Frequency Domain MIMO Identification for Modeling of Structural Dynamics *

Wei Tang

College of Automation, Northwestern Polytechnical University,
Xi’an, 710072, P. R. China. (e-mail: tangwei@nwpu.edu.cn)

Abstract: Frequency domain system identification is widely used in the area of structural dynamic analysis. However, due to modeling errors, disturbances and measurement noise, the identified model may be unstable. In this paper, we propose two algorithms to identify structural dynamics with prior knowledge of stable poles. The first algorithm is based on subspace identification, and the second algorithm is based on maximum likelihood in the frequency domain. The common difference with previous methods in that the new algorithms consider the poles constraint. We show the benefits of the proposed approaches in a simple example where the results of the first algorithm are used as an initial estimate for the second algorithm.

Keywords: System identification; State-space model; MIMO; Subspace identification; Maximum likelihood

1. INTRODUCTION

State-space identification has received considerable attentions in the last decades. Some new methods such as subspace identification (SI), Maximum Likelihood (ML) have been developed to estimate system in state-space form. In addition, some of these methods have been successfully applied to Modelling of structural dynamics. However, in practice, due to disturbing noise and/or nonlinear distortions, the identified state-space model may be unstable. This is against the physical nature of the system. To overcome this problem and to enforce stability of the identified model, different approaches have been proposed in the literatures. (Chui & Maciejowski, 1996; Lacy & Bernstein, 2003; Van Gestel, Suykens, Van Dooren, & De Moor, 2001). The common feature of these methods is that they all guarantee the poles of the identified model to be inside the unit circle by using different constraints.

Recently, the development of “automatic poles selection techniques” (Peeters, Lau, & J. Lanslot, 2008; Vanlanduit et al., 2003) for vibration analysis provides an obvious alternative. These methods allow us to select stable poles form computational poles. Motivated by these issues, we may set up a stable model just by use of the stable poles. If these techniques provide a sufficiently good estimate of the stable poles, then the next question will be how to find the optimal set of state matrices \((A, B, C, D)\) that satisfy the fixed poles constraint? There are few studies in literature for dealing with this problem. The current most popular method is to identify the transfer function model by using orthonormal basis, and then transform it to a minimum realization (Hoog, 2001; Ohta, 2005). In this framework, the identification of a system in state-space form is not straightforward.

This paper explores frequency domain approaches that enable us to derive state-space model with fixed poles directly from measurements. Frequency domain identification of linear time-invariant systems has regained interest during the last years (Pintelon & Schoukens, 2001). It is preferred by the engineers due to some advantages, such as: (i) it is easy to compress data record, (ii) It is possible to combine data from experiments, and (iii) it is easy to decrease the noise level.

More specifically, the current work introduces two frequency domain algorithms for finding state-space realization with fixed poles. The first algorithm is based on a subspace identification algorithm proposed by (McKelvey, Akçay, & Ljung, 1996). The second algorithm is based on Maximum Likelihood (ML) estimation in frequency domain. The latter approach employs expectation–maximization (EM) algorithm. In (Agüero, Yuz, & Goodwin, 2007; Wills, Ninness & Gibson, 2009, Yuz, et al., 2010) The common difference with previous approaches is that state matrix A is pre-known, and this simplify the estimation algorithm since only \((B, C, D)\) need to be estimated. We show the benefits of the proposed approaches in a simple example where the results of the first algorithm are used as an initial estimate for the second algorithm.

This paper is organized as follows: In Section 2, the problem set-up is presented. Then, the first and second algorithms are detailed in Section 3 and 4 respectively. In Section 5 the algorithms are extended to continuous time identification. Simulation results are reported in section 6, and Section 7 concludes the paper.

*This research is supported by National Natural Science Foundation of China (No. 50905140), Research Fund for the Doctoral Program of Higher Education of China (No. 20096102120039) and NPU Aoxiang Star Plan.
2. PROBLEM FORMULATION

2.1 The System of Interest

We consider a discrete-time state-space model representation of a mechanical vibration system as:

\[
\begin{bmatrix}
x_{k+1} \\
y_k
\end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x_k \\
u_k
\end{bmatrix} + \begin{bmatrix} w_k \\
v_k
\end{bmatrix}
\]

(1)

where \( x \in \mathbb{R}^n \), \( u \in \mathbb{R}^m \), \( y \in \mathbb{R}^p \) are the state vector, measured input and output respectively. \( w_k \) and \( v_k \) represent the system and the sensors noises which are assumed zero mean Gaussian white noise sequences.

