begin{bmatrix} y^{\mathsf{T}}(1) & y^{\mathsf{T}}(2) & \cdots & y^{\mathsf{T}}(k) \\ y^{\mathsf{T}}(2) & y^{\mathsf{T}}(3) & \cdots & y^{\mathsf{T}}(k+1) \\ \vdots & \vdots & & \vdots \\ y^{\mathsf{T}}(j) & y^{\mathsf{T}}(j+1) & \cdots & y^{\mathsf{T}}(j+k-1) \end{bmatrix} \quad (33)$$

The use of covariance matrices $\mathcal{R}(0)$ and $\mathcal{R}(1)$ computed from Hankel matrices of the output data sequence $[y(1), ..., y(j+k)]$ in the ERA-DC method is called Eigensystem Realization for Natural Excitation Technique (ERA-NExT)[1].

Note that since ERA-NExT starts with computing $\mathcal{R}(0)$ and $\mathcal{R}(1)$ from Hankel matrices of $[y(1), ..., y(n+k)]$ without first finding Markov parameters. Hankel matrices of Markov parameters $H(0)$ and $H(1)$ are not involved, and the controllability matrix $\mathcal{C}_j$ nor the input matrix estimate $\hat{B}$ can not be computed via ERA-NExT.

[1]This title implies that natural excitations are uncorrelated unit white noise.

7 Observers

Sometimes, it is helpful to identify a system that is a modification of the actual system from which measurements are taken, because the identification process can be more stable for the modified system.

For example, a discrete time linear system can be simply re-written as

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + Gy(k) - Gy(k) & (34) \\ y(k) &= Cx(k) + Du(k) & (35) \end{aligned}$$

which can be re-written

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + GCx(k) + GDu(k) - Gy(k) & (36) \\ &= (A+GC)x(k) + (B+GD)u(k) - Gy(k) & (37) \\ &= [A+GC]x(k) + [B+GD \;,\; \; -G] \left[\begin{array}{c} u(k) \\ y(k) \end{array} \right] & (38) \\ &= \bar{A}x(k) + \bar{B}v(k) & (39) \\ y(k) &= Cx(k) + Du(k) & (40) \end{aligned}$$

where

$$\begin{aligned} \bar{A} &= (A+GC) \;, & (41) \\ \bar{B} &= [B+GD \;,\; -G] \;, \quad \text{and} & (42) \\ v(k) &= \left[\begin{array}{c} u(k) \\ y(k) \end{array} \right] & (43) \end{aligned}$$

By designing the matrix G appropriately, the dynamics matrix of the modified system dynamics matrix $[A+GC]$ can be made much more heavily damped than the dynamics of the original system, A. For example in equation (39) if the matrix G is set equal to the Kalman filter gain, and if $y(k)$ is replaced by output estimation error, then the state x(k) becomes an optimal estimate of the true state. This modification is intrinsic to the success of the Observer/Kalman Identification (OKID) method used in system identification.

8 Estimation of the Observer model Markov parameters

The inputs $(u(k), y(k))$ and outputs $(y(k))$ of the observer system (39) and (40) involve the same quantities as the input and output of the ERA system, but they are just organized in a different way.

A sequence of $k+1$ Markov parameters for the observer model can be estimated from

$$\left[\begin{array}{cccc} y(k) & y(k+1) & \cdots & y(N) \end{array} \right] = \left[\begin{array}{ccccc} D & C\bar{B} & C\bar{A}\bar{B} & \cdots & C\bar{A}^{k-1}\bar{B} \end{array} \right] \left[\begin{array}{cccc} u(k) & u(k+1) & \cdots & u(N) \\ v(k-1) & v(k) & \cdots & v(N-1) \\ v(k-2) & v(k-1) & \cdots & v(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ v(0) & v(1) & \cdots & v(N-k) \end{array} \right] \tag{44}$$

$$y(k) = Du(k) + \sum_{i=1}^{k} \bar{Y}(i) v(k-i) \tag{45}$$

$$y(k) = Du(k) + \sum_{i=1}^{k} [\bar{Y}^{(1)}(i) \ , \ \bar{Y}^{(2)}(i)] \left[\begin{array}{c} u(k-i) \\ y(k-i) \end{array} \right] \tag{46}$$

$$y(k) = Du(k) + \sum_{i=1}^{k} \bar{Y}^{(1)}(i) u(k-i) + \sum_{i=1}^{k} \bar{Y}^{(2)}(i) y(k-i) \tag{47}$$

where the observer Markov parameters $\bar{Y}_k$ are partitioned according to two sub-matrices. The first r columns fill $\bar{Y}^{(1)}(k)$ which multiplies the input $u(k)$ and the last m columns fill $\bar{Y}^{(2)}(k)$ which multiplies the output $y(k)$.

