Time-domain subspace identification

State-space plant model

\[ x(k + 1) = Ax(k) + Bu(k) + w(k) \]
\[ y(k) = Cx(k) + Du(k) + v(k), \]

with dimensions, \( x(k) \in \mathcal{R}^n \), \( u(k) \in \mathcal{R}^{nu} \) and \( y(k) \in \mathcal{R}^ny \).

We are considering both “process noise”, \( w(k) \), and output noise, \( v(k) \).
Time-domain subspace identification

Approach

- Form a matrix (from the data) such that its range is spanned by the extended observability matrix, $O$.
- Use an SVD to estimate the plant order and give the vectors spanning $O$.
- Estimate $\hat{C}$ and $\hat{A}$, via least-squares, from the representation of $O$.
- Estimate $\hat{B}$ and $\hat{D}$ by a least-squares fit to the data.

This is very similar in concept to the frequency domain approach.

Data matrix construction

Consider the output as a function of past inputs and a past state;

\[ y(k + r) = Cx(k + r) + Du(k + r) + v(k + r) \]

\[ = CAx(k + r - 1) + CBu(k + r - 1) + Cw(k + r - 1) + Du(k + r) + v(k + r) \]

\[ \vdots \]

\[ = CA^r x(k) + \]

\[ CA^{r-1} Bu(k) + CA^{r-2} Bu(k + 1) \cdots CBu(k + r - 1) + Du(k + r) \]

\[ + CA^{r-1} w(k) + CA^{r-2} w(k + 1) \cdots Cw(k + r - 1) \]

\[ + v(k + r). \]

The output $y(k + r)$ depends upon:

- $x(k)$,
- $u(k), u(k + 1), \ldots, u(k + r)$,
- $w(k), w(k + 1), \ldots, w(k + r - 1)$,
- $v(k + r)$.
Data matrix construction

Repeat this expansion for \( y(k), y(k + 1), \ldots, y(k + r - 1) \).

Define:

\[
Y_r(k) = \begin{bmatrix}
y(k) \\
y(k + 1) \\
\vdots \\
y(k + r - 1)
\end{bmatrix}, \quad \text{and} \quad U_r(k) = \begin{bmatrix}
u(k) \\
u(k + 1) \\
\vdots \\
u(k + r - 1)
\end{bmatrix},
\]

Then,

\[
Y_r(k) = \begin{bmatrix}
C \\
CA \\
\vdots \\
CA^{r-1}
\end{bmatrix} x(k) + \begin{bmatrix}
D \\
CB \\
\vdots \\
CA^{r-2}B
\end{bmatrix} U_r(k) + N_r(k)
\]

\[
= O_r x(k) + S_r U_r(k) + N_r(k)
\]

Noise contribution

\[
N_r(k) = \begin{bmatrix}
0 & 0 \\
C & 0 \\
\vdots \\
CA^{r-2} & \ldots & CA & 0
\end{bmatrix} W_r(k) + V_r(k),
\]

where,

\[
W_r(k) = \begin{bmatrix}
w(k) \\
w(k + 1) \\
\vdots \\
w(k + r - 1)
\end{bmatrix}, \quad \text{and} \quad V_r(k) = \begin{bmatrix}
v(k) \\
v(k + 1) \\
\vdots \\
v(k + r - 1)
\end{bmatrix}.
\]
Matrix formulation

Create a matrix by stacking these equations side-by-side with each incremented in time by one.

Define:

\[
Y = \begin{bmatrix} Y_r(1) & Y_r(2) & \ldots & Y_r(N) \end{bmatrix},
\]
\[
X = \begin{bmatrix} x(1) & x(2) & \ldots & x(N) \end{bmatrix},
\]
\[
U = \begin{bmatrix} U_r(1) & U_r(2) & \ldots & U_r(N) \end{bmatrix},
\]
\[
W = \begin{bmatrix} W_r(1) & W_r(2) & \ldots & W_r(N) \end{bmatrix},
\]
\[
V = \begin{bmatrix} V_r(1) & V_r(2) & \ldots & V_r(N) \end{bmatrix},
\]

So,

\[
Y = \begin{bmatrix}
  y(k) & y(k+1) & \ldots & y(k+N-1) \\
  y(k+1) & y(k+2) & \ldots & y(k+N) \\
  \vdots & \vdots & \ddots & \vdots \\
  y(k+r-1) & y(k+r) & \ldots & y(k+r+N-2)
\end{bmatrix}.
\]

Block Toeplitz \( Y \in \mathbb{R}^{r n_u \times N} \)

Matrix formulation

Then,

\[
Y = O_r X + S_r U + Q_r W + V.
\]

To remove the effect of \( U \) we choose the dimensions such that \( r n_u < N \) and \( U \) has a non-trivial null-space.