The frequency data is calculated from discrete measurement by using Discrete Fourier Transformation (DFT).

\[
F(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} f_k z_k^n \quad k = 1, \ldots, N
\]

where \( z_k = e^{j\omega(n)} = e^{2\pi ik/N} \), \( f_k \) is discrete time sequence with data length \( N \), and \( F(k) \) is the \( k \)th frequency line. In current paper, the frequency input and output data are represented as \( U(k) \) and \( Y(k) \).

Taking discrete Fourier transformation to both sides of Eq. (1) lead to extension of state-space model in frequency domain (Cauberghe et al., 2006), which takes into account the initial and final conditions of states.

\[
\begin{bmatrix} z_k X(k) \\
y(k)
\end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} X(k) \\
U(k)
\end{bmatrix} + \begin{bmatrix} \alpha \cdot z_k / \sqrt{N} \\
0
\end{bmatrix} + \begin{bmatrix} W(k) \\
Y(k)
\end{bmatrix}
\]

(2)

where \( \alpha = x_0 - x_y \), \( W(k) \) and \( Y(k) \) are the DFT of \( w_k \) and \( v_k \). Using the assumption, that \( w_k \) \( v_k \) are zero mean, Gaussian white noise, then \( \mathbb{N}(k) = [W(k)]^\Omega \) \( V(k)]^\Omega \) are independent zero mean real-complex proper random sequence that are normally distributed, and has probability density function:

\[
p(N(k)) = \frac{1}{2 \pi} \det(P) \frac{1}{2} \exp \left[ -\frac{1}{2} N(k)^T P^{-1} N(k) \right] N(k) \in \mathbb{R}^{m+n}, \quad P = \mathbf{Q} \delta_{i,k}
\]

where \( \delta_{i,k} \) is the Kronecker delta function, and

\[
\mathbf{Q} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{R} \end{bmatrix}
\]

2.2 Identification Problem

Assume we have data from the system in terms of samples of the Fourier transform of the input \( U(k) \) and output signals \( Y(k) \) at a set of \( M \) frequency lines. Then the goal is to find a set of matrices \( A, B, C, D \) in the state-space model from the given frequency domain. We assume that the poles of model are fixed at specific locations.

3. POLES CONSTRAINED SUBSPACE IDENTIFICATION

3.1 Subspace Identification for Extended State-Space Model

Subspace identification in the frequency domain has been developed e.g. in (McKelvey, Akçay, & Ljung, 1996; McKelvey, 2004). In this approaches, one estimates all the matrices that defines a state-space model (i.e. \( A, B, C, D \)). We next, propose a modification of these algorithms in order to take into account the extra term \( \alpha \) in (2).

Eq. (2) can be rewritten as

\[
\begin{bmatrix} z_k X(k) \\
y(k)
\end{bmatrix} = \begin{bmatrix} A & B^U \end{bmatrix} \begin{bmatrix} X(k) + B^U(k) \\
C \end{bmatrix} + \begin{bmatrix} W(k) \\
Y(k)
\end{bmatrix}
\]

(5)

where \( B^U = [B, \alpha U(k)] \) and \( \alpha U(k) = [U(k)]^T z_k / \sqrt{N} \).

Recursive use of Eq. (5) gives

\[
\begin{bmatrix} z_k X(k) \\
y_k Y(k)
\end{bmatrix} = \begin{bmatrix} 1 \\
C \end{bmatrix} N_k + \begin{bmatrix} 0 \\
\Delta N_k \end{bmatrix}
\]

(6)

where \( \Delta^k = [D \ 0] \). Writing Eq.(6) for \( p = 1, \ldots, M \), and stacking them on top of each other gives

\[
Z_p(k) Y(k) = O_p X(k) + ITZ_p(k) U'(k)
\]

(7)

with

\[
Z_p(k) = \begin{bmatrix} z_k^p \\
\Delta^p N_k \end{bmatrix}, \quad O_p = \begin{bmatrix} C \\
\Delta CA \end{bmatrix}
\]