$$\begin{aligned} \bar{Y}(k) &= \left[\ \bar{Y}^{(1)}(k) \ , \quad \bar{Y}^{(2)}(k) \ \right] && (48) \\ &= \left[\ C\bar{A}^{k-1}\bar{B} \ \right] && (49) \\ &= \left[\ C \ [A+GC]^{k-1} \ [B+GD \ , \ -G] \ \right] && (50) \\ &= \left[\ C \ [A+GC]^{k-1} \ [B+GD] \ , \quad -C \ [A+GC]^{k-1} \ G \ \right] && (51) \end{aligned}$$

Do the observer Markov parameters as estimated in equation (44) estimate the *one-step-ahead* predictor for $y(k)$? In other words, the model parameters are defined by a projection of the measured responses $y(k)$ onto a basis that involves the current and previous inputs $[u(k), u(k-1), ...]$ and the previous *measured* responses $[y(k-1), y(k-2), ...]$. (!?)

9 References

1. James A. Cadzow, "Least Squares, Modeling and Signal Processing," *Digital Signal Processing*, 4 (1994) 2-20.

2. James A. Cadzow, "Total Least Squares, Matrix Enhancement, and Signal Processing," *Digital Signal Processing*, 4 (1994) 21-39.

3. George H. James III, Thomas G. Carne, and James P. Lauffer, "The Natural Excitation Technique (NExT) for Modal Parameter Extraction from Operating Wind Turbines," SANDIA Report SAND92-1666.UC-261, February 1993.

4. Jer-Nan Juang and Richard S. Pappa, "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction," *J. Guidance*, 8(5) (1985) 620-627.

5. Jer-Nan Juang and Richard S. Pappa, "Effects of Noise on Modal Parameters Identified by the Eigensystem Realization Algorithm," *J. Guidance*, 9(3) (1986) 294-303.

6. Jer-Nan Juang, Minh Phan, Lucas G. Horta and Richard W. Longman, "Identification of Observer/Kalman Filter Markov Parameters: Theory and Experiments," *J. Guidance, Control, and Dynamics*, 16(2) (1993) 320-329.

10 .m-functions

```
function Q = ctrb(A,B,p)
% Q = ctrb(A,B,p)
% Form the controllability matrix Q = [B AB A^2B A^3B ...  A^(p-1)B ]
% for the discrete time system, x(k+1) = A*x(k) + B*u(k)

 [n,r] = size(B);                                  % number of states and inputs
 if nargin < 3, p = n; else p = max(p,n); end      % number of blocks in Q
 Q = zeros(n, r*p);                                % dimension of Q
 Q(:,1:r) = B;                                     % first block is input matrix
 for k=1:p-1
        AkB = Q(:,(k-1)*r+1:(k-1)*r+r);
        Q(:,k*r+1:k*r+r) =  A * AkB;
 end
% ---------------------------------------------------------------------- CTRB
```

```
function P = obsv(A,C,p)
% P = obsv(A,C,p)
% Form the observability matrix P = [ C ; CA ; CA^2 ; CA^3 ; ... ; C A^(p-1) ]
% for the discrete time system, x(k+1) = A*x(k); y(k) = C*x(k)

 [m,n] = size(C);                                  % number of outputs and states
 if nargin < 3, p = n; else p = max(p,n); end      % number of blocks in P
 P = zeros(m*p,n);                                 % dimension of P
 P(1:m,:) = C;                                     % first block is output matrix
 for k=1:p-1
        CAk = P((k-1)*m+1:(k-1)*m+m,:);
        P(k*m+1:k*m+m,:) = CAk * A;
 end
% ---------------------------------------------------------------------- OBSV
```

```
>> A = [  1  0.5 ; -0.5 0.7 ];  B = [ 1 ; -1 ];  C = [ 1 2 ];  D = 0;  % discrete time SISO
>> n  = 100;
>> dt = 0.05;  % time step, s
>> t  = [0:n]*dt;

>> P = obsv(A,C,n)              % observability matrix
>> Q = ctrb(A,B,n)              % controllability matrix

>> Ymp = [ D  C*ctrb(A,B,n) ]   % Markov parameters
   Ymp = 0.00000  -1.00000  -1.90000  -2.28000  -2.07100  -1.35470  -0.33554  ...

>> Yt = tril(toeplitz(Ymp));    % lower triangular Toeplitz matrix ... works for SISO only
>> [Y,S,U] = svd(Yt);           % singular value decomposition
>> j = 1;
>> plot(t,U(:,j),'--k', t,Yt*U(:,j),'-b', t,S(j,j)*Y(:,j),'--r')
>> legend('u_j', 'Ymp*U_j', '\Sigma_{jj}*Y_j')
```