\[
\Pi_{UT}^\perp = I - U^T (UU^T)^{-1} U,
\]

is the projection onto the space perpendicular to \( U^T \).

\[
U \Pi_{UT}^\perp = U - UU^T (UU^T)^{-1} U = U - U = 0.
\]

Multiplying on the right gives,

\[
Y \Pi_{UT}^\perp = O_r X \Pi_{UT}^\perp + (Q_r W + V) \Pi_{UT}^\perp
\]

\[
= O_r X \Pi_{UT}^\perp + Z \Pi_{UT}^\perp
\]
“Correctness”

In the noise free case the algorithm will recover the true plant.

\[ Y\Pi_{UT}^\perp = O_r X\Pi_{UT}^\perp + Z\Pi_{UT}^\perp \]

noise term

**Noise-free case**

If \( w(k) = 0 \) and \( v(k) = 0 \) then,

\[ Y\Pi_{UT}^\perp = O_r X\Pi_{UT}^\perp \]

and so the range space of \( Y\Pi_{UT}^\perp \) is the extended observability space.

As in the frequency domain subspace ID case we can factorize this into the true \( C \) and \( A \) matrices exactly (up to a similarity transform).

---

**Correlation methods for noise removal**

\[ Y\Pi_{UT}^\perp = O_r X\Pi_{UT}^\perp + Z\Pi_{UT}^\perp \]

noise term

Correlate the noise term with variables that make it converge to zero.

Define:

\[ \Phi \in \mathcal{R}^{s \times N} \quad (s \geq N), \quad (\Phi \text{ will be chosen later}) \]

\[ \Phi = [\phi_s(1) \quad \phi_s(2) \quad \ldots \quad \phi_s(N)] \]

Multiply on the right by \( \Phi^T \) and normalize w.r.t. \( N \).

\[
\frac{1}{N} Y\Pi_{UT}^\perp \Phi^T = O_r \frac{1}{N} X\Pi_{UT}^\perp \Phi^T + \frac{1}{N} Z\Pi_{UT}^\perp \Phi^T \\
= O_r \bar{T}_N + \mathcal{V}_N.
\]
Correlation methods for noise removal

\[ \frac{1}{N} Y \Pi^\perp_{UT} \Phi^T = O_r \tilde{T}_N + V_N. \]

We want to select \( \phi_s(k) \) such that,

\[ \lim_{N \to \infty} V_N = \lim_{N \to \infty} \frac{1}{N} Z \Pi^\perp_{UT} \Phi^T = 0, \]

and

\[ \lim_{N \to \infty} \tilde{T}_N = \lim_{N \to \infty} \frac{1}{N} X \Pi^\perp_{UT} \Phi^T = \tilde{T} \]

has rank \( n \).

This ensures that the noise term, \( V_N \), goes to zero

and that the range space is \( O_r \).

\[ \lim_{N \to \infty} \frac{1}{N} Y \Pi^\perp_{UT} \Phi^T = O_r \tilde{T}. \]

This approach is analogous to the instrumental variable methods for prediction error problems.

Selecting \( \Phi \)

\[ \frac{1}{N} Z \Pi^\perp_{UT} \Phi^T = \frac{1}{N} \sum_{k=1}^{N} Z(k)\phi_s(k)^T - \]

\[ \frac{1}{N} \sum_{k=1}^{N} Z(k)U_r(k)^T \left( \frac{1}{N} \sum_{k=1}^{N} U_r(k)^T U_r(k) \right)^{-1} \frac{1}{N} \sum_{k=1}^{N} U_r(k)\phi_s(k)^T, \]

where \( Z(k) = Q_r W(k) + V(k) \).

Under mild conditions this converges to the expected values:

\[ \lim_{N \to \infty} \frac{1}{N} Z \Pi^\perp_{UT} \Phi^T = \]

\[ E\{Z(k)\phi_s(k)^T\} - E\{Z(k)U_r(k)^T\} R_u^{-1} E\{U_r(k)\phi_s(k)^T\}. \]

Note that \( R_u = E\{U_r(k)U_r(k)^T\} \) is invertible for persistently exciting inputs.
Selecting $\Phi$

\[
\lim_{N \to \infty} \frac{1}{N} Z \Pi_{U_T} \Phi^T = 
E\{Z(k)\phi_s(k)^T\} - \underbrace{E\{Z(k)U_r(k)^T\}}_{\text{zero if } v(k), \ w(k) \text{ and } u(k) \text{ uncorrelated}} R_u^{-1} E\{U_r(k)\phi_s(k)^T\}.
\]

So to get both terms zero we also require that $Z(k)$ and $\phi(k)$ are also uncorrelated. Invertibility of $R_u$ comes from persistency of excitation.