(8)

\[
\Gamma = \begin{bmatrix} D^p \ 0 \ldots \ 0 \\
\Delta CA^p \ 0 \ldots 0 \\
\Delta CA^2 \ 0 \ldots 0 \\
\vdots \\
\Delta CA^{M-1} \ 0 \ldots 0 \\
\Delta CA^{M-2} \ 0 \ldots \ D^p \end{bmatrix}
\]

(9)

Collecting Eq. (7) for \( k = 1,2, \ldots, M \) with the number \( M \) of discrete frequency lines in the interesting frequency band gives

\[
Y_M = Q X_M + \Gamma U_M
\]

(10)

with

\[
Y_M = [Z_1(1) Y(1) \ Z_1(2) Y(2) \ldots \ Z_1(M) Y(M)]
\]

\[
U_M = [Z_1(1) U(1) \ Z_1(2) U(2) \ldots \ Z_1(M) U'(M)]
\]

\[
X_M = X(1) \ X(2) \ldots \ X(M)
\]

Eq. (10) is converted into a real set of equations.
\[ Y_M^{re} = O_M X_M^{re} + U_M^{re} \]  

where \(( \gamma^{re} = [\text{Re}(\gamma) \text{ Im}(\gamma)] \), \text{Re} and \text{Im} represent real and imaginary parts of a complex matrix.

Next, let us denote \( \Pi^{\perp} \) a matrix which projects onto the null space of \( U_M^{re} \). Since \( U_M^{re} \Pi^{\perp} = 0 \) we directly obtain \( Y_M^{re} \Pi^{\perp} = 0 \). In practice, this step is performed by QR-factorization, followed by singular value decomposition.

\[
\begin{bmatrix}
U_M^{re T} \\
\end{bmatrix} =
\begin{bmatrix}
R_{11}^T & 0 \\
R_{12}^T & R_{22}^T
\end{bmatrix}
\begin{bmatrix}
Q_1^T \\
Q_2^T
\end{bmatrix}
\]

Then, the extended observability matrix \( \hat{O}_r \) is estimated.

\[
\hat{O}_r = U_s
\]

The estimates of \( A \) and \( C \) from \( \hat{O}_r \) in a least-squares sense is given by

\[
\hat{\hat{A}} = \hat{O}_r [1:n, (r-1):] + \hat{\hat{\hat{C}}}[1:n, (r-1):] \\
\hat{\hat{C}} = \hat{\hat{O}}_r [1:n, :]
\]

where the notation (\( \dagger \)) denotes the pseudo-inverse of a matrix. With estimated \( \hat{\hat{A}}, \hat{\hat{C}} \), the matrices \( B, D \) is estimated in a least-squares sense from

\[
Y_k = \hat{\hat{C}}(z_k I - \hat{\hat{A}})^{-1} B U_k - D U_k
\]

3.2 Frequency Domain Subspace Identification For System With Known Poles

Classical subspace identification mentioned above just provides an estimate of \( (A, B, C, D) \) without consideration of the poles constraint. Consequently, the estimate \( \hat{A} \) directly calculated by Eq. (16) does not satisfy the poles constraint.

We next present a modification of this algorithm in order to consider this extra constraint.

It is reasonable to assume a real physical system has distinct poles, hence, assume \( A \) has distinct eigenvalues \( \lambda_i \) \((i = 1, 2, \ldots, n)\) with corresponding linear independent eigenvector \( p_i \).

For real eigenvalue \( \lambda_i \), there is eigenvalue equation

\[
A p_i = \lambda_i p_i
\]

For mechanical dynamical system, complex conjugated eigenvalues pair \( \lambda_i, \lambda_i^* \) are commonly used. In this case, eigenvalue equation is written as

\[
A [p_i, p_i^*] = [\lambda_i, \lambda_i^*]
\]

where * denotes conjugate of complex.

With Eq. (19) and (20), we have a similarity transformation of \( A \)

\[
A = P \Lambda \Lambda^{-1}
\]

where \( \Lambda = \text{diag}[\lambda_1, \lambda_2, \ldots, \lambda_n] \), \( P = [p_1, p_2, \ldots, p_n] \).

To estimate matrix \( A \) with fixed eigenvalues, it is clearly necessary to derive the similarity transformation matrix \( P \). This may be simply achieved via the results of the following Lemma.

Lemma 3.1. For a state-space model \( (A, B, C, D) \) with fixed poles, if \( \lambda_i \) is one of eigenvalues of \( A \), and \( p_i \) is the eigenvector corresponding to \( \lambda_i \). Then the eigenvector \( \hat{p}_i \) of matrix \( T^{-1} \hat{A} T \) corresponding to \( \lambda_i \) can be found by solving following equation

\[
\hat{A} \hat{p}_i = 0 \quad \text{subject to} \quad \hat{p}_i^H \hat{p}_i = 1
\]

with

\[
\hat{A} = \begin{bmatrix}
\lambda_1 \hat{C} \\
\vdots \\
\lambda_n \hat{C}
\end{bmatrix} - \hat{\hat{\hat{\hat{O}}}}_r
\]

where \( T \) is nonsingular matrix, and \( T = X^{re} \Pi Q V \Sigma^{-1} \).

Proof. As mentioned in (McKelvey, 2004), the estimated \( \hat{O}_r \) in Eq. (14) only represent the range space of \( \hat{O}_r \). If we set \( T = X^{re} \Pi Q V \Sigma^{-1} \), there is \( \hat{O}_r = \hat{O}_r T \). Thus, combining Eqs.(8), (14) yields

\[
\hat{O}_r = \begin{bmatrix}
C \\
CA \\
\vdots \\
CA^{-1}
\end{bmatrix} = \begin{bmatrix}
\hat{\hat{\hat{C}}} \\
\hat{\hat{\hat{C}}} A \\
\vdots \\
\hat{\hat{\hat{C}}} A^{-1}
\end{bmatrix} = \begin{bmatrix}
\hat{\hat{\hat{C}}} \\
\hat{\hat{\hat{C}}} A \\
\vdots \\
\hat{\hat{\hat{C}}} A^{-1}
\end{bmatrix}
\]

where \( \hat{\hat{\hat{C}}} = CT \), \( \hat{\hat{\hat{A}}} = T^{-1} \hat{A} T \). This implies that the state-space model derived above is actually a realization of original one.

Then, assume that \( \hat{p}_i \) is the eigenvector of \( \hat{\hat{\hat{A}}} \) corresponding to \( \lambda_i \). Multiplying \( \hat{p}_i \) from the right in Eq. (24), we have

\[
\hat{O}_r \hat{p}_i = \begin{bmatrix}
\hat{\hat{\hat{C}}} \\
\hat{\hat{\hat{C}}} A \\
\vdots \\
\hat{\hat{\hat{C}}} A^{-1}
\end{bmatrix} \hat{p}_i = \begin{bmatrix}
\lambda_i \hat{\hat{\hat{C}}} \\
\lambda_i \hat{\hat{\hat{C}}} A \\
\vdots \\
\lambda_i \hat{\hat{\hat{C}}} A^{-1}
\end{bmatrix} \hat{p}_i
\]

For simplicity, Eq. (25) is rewritten as \( \hat{A} \hat{p}_i = 0 \), and constraint \( \hat{p}_i^H \hat{p}_i = 1 \) is posed to ensure a unique nontrivial solution. A numerical solution is developed in order to be the right one.
singular vector of $\mathbf{A}$ corresponding to the smallest singular value.

Lemma 3.1 provides a method to calculate eigenvectors of $\hat{\mathbf{A}}$. By arranging the eigenvectors in columns, we get the transformation matrix $\hat{\mathbf{P}}$:

$$
\hat{\mathbf{P}} = \begin{bmatrix} \hat{p}_1 & \hat{p}_2 & \cdots & \hat{p}_n \end{bmatrix}
$$

Then an estimate of $\hat{\mathbf{A}}$ can be simply obtained by performing the similarity transformation $\hat{\mathbf{A}} = \hat{\mathbf{P}} \hat{\mathbf{A}} \hat{\mathbf{P}}^{-1}$. Using the estimates $\hat{\mathbf{A}}, \hat{\mathbf{C}}$, the matrices $\mathbf{B}', \mathbf{D}$ are estimated as mentioned in last subsection.

In practice, real mechanical system usually has complex conjugated eigenvalue pairs. In this case, the eigenvector corresponding to eigenvalue pairs are also conjugated. Hence, we may compute one of them, and make a conjugate transform to get another one. In this way, the estimate $\hat{\mathbf{A}}$ calculated by similarity transformation is ensured to be real matrix.