Typical choice — past inputs and outputs:

\[
\phi_s(k) = \begin{bmatrix}
  y(k - 1) \\
  \vdots \\
  y(k - s_1) \\
  u(k - 1) \\
  \vdots \\
  u(k - s_2)
\end{bmatrix}
\]

These are uncorrelated with the current noises, $w(k)$ and $v(k)$.

Selecting $\Phi$

Similarly for the $\tilde{T}$ term,

\[
\tilde{T} = \lim_{N \to \infty} \frac{1}{N} X \Pi_{U_T} \Phi^T = 
E\{x(k)\phi_s(k)^T\} - E\{x(k)U_r(k)^T\} R_u^{-1} E\{U_r(k)\phi_s(k)^T\}.
\]

It is more difficult to show that this has rank $n$ as required.

Observe that we need $\phi_s(k)$ to be correlated with both $x(k)$ and $u(k)$.

This supports the choice of previous inputs and outputs.
Algorithm:

1. Select matrix dimensions, $r$, $N$, etc. and the correlation variables, $\phi_s(k)$.
2. Given $y(k)$ and $u(k)$ form,
   \[ M = \frac{1}{N} Y \Pi_{U}^T \Phi^T. \]
3. Select invertible weighting matrices, $W_1$ and $W_2$ and calculate the SVD of the weighted matrix,
   \[ \text{svd}(W_1 MW_2) = U \Sigma V^T. \]
4. Select a model order, $\hat{n}$, from the SVD.
   \[ U \Sigma V^T \approx [U_1 \quad U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \]
   \[ \Sigma_1 \in \mathcal{R}^{\hat{n} \times \hat{n}}. \]

In the noise-free case $\Sigma_2 = 0$. We hope that it is still small in the noisy case.

5. Note that,
   \[ \text{span}(O_r) = \text{span}(W_1^{-1} U_1), \]
   but we can apply a weighting matrix, $R$, to give,
   \[ \hat{O}_r = W_1^{-1} U_1 R. \]
   (R weights the LS fit that follows in step 7)

6. Form the estimate $\hat{C}$ via,
   \[ \hat{C} = \hat{O}_r (1 : n_u, 1 : \hat{n}). \]
   (the top rows of $\hat{O}_r$)

7. Form the estimate $\hat{A}$ via LS fit to,
   \[ \hat{O}_r (n_u + 1 : r n_u, 1 : \hat{n}) = \hat{O}_r (1 : (r - 1) n_u, 1 : \hat{n}) \hat{A}. \]
Algorithm:

8. Estimate $\hat{B}$, $\hat{D}$ (and if necessary, $\hat{x}_0$) from the linear least-squares fit to the data:

$$\{\hat{B}, \hat{D}, \hat{x}_0\} = \arg\min_{B,D,x_0} \frac{1}{N} \sum_{k=1}^{N} \left\| y(k) - Du(k) - \left( \hat{C}(z^{-1} I - \hat{A})^{-1} \right) \left( x_0 \delta(k) + Bu(k) \right) \right\|^2_2$$

where, $\delta(k) = \begin{cases} 1 & \text{if } k = 0, \\ 0 & \text{otherwise} \end{cases}$

---

Algorithm design choices

**Correlation vector:** $\phi_s(k)$

We typically choose a combination of past inputs and past outputs. It must be uncorrelated with respect to the current noises, $w(k)$ and $v(k)$. It must be correlated with respect to the plant state and inputs.

Some algorithms use only past inputs in $\phi_s(k)$ which gives an “output-error” type of formulation. For example: OE-MOESP.

**Prediction horizon:** $r$

We clearly need $r > n$. We frequently also choose $r = s$.

**Least-squares weighting:** $R$

Typically, $R = I$, $R = \Sigma_s$ or $R = \Sigma_s^{1/2}$. 
Algorithm design choices

Weighting matrices: $W_1$ and $W_2$

This choice affects the noise properties of the algorithms.

- **MOESP** (Verhaegen)

  \[ W_1 = I, \quad W_2 = \left( \frac{1}{N} \Phi \Pi_{U^T} \Phi^T \right)^{-1} \Phi \Pi_{U^T}. \]

- **N4SID** (Van Overschee and DeMoor)

  \[ W_1 = I, \quad W_2 = \left( \frac{1}{N} \Phi \Pi_{U^T} \Phi^T \right)^{-1} \Phi. \]

- **IVM** (Viberg)

  \[ W_1 = \left( \frac{1}{N} Y \Pi_{U^T} Y^T \right)^{-1/2}, \quad W_2 = \left( \frac{1}{N} \Phi \Phi^T \right)^{-1/2}. \]

- **CVA** (Larimore)

  \[ W_1 = \left( \frac{1}{N} Y \Pi_{U^T} Y^T \right)^{-1/2}, \quad W_2 = \left( \frac{1}{N} \Phi \Pi_{U^T} \Phi^T \right)^{-1/2}. \]

Bibliography

**Subspace identification**