4 MAXIMUM LIKELIHOOD ESTIMATION USING EM ALGORITHM FOR SYSTEM WITH KNOWN POLES

4.1 Background On The EM Algorithm

Maximum Likelihood (ML) estimation enjoys popularity in the area of system identification. In the ML framework, the following log-likelihood function is maximized

$$
i(\theta) = \log p(\mathbf{Y})
$$

where $\mathbf{Y} = \{Y_1, Y_2, \ldots, Y_M\}$ denotes the given output data set. However, when dealing with state-space model, $i(\theta)$ is parameterized in quite a complicated way by $\theta$, this present significant difficulty. As such, some frequency domain ML estimator using EM algorithm is developed to avoid the difficulty.

The basic idea of the EM algorithm is to decompose the log-likelihood function as

$$
\log p(\mathbf{Y} | \theta) = \log p(\mathbf{X}, \mathbf{Y} | \theta) - \log p(\mathbf{X} | \theta)
$$

(26) where $\mathbf{X} = \{X_1, X_2, \ldots, X_M\}$ denotes the “hidden data” set. Here it is referred to the state of state-space model. Applying the conditional expectation operator $E \{ \cdot | \mathbf{Y}, \theta \}$ to both sides of Eq. (26) provide the expression as follows:

$$
\log p(\mathbf{Y} | \theta) = Q(\theta, \hat{\theta}) - H(\theta, \hat{\theta})
$$

(27) where

$$
Q(\theta, \hat{\theta}) = E \{ \log p(\mathbf{X}, \mathbf{Y} | \theta) | \mathbf{Y}, \hat{\theta} \}
$$

(28) and

$$
H(\theta, \hat{\theta}) = E \{ \log p(\mathbf{Y} | \theta) | \mathbf{Y}, \hat{\theta} \}
$$

(29)

By using the Jensen inequality, it is possible to prove $H(\theta, \hat{\theta}) \geq H(\theta, \hat{\theta})$ (Dempster, 1977). Therefore, any value of $\theta$ for which $Q(\theta, \hat{\theta}) \geq Q(\theta, \hat{\theta})$ implies $i(\theta) \geq i(\theta)$.

This suggests a strategy of maximizing $Q(\theta, \hat{\theta})$, which increases $i(\theta)$, and then setting $\theta_{t+1}$ equal to this maximize and repeating the process. That is well known as EM algorithm with following two steps:

1) E step

Calculate: $Q(\theta, \hat{\theta})$

2) M step

$$
\hat{\theta}_{t+1} = \arg \min_{\theta} Q(\theta, \hat{\theta})
$$

4.2 E step

It is clearly the subspace identification has provided a good estimate of $\hat{\mathbf{A}}$ that satisfies the poles constraint. Thus, we can assume $\hat{\mathbf{A}}$ is pre-known before performing EM algorithm. Consequently, the full parameterized identified problem is converted to finding $(\mathbf{B}, \mathbf{C}, \mathbf{D})$ parameterization for fixed $\hat{\mathbf{A}}$. To ensure the convergence of EM algorithm, the estimated results $(\hat{\mathbf{B}}, \hat{\mathbf{C}}, \hat{\mathbf{D}})$ from subspace identification are chosen to initialize the iterative procedure.

To implement the EM algorithm, the first step is to derive a method for calculation of $Q(\theta, \hat{\theta})$. This may be simply achieved via the results of the following Lemma.

**Lemma 4.1** The function $Q(\theta, \hat{\theta})$ defined in Eq.(29) can be expressed as:

$$
Q(\theta, \hat{\theta}) = F \log|\mathbf{Q}^{-1}| + F \log|\mathbf{R}^{-1}|
$$

$$
- \mathbf{B}' (\mathbf{Q}^{-1} [\mathbf{Q} \mathbf{R} \mathbf{T} - \mathbf{R} \mathbf{Q} T + \mathbf{O} \mathbf{T}^T])
$$

$$
- \mathbf{R}^{-1} (\mathbf{A} - \mathbf{A} T^T - \mathbf{T} \mathbf{A}^T + \mathbf{T} \mathbf{A}^T)
$$

(30)

where

$$
\mathbf{Q} = \mathbf{\Phi} + \mathbf{A} \Gamma \mathbf{T} - \mathbf{T} \mathbf{A}^T - \mathbf{A} \mathbf{T}^T
$$

(31)

$$
\mathbf{Q} = \mathbf{\Phi} - \mathbf{A} \Gamma[ \mathbf{A}^T - \mathbf{A} \mathbf{T}^T]
$$

(32)

$$
\mathbf{B} = \sum_{k=1}^{M} f_k \cdot E \{ X_k X_k^T | \mathbf{Y}_k, \hat{\theta} \}
$$

(33)

$$
\mathbf{B}_1 = \sum_{k=1}^{M} f_k \cdot E \{ X_k X_k^T | \mathbf{Y}_k, \hat{\theta} \}
$$

(34)

$$
\mathbf{B}_2 = \sum_{k=1}^{M} f_k \cdot E \{ X_k | \mathbf{Y}_k, \hat{\theta} \}
$$

(35)

$$
\mathbf{A} = \sum_{k=1}^{M} f_k \cdot Y_k \mathbf{T}_k
$$

(36)

$$
\mathbf{A} = \sum_{k=1}^{M} f_k \cdot E \{ X_k | \mathbf{Y}_k, \hat{\theta} \}
$$

(37)

$$
\mathbf{T} = \begin{bmatrix} I_{11} & I_{12} \\ I_{21} & I_{22} \end{bmatrix}
$$

(38)
\[
\sum_{k=1}^{M} f_k \cdot E \begin{bmatrix}
X_k X_k^H U_k U_k^H - U_k U_k^H U_k U_k^H X_k^H Y_k \\
U_k U_k^H - U_k U_k^H U_k U_k^H X_k^H Y_k \\
U_k U_k^H - U_k U_k^H U_k U_k^H X_k^H Y_k \\
U_k U_k^H - U_k U_k^H U_k U_k^H X_k^H Y_k \\
1/N 
\end{bmatrix}
\]

\( \Pi = \Gamma(1;(n_x + n_u), 1;(n_x + n_u)) \)  \hspace{1cm} \text{(39)}

\( F = \sum_{k=1}^{M} f_k \)  \hspace{1cm} \text{(40)}

\( \Theta_1 = [B, a] \)  \hspace{1cm} \text{(41)}

\( \Theta_2 = [C, D] \)  \hspace{1cm} \text{(42)}

**Proof.** The derivation here draws on that presented in (Agüero 2007), which give an expression of \( Q(\theta, \hat{\theta}) \) for full parameterized state-space model:

\[
Q(\theta, \hat{\theta}) = F \log |Q^{-1}| + F \log |R^+| - tr\left[ Q^{-1}(\Phi - \Theta \Psi' - \Theta \Psi' + \Theta \Psi' + \Theta \Psi' + \Theta \Psi') \right] - tr\left[ R^{-1}(A - A \Theta_\alpha - A \Theta_\alpha + \Theta_\beta \Theta_\alpha \Theta_\alpha + \Theta_\beta \Theta_\alpha \Theta_\alpha) \right]
\]

where

\[
\Psi = [\Psi_1, \Psi_2]
\]

\[
\Theta = [A, B, C, D]
\]

Substitution of Eqs. (44) (45) into the expression \( Q(\theta, \hat{\theta}) \) in Eq. (43) then completes the proof.

The smoothed state estimate and its covariance play important roles in computing \( Q(\theta, \hat{\theta}) \). The method to calculate them can be found in (Wills, Ninness, Gibson, 2009).

4.3 M step

**Lemma 4.2** The stationary points of \( Q(\theta, \hat{\theta}) \) with respect to \( Q \) and \( R \) are given by

\[
\dot{Q} = \frac{1}{F}[T - \Theta \Psi'] - \Theta \Psi' + \Theta_\gamma \Theta_\alpha \]

\[
\dot{R} = \frac{1}{F}[A - A \Theta_\beta - A \Theta_\beta + \Theta_\beta \Theta_\beta \Theta_\alpha \Theta_\alpha]
\]

**Proof.** The first order derivatives of \( Q(\theta, \hat{\theta}) \) with respect to \( Q^{-1} \) is given by

\[
\frac{\partial Q(\theta, \hat{\theta})}{\partial Q^{-1}} = FQ^{-1} - [T - \Theta \Psi' - \Theta \Psi' + \Theta_\beta \Theta_\beta \Theta_\alpha \Theta_\alpha]
\]

which is clearly equal to zero for given by (45). The Eq. (46) can be obtained in an identical manner.

A close form solution is given for the case \( M \) is sufficient big in ((Agüero et.al, 2007) to estimate \( \Theta_{1} \) and \( \Theta_{2} \). In current paper, there is no the term \( \log |Q^{-1}| \) since \( A \) is known. Therefore, the close form can be used for any data length.

5 CONTINUOUS TIME IDENTIFICATION

Identifying a continuous time model in frequency domain may lead to numerical problem due to broad frequency band or high model order. Several attempts have been made in the past to eliminate numerical degeneracy in frequency domain.

It is well known that discrete time model comparing with continuous time model lead to better conditioned algorithm since powers of \( e^{i\alpha} \) form a natural orthogonal basis. Therefore, an alternative approach is to solve the continuous time problem in discrete time domain by using bilinear transformation. The basic mapping relation between \( s \) domain and \( z \) domain is defined as:

\[
s = \frac{2(z - 1)}{T(z + 1)} \Rightarrow z = \frac{2 + sT}{2 - sT}
\]

where \( T \) is a user defined constant.

Given the data set \( U_k, Y_k(k=1,2,\cdots M) \), a continuous time model \((A, B, C, D)\) is obtained by first finding a discrete time model \((A, B, C, D)\) at scaled frequency point \( \theta_j \) via

\[
\omega_j = 2\arctan\left(\frac{\omega_j T}{2}\right)
\]

Then, by use of the inverse mapping (McKelvey et.al, 1996) the continuous time model is derived as:

\[
A_j = \frac{2}{T}(I + A)^{-1}(I - A), \quad B_j = \frac{2}{\sqrt{T}}(I + A)^{-1}B
\]

\[
C_j = \frac{2}{\sqrt{T}}C(I + A)^{-1}, \quad D_j = D - C(I + A)^{-1}B
\]

6 NUMERICAL EXAMPLE

To investigate the properties of the presented algorithms, we will study the theoretical case when the true system can be described with the chosen model structure.

![Fig. 1. Three DOF system with three input and three output](image-url)

The system is a vibrating structure with three discrete masses connected with linear spring and damper elements, \( m_1=m_2=0.1, K_1=K_2=0.8\text{KN/m}, K_3=1\text{KN/m}, C_1=C_2=2\text{Ns/m}, C_3=0.2\text{Ns/m}, \text{and} \ C_4=0.8\text{Ns/m}. \) Three collocated sensor and actuator pairs give a 3×3 MIMO system. The inputs and outputs of this system are force and accelerations, which are represented as \( u_{i,j}(i=1,2,3) \). The random Gaussian sequence is used as input with variance \( \sigma_i^2=0.01I \). The noise sequence are zero mean random Gaussian sequence, independent of the input, and with covariance \( \Omega=0I_n, R=1I_j \). We collect \( N=8192 \) input/output data points for system identification. We run 100 Monte-Carlo simulations with different seed of noise. In each run, the results of first algorithm are used as initial estimate of the second algorithm.
To assess the quality of estimated models, the modelling error $\varepsilon(\hat{\theta})$ is computed via

$$
\varepsilon(\hat{\theta}) = \det \left( \frac{1}{M} \sum_{k=1}^{M} E_k(\hat{\theta}) E_k(\hat{\theta})^H \right) \quad (50)
$$

and compared with the determinant of the sample covariance

$$
\nu = \det \left( \frac{1}{M} \sum_{k=1}^{M} V_k V_k^H \right) \quad (53)
$$

If $\hat{\theta}$ is a good estimate then $\varepsilon(\hat{\theta}) \approx \nu$ is expected. In current work, if $\varepsilon(\hat{\theta}) > 1.3\nu$, this run is recorded as failure. For first algorithm total 7 failures are recorded. By contrast, there is no failure recorded for the second algorithm.

Fig. 2 and 3 shows the estimated model using two algorithms (100 runs) comparing with real model for one of transfer functions (the others are omitted for brevity). Clearly, they all provide good estimate, and the second algorithm based on ML yields better results.

**REFERENCES**


